Model-based reinforcement learning for approximate optimal regulation ⋆

Rushikesh Kamalapurkar [a], Patrick Walters [a], and Warren E. Dixon [a]

[a] *Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, USA*

Abstract

Reinforcement learning (RL)-based online approximate optimal control methods applied to deterministic systems typically require a restrictive persistence of excitation (PE) condition for convergence. This paper develops a concurrent learning (CL)-based implementation of model-based RL to solve approximate optimal regulation problems online under a PE-like rank condition. The development is based on the observation that, given a model of the system, RL can be implemented by evaluating the Bellman error at any number of desired points in the state space. In this result, a parametric system model is considered, and a CL-based parameter identifier is developed to compensate for uncertainty in the parameters. Uniformly ultimately bounded regulation of the system states to a neighborhood of the origin, and convergence of the developed policy to a neighborhood of the optimal policy are established using a Lyapunov-based analysis, and simulation results are presented to demonstrate the performance of the developed controller.

Key words: model-based reinforcement learning; concurrent learning; simulated experience; data-based control; adaptive control; system identification

1 Introduction

Reinforcement learning (RL) enables a cognitive agent to learn desirable behavior from interactions with its environment. In control theory, the desirable behavior is typically quantified using a cost function, and the control problem is formulated as the desire to find the optimal policy that minimizes a cumulative cost. RL techniques for discrete time systems are inherently model-free, and hence, have been a prime focus of research over the past few decades [1].

Recently, various RL-based techniques have been developed to approximately solve optimal control problems for continuous-time and discrete-time deterministic systems [2–12]. The approximate solution is facilitated via value function approximation, where the optimal policy is computed based on an estimate of the value function.

⋆ 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.

Email addresses: rkamalapurkar@ufl.edu (Rushikesh Kamalapurkar), walters8@ufl.edu (Patrick Walters), wdixon@ufl.edu (Warren E. Dixon).

Methods that seek online solutions to optimal control problems are comparable to adaptive control (cf., [3, 8, 10, 12–14] and the references therein). In adaptive control, the estimates for the uncertain parameters in the plant model are updated using the tracking error as a performance metric; whereas, in online RL-based techniques, estimates for the uncertain parameters in the value function are updated using the Bellman error (BE) as a performance metric. Typically, to establish regulation or tracking, adaptive control methods do not require the adaptive estimates to convergence to the true values. However, convergence of the RL-based controller to a neighborhood of the optimal controller requires convergence of the parameter estimates to a neighborhood of their ideal values.

Parameter convergence has been a focus of research in adaptive control for several decades. It is common knowledge that least squares and gradient descent-based update laws generally require persistence of excitation (PE) in the system state for convergence of the parameter estimates. Modification schemes such as projection algorithms, $\sigma-$modification, and $e-$modification are used to guarantee boundedness of parameter estimates and overall system stability; however, these modifications do not guarantee parameter convergence unless the PE condition is satisfied [15–18].

In RL-based approximate online optimal control, the Hamilton-Jacobi-Bellman (HJB) equation along with an estimate of the state derivative (cf. [7, 10]), or an integral form of the HJB equation (cf. [19]) is utilized to approximately evaluate the BE along the system trajectory. The BE, evaluated at a point, provides an indirect measure of the quality of the estimate of the value function evaluated at that point. Hence, the unknown value function parameters are updated based on evaluation of the BE along the system trajectory. Such weight update strategies create two challenges for analyzing convergence. The system states need to satisfy PE, and the system trajectory needs to visit enough points in the state space to generate a good approximation of the value function over the entire domain of operation. These challenges are typically addressed in the related literature (cf. [5, 8, 10, 20–26]) by adding an exploration signal to the control input to ensure sufficient exploration of the domain of operation. However, no analytical methods exist to compute the appropriate exploration signal when the system dynamics are nonlinear.

The aforementioned challenges arise from the restriction that the BE can only be evaluated along the system trajectories. In particular, the integral BE is meaningful as a measure of quality of the value function estimate only if it is evaluated along the system trajectories, and state derivative estimators can only generate numerical estimates of the state derivative along the system trajectories. Recently, [25] demonstrated that experience replay can be used to improve data efficiency in online approximate optimal control by reuse of recorded data. However, since the data needs to be recorded along the system trajectory, the system trajectory under the designed approximate optimal controller needs to provide enough excitation for learning. In general, such excitation is not available; hence, the simulation results in [25] are generated using an added probing signal.

In this paper, and in our preliminary work in [27], a different approach is used to improve data efficiency by observing that if the system dynamics are known, the state derivative, and hence, the BE can be evaluated at any desired point in the state space. Unknown parameters in the value function can therefore be adjusted based on least square minimization of the BE evaluated at any number of arbitrary points in the state space. For example, in an infinite horizon regulation problem, the BE can be computed at points uniformly distributed in a neighborhood around the origin of the state space. The results of this paper indicate that convergence of the unknown parameters in the value function is guaranteed provided the selected points satisfy a rank condition. Since the BE can be evaluated at any desired point in the state space, sufficient exploration can be achieved by appropriately selecting the points to cover the domain of operation. If the system dynamics are partially unknown, an approximation to the BE can be evaluated at any desired point in the state space based on an estimate of the system dynamics.

If each new evaluation of the BE along the system trajectory is interpreted as gaining experience via exploration, the use of a model to evaluate the BE at an unexplored point in the state space can be interpreted as a simulation of the experience. Learning based on simulation of experience has been investigated in results such as [28–33] for stochastic model-based RL; however, these results solve the optimal control problem off-line in the sense that repeated learning trials need to be performed before the algorithm learns the controller, and system stability during the learning phase is not analyzed. This paper furthers the state of the art for nonlinear, control affine plants with linearly parameterizable (LP) uncertainties in the drift dynamics by providing an online solution to deterministic infinite horizon optimal regulation problems. In this paper, a CL-based parameter estimator is developed to exponentially identify the unknown parameters in the system model, and the parameter estimates are used to implement simulation of experience by extrapolating the BE.

The main contributions of this paper include a novel implementation of model-based RL in deterministic nonlinear systems and a detailed stability analysis that establishes simultaneous online identification of system dynamics and online approximate learning of the optimal controller, while maintaining system stability. Simulation results are provided that demonstrate the approximate solution of infinite horizon optimal regulation problems online for inherently unstable nonlinear systems with uncertain drift dynamics. The simulations also demonstrate that the developed method can be used to implement RL without the addition of a probing signal.

2 Problem Formulation

Consider a control affine nonlinear dynamical system [1]

$$\dot{x}(t) = f(x(t)) + g(x(t))\,u(t), \tag{1}$$

where $x : \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ denotes the system state trajectory, $u : \mathbb{R}_{\geq t_0} \to \mathbb{R}^m$ denotes the control input, $f : \mathbb{R}^n \to \mathbb{R}^n$ denotes the drift dynamics, and $g : \mathbb{R}^n \to \mathbb{R}^{n\times m}$ denotes the control effectiveness. In the following, the notation $\phi^u(t; t_0, x^o)$ denotes a trajectory of the system in (1) under the controller u with the initial condition $x^o \in \mathbb{R}^n$ and initial time $t_0 \in \mathbb{R}_{\geq 0}$. [2] The objective is

[1] For notational brevity, unless otherwise specified, the domain of all the functions is assumed to be $\mathbb{R}_{\geq 0}$, where $\mathbb{R}_{\geq a}$ denotes the interval $[a, \infty)$. The notation $\|\cdot\|$ denotes the Euclidean norm for vectors and the Frobenius norm for matrices. The notation $(\cdot)^o$ denotes arbitrary variables.

[2] Whenever the initial time and state are implied or unimportant, a trajectory of the system in (1) evaluated at time t will be denoted by $x(t)$.

to solve the infinite horizon optimal regulation problem online, i.e., to find the optimal policy $u^* : \mathbb{R}^n \to \mathbb{R}^m$ defined as

$$u^*(x^o) \triangleq \underset{u(\tau)\in U|\tau\in\mathbb{R}_{\geq t}}{\arg\min} \int_t^\infty r\left(\phi^u(\tau;t,x^o),u(\tau)\right)d\tau, \quad (2)$$

while regulating the system states to the origin.[3] In (2), $U \in \mathbb{R}^m$ denotes the action space and $r : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\geq 0}$ denotes the instantaneous cost defined as $r(x^o,u^o) \triangleq x^{oT}Qx + u^{oT}Ru^o$, where $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ are constant positive definite symmetric matrices. The class of nonlinear systems considered in this paper is characterized by the following assumption.

Assumption 1 *The drift dynamics f is an unknown, LP locally Lipschitz function such that $f(0) = 0$, and the control effectiveness g is a known bounded locally Lipschitz function. Furthermore, $f' : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ is continuous, where $(\cdot)'$ denotes the partial derivative with respect to the first argument.*

A closed-form solution to the optimal control problem is formulated in terms of the optimal value function $V^* : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ defined as

$$V^*(x^o) \triangleq \min_{u(\tau)\in U|\tau\in\mathbb{R}_{\geq t}} \int_t^\infty r\left(\phi^u(\tau;t,x^o),u(\tau)\right)d\tau. \quad (3)$$

Assuming that the optimal value function is continuously differentiable, it is the unique solution to the corresponding HJB equation [34]

$$V^{*\prime}(x^o)\left(f(x^o)+g(x^o)u^*(x^o)\right) + x^{oT}Qx^o + u^{*T}(x^o)Ru^*(x^o) = 0, \quad (4)$$

for all $x^o \in \mathbb{R}^n$, with the boundary condition $V^*(0) = 0$. The optimal control law can be determined using the optimal value function as $u^*(x^o) = -\frac{1}{2}R^{-1}g^T(x^o)\left(V^{*\prime}(x^o)\right)^T$ [34].

An analytical solution of the HJB equation is generally not feasible; hence, an approximate solution is sought. An approximate solution of the HJB equation is facilitated by replacing V^* and u^* in (4) by their respective subsequently defined parametric estimates $\hat{V}\left(x^o,\hat{W}_c^o\right)$ and $\hat{u}\left(x^o,\hat{W}_a^o\right)$ to compute the BE $\delta : \mathbb{R}^{n+2L} \to \mathbb{R}$ as

$$\delta\left(x^o,\hat{W}_c^o,\hat{W}_a^o\right) = x^{oT}Qx^o + \hat{u}^T\left(x^o,\hat{W}_a^o\right)R\hat{u}\left(x^o,\hat{W}_a^o\right) + \hat{V}'\left(x^o,\hat{W}_c^o\right)\left(f(x^o)+g(x^o)\hat{u}\left(x^o,\hat{W}_a^o\right)\right), \quad (5)$$

where $\hat{W}_c^o \in \mathbb{R}^L$ and $\hat{W}_a^o \in \mathbb{R}^L$ are the estimates of the unknown parameters in the approximation of the value function, and the policy, respectively. Since the BE depends on the uncertain drift dynamics f, an estimate of the system dynamics is required to evaluate the BE at any given point $x^o \in \mathbb{R}^n$

3 Approximate Optimal Control

3.1 System identification

The main contribution of this paper is a novel implementation of simulation of experience for online approximate optimal control of deterministic nonlinear systems. If a system model is available, then the approximate optimal control technique can be implemented using the model. However, if an exact model of the system is unavailable, then parametric system identification can be employed to generate an estimate of the system model. A possible approach is to use parameters that are estimated offline in a separate experiment. A more useful approach is to use the offline estimate as the initial guess, and to employ a dynamic system identification technique capable of refining the initial guess based on input-output data.

To facilitate online system identification, let $f(x^o) = Y(x^o)\theta$ denote the linear parametrization of the function f, where $Y : \mathbb{R}^n \to \mathbb{R}^{n\times p}$ is the regression matrix and $\theta \in \mathbb{R}^p$ is the vector of constant unknown parameters. Let $\hat{\theta} \in \mathbb{R}^p$ be an estimate of the unknown parameter vector θ. The following development assumes that an adaptive system identifier that satisfies conditions detailed in Assumption 2 is available. For completeness, a concurrent learning-based system identifier that satisfies Assumption 2 is presented in Appendix A.

Assumption 2 *A compact set $\Theta \subset \mathbb{R}^p$ such that $\theta \in \Theta$ is known a priori. The estimates $\hat{\theta} : \mathbb{R}_{\geq t_0} \to \mathbb{R}^p$ are updated based on a switched update law of the form*

$$\dot{\hat{\theta}}(t) = f_{\theta s}\left(\hat{\theta}(t),t\right), \quad (6)$$

$\hat{\theta}(t_0) = \hat{\theta}_0 \in \Theta$, where $s \in \mathbb{N}$ denotes the switching index and $\{f_{\theta s} : \mathbb{R}^p \times \mathbb{R}_{\geq 0} \to \mathbb{R}^p\}_{s\in\mathbb{N}}$ denotes a family of continuously differentiable functions. The dynamics of the parameter estimation error $\tilde{\theta} : \mathbb{R}_{\geq t_0} \to \mathbb{R}^p$, defined as $\tilde{\theta}(t) \triangleq \theta - \hat{\theta}(t)$ can be expressed as $\dot{\tilde{\theta}}(t) = f_{\theta s}\left(\theta - \tilde{\theta}(t),t\right)$. Furthermore, there exists a continuously differentiable function $V_\theta : \mathbb{R}^p \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ that satisfies

$$\underline{v}_\theta\left(\left\|\tilde{\theta}^o\right\|\right) \leq V_\theta\left(\tilde{\theta}^o,t\right) \leq \overline{v}_\theta\left(\left\|\tilde{\theta}^o\right\|\right), \quad (7)$$

[3] The definition in (2) implicitly assumes existence of the optimal policy.

$$V_\theta'\left(\tilde{\theta}^o,t\right)\left(-f_{\theta s}\left(\theta-\tilde{\theta}^o,t\right)\right)+\frac{\partial V_\theta\left(\tilde{\theta}^o,t\right)}{\partial t}\leq -K\left\|\tilde{\theta}^o\right\|^2+D\left\|\tilde{\theta}^o\right\|, \tag{8}$$

for all $s\in\mathbb{N}$, $t\in\mathbb{R}_{\geq t_0}$, *and* $\tilde{\theta}^o\in\mathbb{R}^p$, *where* $\underline{v}_\theta$, $\overline{v}_\theta:\mathbb{R}_{\geq 0}\to\mathbb{R}_{\geq 0}$ *are class* $\mathcal{K}$ *functions,* $K\in\mathbb{R}_{>0}$ *is an adjustable parameter, and* $D\in\mathbb{R}_{>0}$ *is a positive constant.*[4]

Using an estimate $\hat{\theta}^o$, the BE in (5) can be approximated by $\hat{\delta}:\mathbb{R}^{n+2L+p}\to\mathbb{R}$ as

$$\hat{\delta}\left(x^o,\hat{W}_c^o,\hat{W}_a^o,\hat{\theta}^o\right)=x^{oT}Qx^o+\hat{u}^T\left(x^o,\hat{W}_a^o\right)R\hat{u}\left(x^o,\hat{W}_a^o\right)$$
$$+\hat{V}'\left(x^o,\hat{W}_c^o\right)\left(Y\left(x^o\right)\hat{\theta}^o+g\left(x^o\right)\hat{u}\left(x^o,\hat{W}_a^o\right)\right). \tag{9}$$

In the following, the approximate BE in (9) is used to obtain an approximate solution to the HJB equation in (4).

3.2 Value function approximation

Approximations to the optimal value function V^* and the optimal policy u^* are designed based on neural network (NN)-based representations. Given any compact set $\chi\subset\mathbb{R}^n$ and positive constants $\bar{\epsilon},\bar{\epsilon}'\in\mathbb{R}$, the universal approximation property of NNs can be exploited to represent the optimal value function V^* as $V^*\left(x^o\right)=W^T\sigma\left(x^o\right)+\epsilon\left(x^o\right)$, for all $x^o\in\chi$, where $W\in\mathbb{R}^L$ is the ideal weight matrix, which is bounded above by a known positive constant $\bar{W}$ in the sense that $\|W\|\leq\bar{W}$, $\sigma:\mathbb{R}^n\to\mathbb{R}^L$ is a continuously differentiable nonlinear activation function such that $\sigma(0)=0$ and $\sigma'(0)=0$, $L\in\mathbb{N}$ is the number of neurons, and $\epsilon:\mathbb{R}^n\to\mathbb{R}$ is the function reconstruction error such that $\sup_{x^o\in\chi}|\epsilon\left(x^o\right)|\leq\bar{\epsilon}$ and $\sup_{x^o\in\chi}|\epsilon'\left(x^o\right)|\leq\bar{\epsilon}'$.

Based on the NN representation of the value function a NN-based representation of the optimal controller is derived as $u^*\left(x^o\right)=-\frac{1}{2}R^{-1}g^T\left(x^o\right)\left(\sigma'^T\left(x^o\right)W+\epsilon'^T\left(x^o\right)\right)$. The NN-based approximations $\hat{V}:\mathbb{R}^n\times\mathbb{R}^L\to\mathbb{R}$ and $\hat{u}:\mathbb{R}^n\times\mathbb{R}^L\to\mathbb{R}^m$ are defined as

$$\hat{V}\left(x^o,\hat{W}_c^o\right)\triangleq\hat{W}_c^{oT}\sigma\left(x^o\right),$$
$$\hat{u}\left(x^o,\hat{W}_a^o\right)\triangleq-\frac{1}{2}R^{-1}g^T\left(x^o\right)\sigma'^T\left(x^o\right)\hat{W}_a^o, \tag{10}$$

where $\hat{W}_c^o\in\mathbb{R}^L$ and $\hat{W}_a^o\in\mathbb{R}^L$ are the estimates of W. The use of two sets of weights to estimate the same set of ideal weights is motivated by the stability analysis and the fact that it enables a formulation of the BE that is linear in the value function weight estimates $\hat{W}_c^o$, enabling a least squares-based adaptive update law.

3.3 Simulation of experience via BE extrapolation

In traditional RL-based algorithms, the value function estimate and the policy estimate are updated based on observed data. The use of observed data to learn the value function naturally leads to a sufficient exploration condition which demands sufficient richness in the observed data. In stochastic systems, this is achieved using a randomized stationary policy (cf. [7, 35, 36]), whereas in deterministic systems, a probing noise is added to the derived control law (cf. [8, 10, 37–39]).

The technique developed in this result implements simulation of experience in a model-based RL scheme by using $Y\hat{\theta}$ as an estimate of the uncertain drift dynamics f to extrapolate the approximate BE to a predefined set of points $\{x_i\in\mathbb{R}^n\mid i=1,\cdots,N\}$ in the state space. In the following, $\hat{\delta}_t:\mathbb{R}_{\geq t_0}\to\mathbb{R}$ denotes the approximate BE in (9) evaluated along the trajectories of (1), (6), (11), and (13) as $\hat{\delta}_t(t)\triangleq\hat{\delta}\left(x(t),\hat{W}_c(t),\hat{W}_a(t),\hat{\theta}(t)\right)$ and $\hat{\delta}_{ti}:\mathbb{R}_{\geq t_0}\to\mathbb{R}$ denotes the approximate BE extrapolated to the points $\{x_i\in\mathbb{R}^n\mid i=1,\cdots,N\}$ along the trajectories of (6), (11), and (13) as $\hat{\delta}_{ti}\triangleq\hat{\delta}\left(x_i,\hat{W}_c(t),\hat{W}_a(t),\hat{\theta}(t)\right)$.

A least-squares update law for the value function weights is designed based on the subsequent stability analysis as

$$\dot{\hat{W}}_c(t)=-\eta_{c1}\Gamma\frac{\omega(t)}{\rho(t)}\hat{\delta}_t(t)-\frac{\eta_{c2}}{N}\Gamma\sum_{i=1}^{N}\frac{\omega_i(t)}{\rho_i(t)}\hat{\delta}_{ti}(t), \tag{11}$$

$$\dot{\Gamma}(t)=\left(\beta\Gamma(t)-\eta_{c1}\frac{\Gamma(t)\omega(t)\omega(t)^T\Gamma(t)}{\rho^2(t)}\right)\mathbf{1}_{\{\|\Gamma\|\leq\overline{\Gamma}\}}, \tag{12}$$

$\|\Gamma(t_0)\|\leq\overline{\Gamma}$, where $\Gamma:\mathbb{R}_{\geq t_0}\to\mathbb{R}^{L\times L}$ is a time-varying least-squares gain matrix, $\omega(t)\triangleq\sigma'(x(t))\left(Y(x(t))\hat{\theta}(t)+g(x(t))\hat{u}\left(x(t),\hat{W}_a(t)\right)\right)$, $\omega_i(t)\triangleq\sigma'(x_i)\left(Y(x_i)\hat{\theta}(t)+g(x_i)\hat{u}\left(x_i,\hat{W}_a(t)\right)\right)$, $\rho(t)\triangleq 1+\nu\omega^T(t)\Gamma(t)\omega(t)$, $\rho_i(t)\triangleq 1+\nu\omega_i^T(t)\Gamma(t)\omega_i(t)$, where $\nu\in\mathbb{R}$ is a constant positive normalization gain, $\mathbf{1}_{\{\cdot\}}$ denotes the indicator function, $\overline{\Gamma}>0\in\mathbb{R}$ is a saturation constant, $\beta>0\in\mathbb{R}$ is a constant forgetting factor, and $\eta_{c1},\eta_{c2}>0\in\mathbb{R}$ are constant adaptation gains.

The policy weights are updated based on the subsequent

[4] The subsequent analysis in Section 4 indicates that when a system identifier that satisfies Assumption 2 is employed to facilitate online optimal control, the ratio $\frac{D}{K}$ needs to be sufficiently small to establish set-point regulation and convergence to optimality.

stability analysis as [5]

$$\dot{\hat{W}}_a(t) = -\eta_{a1}\left(\hat{W}_a(t) - \hat{W}_c(t)\right) - \eta_{a2}\hat{W}_a(t) + \frac{\eta_{c1} G_\sigma^T(t)\hat{W}_a(t)\omega^T(t)}{4\rho(t)}\hat{W}_c(t) + \sum_{i=1}^{N}\frac{\eta_{c2} G_{\sigma i}^T \hat{W}_a(t)\omega_i^T(t)}{4N\rho_i(t)}\hat{W}_c(t), \quad (13)$$

where $\eta_{a1}, \eta_{a2} \in \mathbb{R}$ are positive constant adaptation gains, $G_\sigma(t) \triangleq \sigma'(x(t)) g(x(t)) R^{-1} g^T(x(t)) \sigma'^T(x(t))$, $G_{\sigma i} \triangleq \sigma_i' g_i R^{-1} g_i^T \sigma_i'^T \in \mathbb{R}^{L\times L}$, where $g_i = g(x_i)$ and $\sigma_i' = \sigma'(x_i)$.

The update law in (11) ensures that the adaptation gain matrix is bounded such that

$$\underline{\Gamma} \leq \|\Gamma(t)\| \leq \overline{\Gamma}, \ \forall t \in \mathbb{R}_{\geq t_0}. \quad (14)$$

Using the weight estimates $\hat{W}_a$, the controller for the system in (1) is designed as

$$u(t) = \hat{u}\left(x(t), \hat{W}_a(t)\right). \quad (15)$$

The following rank condition facilitates the subsequent stability analysis.

Assumption 3 *There exists a finite set of fixed points* $\{x_i \in \mathbb{R}^n \mid i = 1, \cdots, N\}$ *such that* $\forall t \in \mathbb{R}_{\geq t_0}$

$$0 < \underline{c} \triangleq \frac{1}{N}\left(\inf_{t\in\mathbb{R}_{\geq t_0}}\left(\lambda_{\min}\left\{\sum_{i=1}^{N}\frac{\omega_i(t)\omega_i^T(t)}{\rho_i(t)}\right\}\right)\right), \quad (16)$$

where $\lambda_{min}\{\cdot\}$ *denotes the minimum eigenvalue.*

The rank condition in (16) depends on the estimates $\hat{\theta}$ and $\hat{W}_a$; hence, in general, it is impossible to guarantee a priori. However, unlike the PE condition in previous results such as [8, 10, 37–39], the condition in (16) can be verified online at each time t. Furthermore, the condition in (16) can be heuristically met by collecting redundant data, i.e., by selecting more points than the number of neurons by choosing $N \gg L$.

The update law in (11) is fundamentally different from the CL adaptive update in results such as [41, 42], in the sense that the points $\{x_i \in \mathbb{R}^n \mid i = 1, \cdots, N\}$ are selected a priori based on prior information about the desired behavior of the system. Given the system dynamics, or an estimate of the system dynamics, the approximate BE can be extrapolated to any desired point in the state space, whereas in adaptive control, the prediction error is used as a metric which can only be evaluated at observed data points along the state trajectory.

[5] Using the fact that the ideal weights are bounded, a projection-based (cf. [40]) update law $\dot{\hat{W}}_a(t) = proj\left\{-\eta_{a1}\left(\hat{W}_a(t) - \hat{W}_c(t)\right)\right\}$ can be utilized to update the policy weights. Since the policy weights are bounded a priori by the projection algorithm, a less complex stability analysis can be used to establish the result in Theorem 1.

4 Stability analysis

For notational brevity, the dependence of all the functions on the system states and time is suppressed hereafter unless required for clarity of exposition. To facilitate the subsequent stability analysis, the approximate BE is expressed in terms of the weight estimation errors $\tilde{W}_c \triangleq W - \hat{W}_c$ and $\tilde{W}_a \triangleq W - \hat{W}_a$. Subtracting (4) from (9), an unmeasurable form of the instantaneous BE can be expressed as

$$\hat{\delta}_t = -\omega^T\tilde{W}_c - W^T\sigma' Y\tilde{\theta} + \frac{1}{4}\tilde{W}_a^T G_\sigma \tilde{W}_a + \frac{1}{4}G_\epsilon - \epsilon' f + \frac{1}{2}W^T\sigma' G\epsilon'^T, \quad (17)$$

where $G \triangleq gR^{-1}g^T \in \mathbb{R}^{n\times n}$ and $G_\epsilon \triangleq \epsilon' G \epsilon'^T \in \mathbb{R}$. Similarly, the approximate BE evaluated at the sampled states $\{x_i \mid i = 1, \cdots, N\}$ can be expressed as

$$\hat{\delta}_{ti} = -\omega_i^T\tilde{W}_c + \frac{1}{4}\tilde{W}_a^T G_{\sigma i}\tilde{W}_a - W^T\sigma_i' Y_i\tilde{\theta} + \Delta_i, \quad (18)$$

where $Y_i = Y(x_i)$, $\epsilon_i' = \epsilon'(x_i)$, $f_i = f(x_i)$, $G_i \triangleq g_i R^{-1} g_i^T \in \mathbb{R}^{n\times n}$, $G_{\epsilon i} \triangleq \epsilon_i' G_i \epsilon_i'^T \in \mathbb{R}$, and $\Delta_i \triangleq \frac{1}{2}W^T\sigma_i' G_i \epsilon_i'^T + \frac{1}{4}G_{\epsilon i} - \epsilon_i' f_i \in \mathbb{R}$ is a constant.

On any compact set $\chi \subset \mathbb{R}^n$ the function Y is Lipschitz continuous, and hence, there exists a positive constant $L_Y \in \mathbb{R}$ such that [6]

$$\|Y\| \leq L_Y \|x\|, \forall x \in \chi. \quad (19)$$

Using (14), the normalized regressor $\frac{\omega}{\rho}$ can be bounded as

$$\sup_{t\in\mathbb{R}_{\geq t_0}}\left\|\frac{\omega}{\rho}\right\| \leq \frac{1}{2\sqrt{\nu\underline{\Gamma}}}. \quad (20)$$

For brevity of notation, given a compact set $\chi \subset \mathbb{R}^n$, the operator $\overline{(\cdot)} \triangleq \sup_{x\in\chi}(\cdot) : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ and the following positive constants are defined.

$$\vartheta_1 \triangleq \frac{\eta_{c1} L_Y \|\theta\| \overline{\epsilon'}}{4\sqrt{\nu\underline{\Gamma}}}, \quad \vartheta_2 \triangleq \sum_{i=1}^{N}\left(\frac{\eta_{c2}\|\sigma_i' Y_i\|\overline{W}}{4N\sqrt{\nu\underline{\Gamma}}}\right),$$

[6] The Lipschitz property is exploited here for clarity of exposition. The bound in (19) can be easily generalized to $\|Y(x)\| \leq L_Y(\|x\|)\|x\|$, where $L_Y : \mathbb{R} \to \mathbb{R}$ is a positive, non-decreasing function.

$$\vartheta_3 \triangleq \frac{L_Y \eta_{c1} \overline{W} \overline{\|\sigma'\|}}{4\sqrt{\nu \underline{\Gamma}}}, \quad \vartheta_4 \triangleq \overline{\left\| \frac{1}{4} G_\epsilon \right\|},$$
$$\vartheta_5 \triangleq \frac{\eta_{c1} \overline{\|2W^T \sigma' G \epsilon'^T + G_\epsilon\|}}{8\sqrt{\nu \underline{\Gamma}}} + \left\| \sum_{i=1}^{N} \frac{\eta_{c2} \omega_i \Delta_i}{N \rho_i} \right\|,$$
$$\vartheta_6 \triangleq \overline{\left\| \frac{1}{2} W^T G_\sigma + \frac{1}{2} \epsilon' G^T \sigma'^T \right\|} + \vartheta_7 \overline{W}^2 + \eta_{a2} \overline{W},$$
$$\vartheta_7 \triangleq \frac{\eta_{c1} \overline{\|G_\sigma\|}}{8\sqrt{\nu \underline{\Gamma}}} + \sum_{i=1}^{N} \left(\frac{\eta_{c2} \|G_{\sigma i}\|}{8N\sqrt{\nu \underline{\Gamma}}} \right), \quad \underline{q} \triangleq \lambda_{\min}\{Q\},$$
$$v_l = \frac{1}{2} \min \left(\frac{\underline{q}}{2}, \frac{\eta_{c2} \underline{c}}{3}, \frac{\eta_{a1} + 2\eta_{a2}}{6}, \frac{K}{4} \right),$$
$$\iota = \frac{3\vartheta_5^2}{4\eta_{c2}\underline{c}} + \frac{3\vartheta_6^2}{2(\eta_{a1} + 2\eta_{a2})} + \frac{D^2}{2K} + \vartheta_4. \tag{21}$$

Let $Z : \mathbb{R}_{\geq t_0} \to \mathbb{R}^{n+2L+p}$ denote the concatenated trajectories of $\dot{Z}(t) = h(Z(t), t)$, defined as $Z(t) \triangleq \left[x^T(t), \tilde{W}_c^T(t), \tilde{W}_a^T(t), \tilde{\theta}^T(t) \right]^T$, where $h : \mathbb{R}^{n+2L+p} \times \mathbb{R}_{\geq t_0} \to \mathbb{R}^{n+2L+p}$ is a concatenation of the dynamics in (1), (6), (11), and (13). The sufficient conditions for ultimate boundedness of Z are derived based on the subsequent stability analysis as

$$\frac{\eta_{a1} + 2\eta_{a2}}{6} > \vartheta_7 \overline{W} \left(\frac{2\zeta_2 + 1}{2\zeta_2} \right), \quad \frac{K}{4} > \frac{\vartheta_2 + \zeta_1 \zeta_3 \vartheta_3 \overline{Z}}{\zeta_1},$$
$$\frac{\eta_{c2}}{3} > \frac{\zeta_2 \vartheta_7 \overline{W} + \eta_{a1} + 2\left(\vartheta_1 + \zeta_1 \vartheta_2 + (\vartheta_3/\zeta_3)\overline{Z}\right)}{2\underline{c}},$$
$$\frac{\underline{q}}{2} > \vartheta_1, \tag{22}$$

where $\overline{Z} \triangleq \underline{v}^{-1}\left(\overline{v}\left(\max\left(\|Z(t_0)\|, \sqrt{\frac{\iota}{v_l}} \right) \right) \right)$, $\zeta_1, \zeta_2, \zeta_3 \in \mathbb{R}$ are known positive adjustable constants, and $\underline{v}$ and $\overline{v}$ are subsequently defined class $\mathcal{K}$ functions. The Lipschitz constants in (19) and the NN function approximation errors depend on the underlying compact set; hence, given a bound on the initial condition $Z(t_0)$ for the concatenated state Z, a compact set that contains the concatenated state trajectory needs to be established before adaptation gains satisfying the conditions in (22) can be selected. In the following, based on the subsequent stability analysis, an algorithm is developed to compute the required compact set, denoted by $\mathcal{Z} \subset \mathbb{R}^{2n+2L+p}$. In Algorithm 1, the notation $\{(\cdot)\}_i$ denotes the value of $(\cdot)$ computed in the i^{th} iteration. Since the constants ι and v_l depend on L_Y only through the products $L_Y \overline{\epsilon}'$ and $\frac{L_Y}{\zeta_3}$, Algorithm 1 ensures that

$$\sqrt{\frac{\iota}{v_l}} \leq \frac{1}{2} \text{diam}(\mathcal{Z}), \tag{23}$$

where $\text{diam}(\mathcal{Z})$ denotes the diameter of the set $\mathcal{Z}$ defined as $\text{diam}(\mathcal{Z}) \triangleq \sup\{\|x - y\| \mid x, y \in \mathcal{Z}\}$. The main result of this paper can now be stated as follows.

Algorithm 1 Gain Selection

First iteration:

Given $\overline{z} \in \mathbb{R}_{\geq 0}$ such that $\|Z(t_0)\| < \overline{z}$, let $\mathcal{Z}_1 \triangleq \left\{ \xi \in \mathbb{R}^{2n+2L+p} \mid \|\xi\| \leq \underline{v}^{-1}(\overline{v}(\overline{z})) \right\}$. Using $\mathcal{Z}_1$, compute the bounds in (21) and select the gains according to (22). If $\left\{ \sqrt{\frac{\iota}{v_l}} \right\}_1 \leq \overline{z}$, set $\mathcal{Z} = \mathcal{Z}_1$ and terminate.

Second iteration:

If $\overline{z} < \left\{ \sqrt{\frac{\iota}{v_l}} \right\}_1$, let $\mathcal{Z}_2 \triangleq \left\{ \xi \in \mathbb{R}^{2n+2L+p} \mid \|\xi\| \leq \underline{v}^{-1}\left(\overline{v}\left(\left\{ \sqrt{\frac{\iota}{v_l}} \right\}_1 \right) \right) \right\}$. Using $\mathcal{Z}_2$, compute the bounds in (21) and select the gains according to (22). If $\left\{ \frac{\iota}{v_l} \right\}_2 \leq \left\{ \frac{\iota}{v_l} \right\}_1$, set $\mathcal{Z} = \mathcal{Z}_2$ and terminate.

Third iteration:

If $\left\{ \frac{\iota}{v_l} \right\}_2 > \left\{ \frac{\iota}{v_l} \right\}_1$, increase the number of NN neurons to $\{L\}_3$ to ensure $\{L_Y\}_2 \{\overline{\epsilon}'\}_3 \leq \{L_Y\}_2 \{\overline{\epsilon}'\}_2, \forall i = 1, .., N$, increase the constant ζ_3 to ensure $\frac{\{L_Y\}_2}{\{\zeta_3\}_3} \leq \frac{\{L_Y\}_2}{\{\zeta_3\}_2}$, and increase the gains K and η_{a1} to satisfy the gain conditions in (22). Provided the constant $\underline{c}$ is large enough and D is small enough, these adjustments ensure $\{\iota\}_3 \leq \{\iota\}_2$. Set $\mathcal{Z} = \left\{ \xi \in \mathbb{R}^{2n+2L+p} \mid \|\xi\| \leq \underline{v}^{-1}\left(\overline{v}\left(\left\{ \sqrt{\frac{\iota}{v_l}} \right\}_2 \right) \right) \right\}$ and terminate.

Theorem 1 *Provided Assumptions (1) - (3) hold and gains $\underline{q}$, η_{c2}, η_{a2}, and K are selected large enough using Algorithm 1, the controller in (15) along with the adaptive update laws in (11) and (13) ensure that the state x, the value function weight estimation error $\tilde{W}_c$, and the policy weight estimation error $\tilde{W}_a$ are uniformly ultimately bounded (UUB).*

PROOF. Let $V_L : \mathbb{R}^{n+2L+p} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a continuously differentiable positive definite candidate Lyapunov function defined as

$$V_L(Z^o, t) \triangleq V^*(x^o) + \frac{1}{2} \tilde{W}_c^{oT} \Gamma^{-1}(t) \tilde{W}_c^o + \frac{1}{2} \tilde{W}_a^{oT} \tilde{W}_a^o + V_\theta\left(\tilde{\theta}^o, t \right), \tag{24}$$

where V^* is the optimal value function, V_θ was introduced in Assumption 2, and $Z^o \triangleq \left[x^{oT}, \tilde{W}_c^{oT}, \tilde{W}_a^{oT}, \tilde{\theta}^{oT} \right]^T$. Using the fact that V^* is positive definite, (7), (14) and Lemma 4.3 from [43] yield

$$\underline{v}(\|Z^o\|) \leq V_L(Z^o, t) \leq \overline{v}(\|Z^o\|), \tag{25}$$

for all $t \in \mathbb{R}_{\geq t_0}$ and for all $Z^o \in \mathbb{R}^{n+2L+p}$, where $\underline{v}$, $\overline{v} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class $\mathcal{K}$ functions.

Provided the gains are selected based using Algorithm 1, substituting for the approximate BEs from (17) and (18), using the bounds in (19) and (20), and using Young's inequality, the time derivative of (24) evaluated along the trajectory Z can be upper-bounded as

$$V_L'(Z^o,t)\,h(Z^o,t)+\frac{\partial V_L(Z^o,t)}{\partial t}\leq -v_l\left\|Z^o\right\|^2, \quad (26)$$

for all $\|Z^o\|\geq\sqrt{\frac{\iota}{v_l}}>0$, $Z^o\in\mathcal{Z}$ and $t\in\mathbb{R}_{\geq t_0}$.[7] Using (25), (23) and (26), Theorem 4.18 in [43] can now be invoked to conclude that Z is UUB in the sense that $\limsup_{t\to\infty}\|Z(t)\|\leq \underline{v}^{-1}\left(\overline{v}\left(\sqrt{\frac{\iota}{v_l}}\right)\right)$. Furthermore, the concatenated state trajectories are bounded such that $\|Z(t)\|\leq\overline{Z}$ for all $t\in\mathbb{R}_{\geq t_0}$. Since the estimates $\hat{W}_a$ approximate the ideal weights W, the policy $\hat{u}$ approximates the optimal policy u^*.

5 Simulation

This section presents two simulations to demonstrate the performance and the applicability of the developed technique. First, the performance of the developed controller is demonstrated through an approximate solution of an optimal control problem that has a known analytical solution. Based on the known solution, an exact polynomial basis is used for value function approximation. The second simulation demonstrates the applicability of the developed technique in the case where the analytical solution, and hence, the basis for value function approximation is unknown. In this case, since the optimal solution is unknown, the optimal trajectories obtained using the developed technique are compared with optimal trajectories obtained through numerical optimal control techniques.

5.1 Problem with a known basis

The performance of the developed controller is demonstrated by simulating a nonlinear, control affine system with a two dimensional state $x=[x_1,\ x_2]^T$. The system dynamics are described by (1), where [8]

$$f=\begin{bmatrix} x_1 & x_2 & 0 & 0\\ 0 & 0 & x_1 & x_2\left(1-(\cos(2x_1)+2)^2\right)\end{bmatrix}\begin{bmatrix}a\\b\\c\\d\end{bmatrix},$$

$$g=\begin{bmatrix}0\\ (\cos(2x_1)+2)^T\end{bmatrix}. \quad (27)$$

In (27) a, b, c, $d\in\mathbb{R}$ are positive unknown parameters. The parameters are selected as[8] $a=-1$, $b=1$, $c=-0.5$, and $d=-0.5$. The control objective is to minimize the cost in (3), where $Q=I_{2\times 2}$ and $R=1$, where $I_{n\times n}$ denotes an $n\times n$ identity matrix. The optimal value function and optimal control for the system in (27) are given by $V^*(x)=\frac{1}{2}x_1^2+x_2^2$, and $u^*(x)=-(\cos(2x_1)+2)x_2$ (cf. [8]).

To facilitate the identifier design, thirty data points are recorded using a singular value maximizing algorithm (cf. [42]) for the CL-based adaptive update law in (A.2). The state derivative at the recorded data points is computed using a fifth order Savitzky-Golay smoothing filter (cf. [44]).

To facilitate the ADP-based controller, the basis function $\sigma:\mathbb{R}^2\to\mathbb{R}^3$ for value function approximation is selected as $\sigma=\left[x_1^2,\ x_1x_2,\ x_2^2\right]$. Based on the analytical solution, the ideal weights are $W=[0.5,\ 0,\ 1]^T$. The data points for the CL-based update law in (11) are selected to be on a 5×5 grid around the origin. The learning gains are selected as $\eta_{c1}=1$, $\eta_{c2}=15$, $\eta_{a1}=100$, $\eta_{a2}=0.1$, and $\nu=0.005$. The gains for the system identifier developed in Appendix A are selected as $k_x=10I_{2\times 2}$, $\Gamma_\theta=20I_{4\times 4}$, and $k_\theta=30$. The policy and the value function weight estimates are initialized using a stabilizing set of initial weights as $\hat{W}_c(0)=\hat{W}_a(0)=[1,\ 1,\ 1]^T$ and the least squares gain is initialized as $\Gamma(0)=100I_{3\times 3}$. The initial condition for the system state is selected as $x(0)=[-1,\ -1]^T$, the state estimates $\hat{x}$ are initialized to be zero, the parameter estimates $\hat{\theta}$ are initialized to be one , and the data stack for CL is recorded online.

Figure 1 demonstrates that the system state is regulated to the origin, the unknown parameters in the drift dynamics are identified, and the value function and the policy weights converge to their true values. Furthermore, unlike previous results, a probing signal to ensure PE is not required. Figure 2 demonstrates the satisfaction of Assumptions 3 and 4.

5.2 Problem with an unknown basis

To demonstrate the applicability of the developed controller, a nonlinear, control affine system with a four dimensional state $x=[x_1,\ x_2,\ x_3,\ x_4]^T$ is simulated. The system dynamics are described in [45, Equation 31], with the states selected as[9] $x_1=q_1$, $x_2=q_2$, $x_3=\dot{q}_1$, and $x_4=\dot{q}_2$, and the inertia parameters p_1, p_2, and p_3, and the friction coefficients are considered unknown. The

[7] Since V_θ is a common Lyapunov function for the switched subsystem in (6), and the terms introduced by the update law (12) do not contribute to the bound in (26), V_L is a common Lyapunov function for the complete error system.

[8] The origin is an unstable equilibrium point of the unforced system $\dot{x}=f(x)$.

[9] The origin is a marginally stable equilibrium point of the unforced system $\dot{x}=f(x)$.

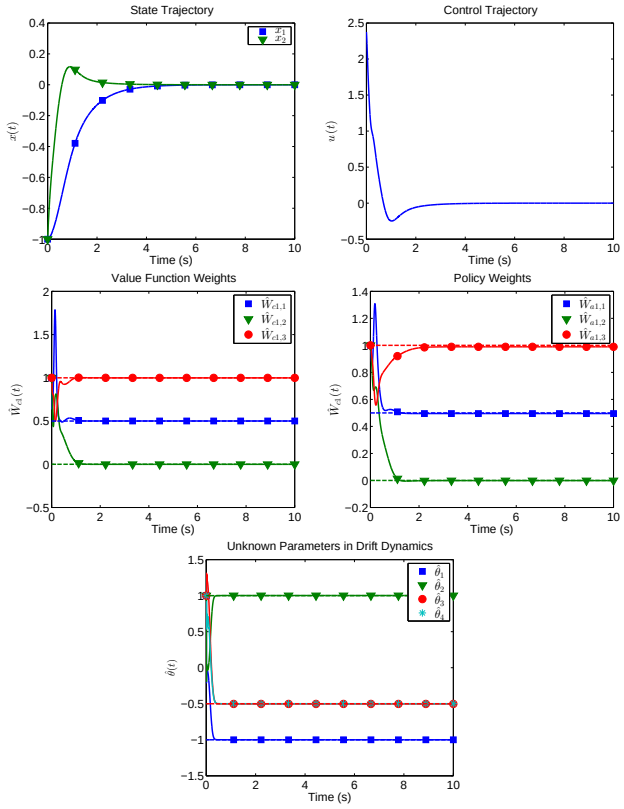

Figure 1. System trajectories generated using the developed technique, and compared to the analytical solution.

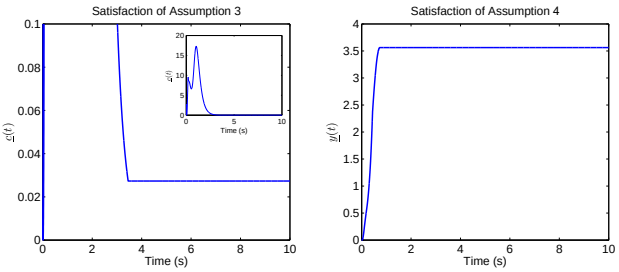

Figure 2. Satisfaction of Assumptions 3 and 4 for the simulation with known basis.

control objective is to minimize the cost in (3), where $Q = \text{diag}([10,\ 10,\ 1,\ 1])$ and $R = \text{diag}([1,\ 1])$.

The basis function $\sigma : \mathbb{R}^4 \rightarrow \mathbb{R}^{10}$ for value function approximation is selected as $\sigma(x) = \left[x_1x_3,\ x_2x_4,\ x_3x_2,\ x_4x_1,\ x_1x_2,\ x_4x_3,\ x_1^2,\ x_2^2,\ x_3^2,\ x_4^2\right]$. The data points for the CL-based update law in (11) are selected to be on a $3 \times 3 \times 3 \times 3$ grid around the origin, and the policy weights are updated using a projection-based update law. The learning gains are selected as $\eta_{c1} = 1$, $\eta_{c2} = 30$, $\eta_{a1} = 0.1$, and $\nu = 0.0005$. The gains for the system identifier developed in Appendix A are selected as $k_x = 10I_{4\times4}$, $\Gamma_\theta = \text{diag}([90,\ 50,\ 160,\ 50])$, and $k_\theta = 1.1$. The least squares gain is initialized as $\Gamma(0) = 1000I_{10\times10}$ and the policy and the

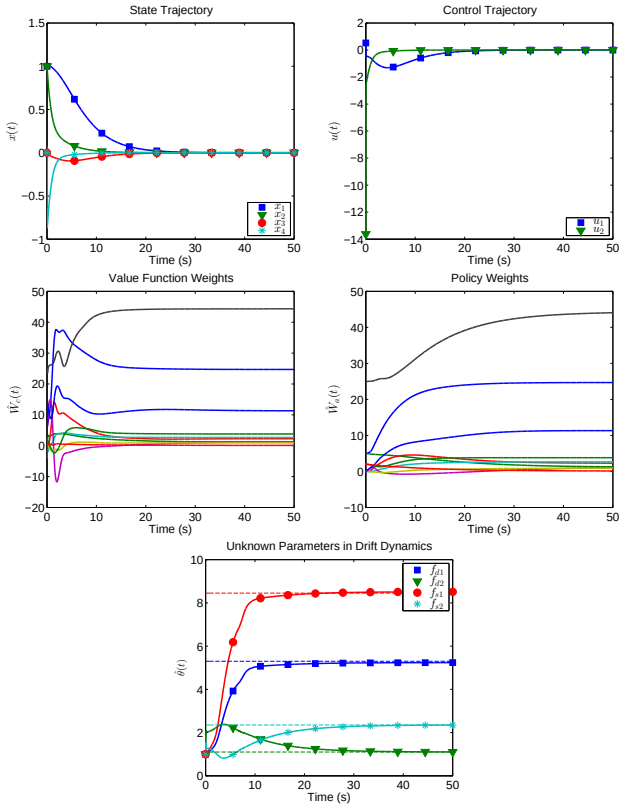

Figure 3. System trajectories generated using the developed technique, where the drift parameter estimates are compared to the actual drift parameters.

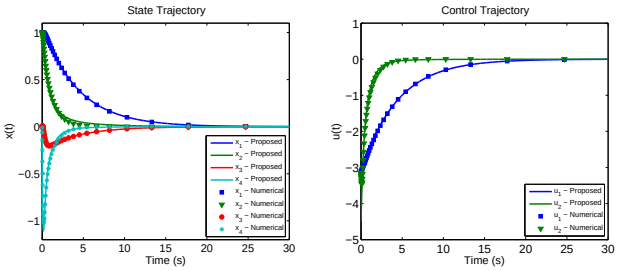

Figure 4. State and control trajectories generated using feedback policy $\hat{u}^*(x)$ compared to a numerical optimal solution.

value function weight estimates are initialized as $\hat{W}_c(0) = \hat{W}_a(0) = [5,\ 5,\ 0,\ 0,\ 0,\ 0,\ 25,\ 0,\ 2\ ,2]^T$. The initial condition for the system state is selected as $x(0) = [1,\ 1,\ 0,\ 0]^T$, the state estimates $\hat{x}$ are initialized to be zero, the parameter estimates $\hat{\theta}$ are initialized to be one, and the data stack for CL is recorded online.

Figure 3 demonstrates that the system state is regulated to the origin, the unknown parameters in the drift dynamics are identified, and the value function and the policy weights converge. Figure 5 demonstrates the satisfaction of Assumptions 3 and 4. The value function and the policy weights converge to the following values.

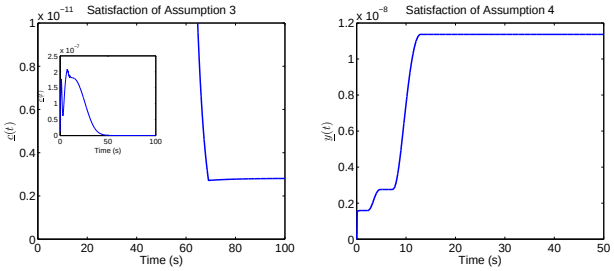

Figure 5. Satisfaction of Assumptions 3 and 4 for the simulation with unknown basis.

$$\hat{W}_c^* = \hat{W}_a^* = [24.7,\ 1.19,\ 2.25,\ 2.67,\ 1.18,\ 0.93,\ 44.34,\ 11.31,\ 3.81\,, 0.10]^T. \quad (28)$$

Since the true values of the value function weights are unknown, the weights in (28) can not be compared to their true values. However, a measure of proximity of the weights in (28) to the ideal weights W can be obtained by comparing the system trajectories resulting from applying the feedback control policy $\hat{u}^*(x) = -\frac{1}{2}R^{-1}g^T(x)\,\sigma'^T(x)\,\hat{W}_a^*$ to the system, against numerically computed optimal system trajectories. In Figure 4, the numerical optimal solution is obtained using an infinite-horizon Gauss pseudospectral method (cf. [46]) using 45 collocation points. Figure 4 indicates that the weights in (28) generate state and control trajectories that closely match the numerically computed optimal trajectories.

6 Conclusion

An online approximate optimal controller is developed, where the value function is approximated without PE in the system states via novel use of a model to evaluate the BE over unexplored areas of the state-space. The PE condition is replaced by a set of rank conditions that can be verified online using current and recorded observations. UUB regulation of the system states to a neighborhood of the origin, and convergence of the policy to a neighborhood of the optimal policy are established using a Lyapunov-based analysis. Simulations demonstrate that the developed technique approximates the system model and the optimal controller on-line, while maintaining system stability, without the use of a probing signal.

References

[1] L. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, vol. 4, pp. 237–285, 1996.

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

[3] R. Padhi, N. Unnikrishnan, X. Wang, and S. Balakrishnan, "A single network adaptive critic (SNAC) architecture for optimal control synthesis for a class of nonlinear systems," *Neural Netw.*, vol. 19, no. 10, pp. 1648–1660, 2006.

[4] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 38, pp. 943–949, 2008.

[5] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, 2009.

[6] 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.

[7] P. Mehta and S. Meyn, "Q-learning and pontryagin's minimum principle," in *Proc. IEEE Conf. Decis. Control*, Dec. 2009, pp. 3598 –3605.

[8] 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.

[9] 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, 2011.

[10] 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, 2013.

[11] H. Zhang, L. Cui, and Y. Luo, "Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network adp," *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 206–216, 2013.

[12] 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.

[13] P. He and S. Jagannathan, "Reinforcement learning neural-network-based controller for nonlinear discrete-time systems with input constraints," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 37, no. 2, pp. 425–436, 2007.

[14] 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.

[15] K. S. Narendra and A. M. Annaswamy, "A new adaptive law for robust adaptive control without persistent excitation," *IEEE Trans. Autom. Control*, vol. 32, pp. 134–145, 1987.

[16] K. Narendra and A. Annaswamy, *Stable Adaptive Systems*. Prentice-Hall, Inc., 1989.

[17] S. Sastry and M. Bodson, *Adaptive Control: Stability, Convergence, and Robustness*. Upper Saddle River, NJ: Prentice-Hall, 1989.

[18] P. Ioannou and J. Sun, *Robust Adaptive Control*. Prentice Hall, 1996.

[19] D. Vrabie, "Online adaptive optimal control for continuous-time systems," Ph.D. dissertation, University of Texas at Arlington, 2010.

[20] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Model-free q-learning designs for linear discrete-time zero-sum games with application to H_∞ control," *Automatica*, vol. 43, pp. 473–481, 2007.

[21] K. Vamvoudakis and F. Lewis, "Multi-player non-zero-sum games: Online adaptive learning solution of coupled

hamilton-jacobi equations," *Automatica*, vol. 47, pp. 1556–1569, 2011.

[22] K. G. Vamvoudakis, F. L. Lewis, and G. R. Hudas, "Multi-agent differential graphical games: Online adaptive learning solution for synchronization with optimality," *Automatica*, vol. 48, no. 8, pp. 1598 – 1611, 2012.

[23] H. Modares, F. Lewis, and M.-B. Naghibi-Sistani, "Adaptive optimal control of unknown constrained-input systems using policy iteration and neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 10, pp. 1513–1525, 2013.

[24] 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, April 2014.

[25] 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.

[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] R. Kamalapurkar, P. Walters, and W. E. Dixon, "Concurrent learning-based approximate optimal regulation," in *Proc. IEEE Conf. Decis. Control*, Florence, IT, Dec. 2013, pp. 6256–6261.

[28] S. P. Singh, "Reinforcement learning with a hierarchy of abstract models," in *AAAI Natl. Conf. Artif. Intell.*, vol. 92, 1992, pp. 202–207.

[29] C. G. Atkeson and S. Schaal, "Robot learning from demonstration," in *Int. Conf. Mach. Learn.*, vol. 97, 1997, pp. 12–20.

[30] P. Abbeel, M. Quigley, and A. Y. Ng, "Using inaccurate models in reinforcement learning," in *Int. Conf. Mach. Learn.* New York, NY, USA: ACM, 2006, pp. 1–8.

[31] M. P. Deisenroth, *Efficient reinforcement learning using Gaussian processes*. KIT Scientific Publishing, 2010.

[32] D. Mitrovic, S. Klanke, and S. Vijayakumar, "Adaptive optimal feedback control with learned internal dynamics models," in *From Motor Learning to Interaction Learning in Robots*, ser. Studies in Computational Intelligence, O. Sigaud and J. Peters, Eds. Springer Berlin Heidelberg, 2010, vol. 264, pp. 65–84.

[33] M. P. Deisenroth and C. E. Rasmussen, "Pilco: A model-based and data-efficient approach to policy search," in *Int. Conf. Mach. Learn.*, 2011, pp. 465–472.

[34] D. E. Kirk, *Optimal Control Theory: An Introduction*. Dover, 2004.

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

[36] V. Konda and J. Tsitsiklis, "On actor-critic algorithms," *SIAM J. Control Optim.*, vol. 42, no. 4, pp. 1143–1166, 2004.

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

[38] K. Vamvoudakis and F. Lewis, "Online synchronous policy iteration method for optimal control," in *Recent Advances in Intelligent Control Systems*, W. Yu, Ed. Springer, 2009, pp. 357–374.

[39] T. Dierks and S. Jagannathan, "Optimal control of affine nonlinear continuous-time systems," in *Proc. Am. Control Conf.*, 2010, pp. 1568–1573.

[40] W. E. Dixon, A. Behal, D. M. Dawson, and S. Nagarkatti, *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach*. Birkhauser: Boston, 2003.

[41] 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, March 2011.

[42] 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.

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

[44] A. Savitzky and M. J. E. Golay, "Smoothing and differentiation of data by simplified least squares procedures." *Anal. Chem.*, vol. 36, no. 8, pp. 1627–1639, 1964.

[45] S. Bhasin, R. Kamalapurkar, H. T. Dinh, and W. Dixon, "Robust identification-based state derivative estimation for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 1, pp. 187–192, 2013.

[46] D. Garg, W. W. Hager, and A. V. Rao, "Pseudospectral methods for solving infinite-horizon optimal control problems," *Automatica*, vol. 47, no. 4, pp. 829 – 837, 2011.

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

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

A System Identification

A.1 CL-based parameter update

In traditional adaptive control, convergence of the estimates $\hat{\theta}$ to their true values θ is ensured by assuming that a *persistent* excitation condition is satisfied [16–18]. To ensure convergence under a *finite* excitation condition, this result employs a CL-based approach to update the parameter estimates using recorded input-output data [41, 42, 47].

Assumption 4 *[41, 42] A collection $\mathcal{H}_{id}$ of triplets $\{(a_j, b_j, c_j) \mid a_j \in \mathbb{R}^n,\ b_j \in \mathbb{R}^n,\ c_j \in \mathbb{R}^m\}_{j=1}^{M}$ that satisfies*

$$rank\left(\sum_{j=1}^{M} Y^T(a_j)\, Y(a_j)\right) = p,$$
$$\|b_j - f(a_j) + g(a_j)\, c_j\| < \overline{d},\ \forall j, \tag{A.1}$$

is available a priori, where $\overline{d} \in \mathbb{R}_{\geq 0}$ is a positive constant. [10]

To satisfy Assumption 4, data recorded in a previous run of the system can be utilized, or the data stack can be recorded by running the system using a different known stabilizing controller for a finite amount of time until the recorded data satisfies the rank condition (A.1).

In some cases, a data stack may not be available a priori. For such applications, the data stack can be recorded online, i.e., the points a_j and c_j can be recorded along the system trajectory as $a_j = x(t_j)$ and $c_j = u(t_j)$ for some $t_j \in \mathbb{R}_{\geq t_0}$. Provided the system states are exciting over a finite time interval $t \in [t_0, t_0 + \overline{t}]$ (versus $t \in [t_0, \infty)$ as in traditional PE-based approaches) until the data stack satisfies (A.1), then a modified form of the controller developed in Section 3 can be used over the time interval $t \in [t_0, t_0 + \overline{t}]$, and the controller developed in Section 3 can be used thereafter. The required modifications to the controller, and the resulting modifications to the stability analysis are provided in [48, Appendix A].

Based on Assumption 4, the update law for the parameter estimates is designed as

$$\dot{\hat{\theta}} = \frac{\Gamma_\theta k_\theta}{M} \sum_{j=1}^{M} Y^T(a_j) \left(b_j - g(a_j) c_j - Y(a_j) \hat{\theta} \right), \quad \text{(A.2)}$$

where $\Gamma_\theta \in \mathbb{R}^{p \times p}$ is a constant positive definite adaptation gain matrix and $k_\theta \in \mathbb{R}$ is a constant positive CL gain. From (1) and the definition of $\tilde{\theta}$, the bracketed term in (A.2), can be expressed as $b_j - g(a_j) c_j - Y(a_j) \hat{\theta} = Y(a_j) \tilde{\theta} + d_j$, where $d_j \triangleq b_j - f(a_j) + g(a_j) c_j \in \mathbb{R}^n$, and the parameter update law in (A.2) can be expressed in the advantageous form

$$\dot{\hat{\theta}} = \frac{\Gamma_\theta k_\theta}{M} \left(\sum_{j=1}^{M} Y^T(a_j) Y(a_j) \right) \tilde{\theta} + \frac{\Gamma_\theta k_\theta}{M} \sum_{j=1}^{M} Y^T(a_j) d_j. \quad \text{(A.3)}$$

The rate of convergence of the parameter estimates to a neighborhood of their ideal values is directly (and the ultimate bound is inversely) proportional to the minimum singular value of the matrix $\sum_{j=1}^{M} Y^T(a_j) Y(a_j)$; hence, the performance of the estimator can be improved online if a triplet (a_j, b_j, c_j) in $\mathcal{H}_{id}$ is replaced with an updated triplet (a_k, b_k, c_k) that increases the singular value of $\sum_{j=1}^{M} Y^T(a_j) Y(a_j)$. The stability analysis in Section 4 allows for this approach through the use of a singular value maximizing algorithm (cf. [42, 47]).

[10] Since $\theta \in \Theta$, where Θ is a compact set, the assumption that $\overline{d}$ is independent of θ is justified.

A.2 Convergence analysis

Let $V_\theta : \mathbb{R}^{n+p} \to \mathbb{R}_{\geq 0}$ be a positive definite continuously differentiable candidate Lyapunov function defined as

$$V_\theta\left(\tilde{\theta}\right) \triangleq \frac{1}{2} \tilde{\theta}^T \Gamma_\theta^{-1} \tilde{\theta},$$

The following bounds on the Lyapunov function can be established:

$$\frac{\underline{\gamma}}{2} \left\| \tilde{\theta} \right\|^2 \leq V_\theta\left(\tilde{\theta}\right) \leq \frac{\overline{\gamma}}{2} \left\| \tilde{\theta} \right\|^2,$$

where $\underline{\gamma}, \overline{\gamma} \in \mathbb{R}$ denote the minimum and the maximum eigenvalues of the matrix Γ_θ^{-1}. Using (A.3), the Lyapunov derivative can be expressed as

$$\dot{V}_\theta = -\tilde{\theta}^T \frac{k_\theta}{M} \left(\sum_{j=1}^{M} Y^T(a_j) Y(a_j) \right) \tilde{\theta} - \frac{k_\theta}{M} \tilde{\theta}^T \sum_{j=1}^{M} Y^T(a_j) d_j.$$

Let $\underline{y} \in \mathbb{R}$ be the minimum eigenvalue of $\left(\frac{1}{M} \sum_{j=1}^{M} Y^T(a_j) Y(a_j) \right)$. Since $\left(\sum_{j=1}^{M} Y^T(a_j) Y(a_j) \right)$ is symmetric and positive semi-definite, (A.1) can be used to conclude that it is also positive definite, and hence $\underline{y} > 0$. Hence, the Lyapunov derivative can be bounded as [11]

$$\dot{V}_0 \leq -\underline{y} k_\theta \left\| \tilde{\theta} \right\|^2 + k_\theta d_\theta \left\| \tilde{\theta} \right\|,$$

where $d_\theta = \overline{dY}$, $\overline{Y} = \max_{j=1,\cdots,M} (\|Y(a_j)\|)$. Hence, $\left\| \tilde{\theta} \right\|$ exponentially decays to an ultimate bound as $t \to \infty$. The CL-based system identifier satisfies Assumption 2 with $K = \underline{y} k_\theta$ and $D = k_\theta d_\theta$. To satisfy the last inequality in (22), the quantity $\frac{\iota}{v_l}$ needs to be small. Based on the definitions in (21), the quantity $\frac{\iota}{v_l}$ is proportional to $\frac{D^2}{K^2}$, which is proportional to $\frac{d_\theta^2}{\underline{y}^2}$. From the definitions of d_θ and $\underline{y}$,

$$\frac{d_\theta^2}{\underline{y}^2} = \overline{d}^2 \frac{\left(\sum_{j=1}^{M} \|Y(a_j)\| \right)^2}{\left(\lambda_{\min} \left(\sum_{j=1}^{M} Y^T(a_j) Y(a_j) \right) \right)^2}.$$

Thus, in general, a small $\overline{d}$ (i.e., accurate numerical differentiation) is required to obtain the result in Theorem 1.

[11] If $\mathcal{H}_{id}$ is updated with new data, the update law (A.3) forms a switched system. Provided(A.1) holds, and $\mathcal{H}_{id}$ is updated using a singular value maximizing algorithm, V_θ is a common Lyapunov function for the switched system (cf. [42]).