Automatica 74 (2016) 247–258

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Efficient model-based reinforcement learning for approximate online optimal control☆

Rushikesh Kamalapurkar [a], Joel A. Rosenfeld [b], Warren E. Dixon [b]

[a] School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, USA
[b] Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, USA

ARTICLE INFO

Article history:
Received 3 April 2015
Received in revised form
6 May 2016
Accepted 26 July 2016

Keywords:
Model-based reinforcement learning
Data-based control
Adaptive control
Local approximation

ABSTRACT

An infinite horizon optimal regulation problem is solved online for a deterministic control-affine nonlinear dynamical system using a state following (StaF) kernel method to approximate the value function. Unlike traditional methods that aim to approximate a function over a large compact set, the StaF kernel method aims to approximate a function in a small neighborhood of a state that travels within a compact set. Simulation results demonstrate that stability and approximate optimality of the control system can be achieved with significantly fewer basis functions than may be required for global approximation methods.

1. Introduction

Reinforcement learning (RL) has become a popular tool for determining online solutions of optimal control problems for systems with finite state and action spaces (Bertsekas, 2007; Bertsekas & Tsitsiklis, 1996; Konda & Tsitsiklis, 2004; Mehta & Meyn, 2009; Sutton & Barto, 1998; Szepesvári, 2010). Due to various technical and practical difficulties, implementation of RL-based closed-loop controllers on hardware platforms remains a challenge. In recent years, adaptive dynamic programming (ADP) has been successfully used to realize RL in deterministic autonomous control-affine systems to solve optimal control problems via value function approximation (Al-Tamimi, Lewis, & Abu-Khalaf, 2008; Bhasin et al., 2013; Deisenroth & Rasmussen, 2011; Dierks, Thumati, & Jagannathan, 2009; Doya, 2000; Lewis & Vrabie, 2009; Mehta & Meyn, 2009; Padhi, Unnikrishnan, Wang, & Balakrishnan, 2006; Vamvoudakis & Lewis, 2010; Zhang, Cui, & Luo, 2013; Zhang, Cui, Zhang, & Luo, 2011; Zhang, Liu, Luo, & Wang, 2013). ADP techniques employ parametric function approximation methods (typically neural networks (NNs)) to approximate the value function (Heydari & Balakrishnan, 2013; Kiumarsi, Lewis, Modares, Karimpour, & Naghibi-Sistani, 2014; Liu, Huang, Wang, & Wei, 2013; Padhi et al., 2006; Yang, Liu, & Wang, 2014). ADP-based controllers are void of predesigned stabilizing feedback and are completely defined by the estimated parameters. Hence, the error between the optimal and the estimated value function is required to decay to a sufficiently small bound sufficiently fast to establish closed-loop stability. The size of the error bound is determined by the selected basis functions, and the convergence rate is determined by richness of the data used for learning.

Sufficiently accurate approximation of the value function over a sufficiently large neighborhood often requires a large number of basis functions, and hence, introduces a large number of unknown parameters. One way to achieve accurate function approximation with fewer unknown parameters is to use prior knowledge about the system to determine the basis functions. However, for general nonlinear systems, prior knowledge of the features of the optimal value function is generally not available; hence, a large number of generic basis functions is often the only feasible option.

Fast approximation of the value function over a large neighborhood requires sufficiently rich data to be available for learning. In traditional ADP methods such as Bhasin et al. (2013), Vamvoudakis and Lewis (2009) and Vamvoudakis and Lewis (2010), richness of data manifests itself as the amount of excitation in the system. In experience replay-based techniques such as Chowdhary (2010), Chowdhary and Johnson (2011a), Chowdhary, Yucelen, Mühlegg (2013) and Modares, Lewis, and Naghibi-Sistani (2014), richness of data is quantified by eigenvalues of a recorded history stack. In

☆ The material in this paper was partially presented at the 2015 American Control Conference, July 1–3, 2015, Chicago, IL, USA. This paper was recommended for publication in revised form by Associate Editor Shuzhi Sam Ge under the direction of Editor Miroslav Krstic.

E-mail addresses: rushikesh.kamalapurkar@okstate.edu (R. Kamalapurkar), joelar@ufl.edu (J.A. Rosenfeld), wdixon@ufl.edu (W.E. Dixon).

http://dx.doi.org/10.1016/j.automatica.2016.08.004
0005-1098/

model-based RL techniques such as Kamalapurkar, Andrews, Walters, and Dixon (2014), Kamalapurkar, Klotz, and Dixon (2014) and Kamalapurkar, Walters, and Dixon (2013), richness of data corresponds to the eigenvalues of a learning matrix. As the dimension of the system and the number of basis functions increases, the richer data is required to achieve learning. In traditional ADP methods, the demand for rich data is met by adding excitation signals to the controller, thereby causing undesirable oscillations. In experience replay-based ADP methods and in model-based RL, the demand for richer data causes exponential growth in the required data storage. Hence, implementations of traditional ADP techniques such as Al-Tamimi et al. (2008), Bhasin et al. (2013), Dierks et al. (2009), Doya (2000), Lewis and Vrabie (2009), Mehta and Meyn (2009), Padhi et al. (2006), Vamvoudakis and Lewis (2009, 2010), Zhang, Cui et al. (2013); Zhang et al. (2011); Zhang, Liu et al. (2013) and data-driven ADP techniques such as Kamalapurkar, Andrews et al. (2014), Kamalapurkar, Klotz et al. (2014), Kamalapurkar et al. (2013), Luo, Wu, Huang, and Liu (2014), Modares et al. (2014) and Yang, Liu, and Wei (2014) in high dimensional systems are scarcely found in the literature.

In this paper, a novel model-based RL technique is developed to achieve sufficient excitation without causing undesirable oscillations and expenditure of control effort like traditional ADP techniques and at a lower computational cost than state-of-the-art data-driven ADP techniques. Motivated by the fact that the computational effort required to implement ADP and the data-richness required to achieve convergence both decrease with decreasing number of basis functions, this paper focuses on reduction of the number of basis functions used for value function approximation.

A key contribution of this paper and our preliminary work in Kamalapurkar, Rosenfeld, and Dixon (2015) is the observation that online implementation of an ADP-based approximate optimal controller does not require an estimate of the optimal value function over the entire domain of operation of the system. Instead, only an estimate of the slope of the value function evaluated at the current state is required for feedback. Since it is reasonable to postulate that approximation of the value function over a local domain would require fewer basis functions than approximation over the entire domain of operation, this paper focuses on reduction of the size of the approximation domain. Such a reduction is achieved via selection of basis functions that travel with the system state (referred to as state-following (StaF) kernels).

Unlike traditional value function approximation, where the unknown parameters are constants, the unknown parameters corresponding to the StaF kernels are functions of the system state. The Lyapunov-based stability analysis presented in Section 4 is facilitated by the fact that the ideal weights are continuously differentiable functions of the system state. To facilitate the proof of continuous differentiability, the StaF kernels are selected from a Reproducing Kernel Hilbert Space (RKHS). Other function approximation methods, such as radial basis functions, sigmoids, higher order neural networks, support vector machines, etc., can potentially be utilized in a state-following manner to achieve similar results provided continuous differentiability of the ideal weights can be established. An examination of smoothness properties of the ideal weights resulting from a state-following implementation of the aforementioned function approximation methods is out of the scope of this paper.

A key contribution of this paper over our preliminary work in Kamalapurkar et al. (2015) is the observation that model-based RL techniques can be implemented without storing any data if the available model is used to simulate persistent excitation. In other words, an excitation signal added to the simulated system, instead of the actual physical system, can be used to learn the value function. Excitation via simulation is implemented using Bellman error (BE) extrapolation (cf. Kamalapurkar, Andrews et al., 2014, Kamalapurkar, Klotz et al., 2014 and Kamalapurkar et al., 2013); however, instead of a large number of autonomous extrapolation functions employed in results such as Kamalapurkar, Andrews et al. (2014), Kamalapurkar, Klotz et al. (2014) and Kamalapurkar et al. (2013), a single time-varying extrapolation function is selected, where the time-variation of the extrapolation function simulates excitation. The use of a single extrapolation point introduces a technical challenge since the BE extrapolation matrix is rank deficient at each time instance. The aforementioned challenge is addressed in Section 4.3 by modifying the stability analysis to utilize persistent excitation of the extrapolated regressor matrix. Simulation results including comparisons with state-of-the-art model-based RL techniques are presented to demonstrate the effectiveness of the developed technique.

In the following, Section 2 summarizes key results from our preliminary work in Rosenfeld, Kamalapurkar, and Dixon (2015), where the theory of reproducing kernel Hilbert spaces (RKHSs) is used to establish continuous differentiability of the ideal weights with respect to the system state, and the postulate that approximation of a function over a small neighborhood requires fewer basis functions is stated and proved. In Section 3 the StaF-based function approximation approach is used to approximately solve an optimal regulation problem online using exact model knowledge via value function approximation. Section 4 is dedicated to Lyapunov-based stability analysis of the developed technique. Section 5 extends the developed technique to systems with uncertain drift dynamics. Section 6 presents comparative simulation results and Section 7 provides concluding remarks.

2. StaF kernel functions

Let H be a universal RKHS over a compact set $\chi \subset \mathbb{R}^n$ with a continuously differentiable positive definite kernel $k : \chi \times \chi \to \mathbb{R}$. Let $\overline{V}^* : \chi \to \mathbb{R}$ be a function such that $\overline{V}^* \in H$. Let $C \triangleq [c_1, c_2, \ldots, c_L]^T \in \chi^L$ be a set of distinct centers, and let $\sigma : \chi \times \chi^L \to \mathbb{R}^L$ be defined as $\sigma(x, C) = [k(x, c_1), \ldots, k(x, c_L)]^T$. Then, there exists a unique set of weights W_H such that

$$W_H(C) = \arg\min_{a \in \mathbb{R}^L} \left\| a^T \sigma(\cdot, C) - \overline{V}^* \right\|_H,$$

where $\|\cdot\|_H$ denotes the Hilbert space norm. Furthermore, for any given $\epsilon > 0$, there exists a constant $L \in \mathbb{N}$, a set of centers, $C \in \chi^L$, and a set of weights, $W \in \mathbb{R}^L$, such that $\left\| W^T \sigma(\cdot, C) - \overline{V}^* \right\|_H \leq \epsilon$. On compact sets, the Hilbert space norm corresponding to a Hilbert space with continuously differentiable kernels dominates the supremum norm of functions and their derivatives (Steinwart & Christmann, 2008, Corollary 4.36). Hence, the function can be approximated as well as its derivative, that is, there exist centers and weights for which, $\left\| W^T \sigma(\cdot, C) - \overline{V}^* \right\|_{\chi,\infty} < \epsilon$ and $\left\| W^T \nabla\sigma(\cdot, C) - \nabla\overline{V}^* \right\|_{\chi,\infty} < \epsilon$.[1]

Let $B_r(x) \subset \chi$ denote a closed ball of radius r centered at the current state x. Let $H_{x,r}$ denote the restriction of the Hilbert space H to $B_r(x)$. Then, $H_{x,r}$ is a Hilbert space with the restricted kernel $k_{x,r} : B_r(x) \times B_r(x) \to \mathbb{R}$ defined as $k_{x,r}(y, z) = k(y, z), \forall (y, z) \in B_r(x) \times B_r(x)$. The following result, first stated and proved in Rosenfeld et al. (2015) is stated here to motivate the use of StaF kernels.

[1] The notation $\nabla f(x, y, \ldots)$ denotes the partial derivative of f with respect to the first argument and the notation $\|f\|_{A,\infty}$ denotes the supremum of the absolute value (or the pointwise norm, if f is vector-valued) of f over the set A.

Theorem 1 (Rosenfeld et al., 2015). *Let $\epsilon, r > 0$ and let p denote a polynomial that approximates $\overline{V}^*$ within an error ϵ over $B_r(x)$. Let $N(r, x, \epsilon)$ denote the degree of p. Let $k(y, x) = e^{y^T x}$ be the exponential kernel function, which corresponds to a universal RKHS. Then, for each $x \in \chi$, there exists a finite number of centers, $c_1, c_2, \ldots, c_{M(r,x,\epsilon)} \in B_r(x)$ and weights $w_1, w_2, \ldots, w_{M(r,x,\epsilon)}$ such that*

$$\left\| \overline{V}^*(y) - \sum_{i=1}^{M(r,x,\epsilon)} w_i e^{y^T c_i} \right\|_{B_r(x),\infty} < \epsilon,$$

where $M(r, x, \epsilon) < \binom{n+N(r,x,\epsilon)+S(r,x,\epsilon)}{N(r,x,\epsilon)+S(r,x,\epsilon)}$, asymptotically, for some $S(r, x, \epsilon) \in \mathbb{N}$. Moreover, r, $N(r, x, \epsilon)$ and $S(r, x, \epsilon)$ can be bounded uniformly over χ for any fixed ϵ.[2]

The Weierstrass theorem indicates that as r decreases, the degree $N(r, x, \epsilon)$ of the polynomial needed to achieve the same error ϵ over $B_r(x)$ decreases (Lorentz, 1986). Hence, by Theorem 1, approximation of a function over a smaller domain requires a smaller number of exponential kernels. Furthermore, provided the region of interest is small enough, the number of kernels required to approximate continuous functions with arbitrary accuracy can be reduced to $\binom{n+2}{2}$.

In the StaF approach, the centers are selected to follow the current state x, i.e., the locations of the centers are defined as a function of the system state. Since the system state evolves in time, the ideal weights are not constant. To approximate the ideal weights using gradient-based algorithms, it is essential that the weights change smoothly with respect to the system state. The following result, first stated and proved in Rosenfeld et al. (2015) establishes continuity of the ideal weights as a function of the centers.

Theorem 2 (Rosenfeld et al., 2015). *Let the kernel function k be such that the functions $k(\cdot, x)$ are l-times continuously differentiable for all $x \in \chi$. Let $C \triangleq [c_1, c_2, \ldots, c_L]^T$ be a set of distinct centers such that $c_i \in B_r(x)$, $\forall i = 1, \ldots, L$, with associated ideal weights*

$$W_{H_{x,r}}(C) = \arg\min_{a \in R^M} \left\| \sum_{i=1}^{M} a_i k(\cdot, c_i) - V(\cdot) \right\|_{H_{x,r}}. \quad (1)$$

Then, the function $W_{H_{x,r}}$ is l-times continuously differentiable with respect to each component of C.

Theorem 1 motivates the use of StaF kernels for model-based RL, and Theorem 2 facilitates implementation of gradient-based update laws to learn the time-varying ideal weights in real-time.

3. StaF kernel functions for online approximate optimal control

3.1. Problem formulation

Consider a control affine nonlinear dynamical system of the form

$$\dot{x}(t) = f(x(t)) + g(x(t))\, u(t), \quad (2)$$

$t \in \mathbb{R}_{\geq t_0}$,[3] where t_0 denotes the initial time, $x : \mathbb{R}_{\geq t_0} \to \mathbb{R}^n$ denotes the system state $f : \mathbb{R}^n \to \mathbb{R}^n$ and $g : \mathbb{R}^n \to \mathbb{R}^{n \times m}$ denote the drift dynamics and the control effectiveness, respectively, and $u : \mathbb{R}_{\geq t_0} \to \mathbb{R}^m$ denotes the control input. The functions f and g are assumed to be known and locally Lipschitz continuous. Furthermore, $f(\mathbf{0}_{n\times 1}) = \mathbf{0}_{n\times 1}$ and $\nabla f : \mathbb{R}^n \to \mathbb{R}^{n\times n}$ is continuous.[4] In the following, the notation $\phi^u(t; t_0, x_0)$ denotes the trajectory of the system in (2) under the control signal u with the initial condition $x_0 \in \mathbb{R}^n$ and initial time $t_0 \in \mathbb{R}_{\geq 0}$.

Remark. Selection of an optimal regulation problem and the assumption that the system dynamics are known are motivated by ease of exposition. Using a concurrent learning (CL)-based adaptive system identifier and the state augmentation technique developed in Kamalapurkar, Andrews et al. (2014), the technique developed in this paper can be extended to a class of trajectory tracking problems in the presence of uncertainties in the system drift dynamics. For a detailed description of StaF-based online approximate optimal control under uncertainty, see Section 5. Simulation results in Section 6.2 demonstrate the performance of such an extension.

The control objective is to solve the infinite-horizon optimal regulation problem online, i.e., to design a control signal u online to minimize the cost functional

$$J(x, u) \triangleq \int_{t_0}^{\infty} r(x(\tau), u(\tau))\, d\tau, \quad (3)$$

under the dynamic constraint in (2) while regulating the system state to the origin. In (3), $r : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}_{\geq 0}$ denotes the instantaneous cost defined as

$$r(x^o, u^o) \triangleq Q(x^o) + u^{oT} R u^o, \quad (4)$$

for all $x^o \in \mathbb{R}^n$ and $u^o \in \mathbb{R}^m$, where $Q : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ is a positive definite function, and $R \in \mathbb{R}^{m\times m}$ is a constant positive definite symmetric matrix.[5]

3.2. Exact solution

It is well known that since the functions f, g, and Q are stationary (time-invariant) and the time-horizon is infinite, the optimal control input is a stationary state-feedback policy $u(t) = \xi(x(t))$ for some function $\xi : \mathbb{R}^n \to \mathbb{R}^m$. Furthermore, the value function is also a stationary function (Liberzon, 2012, Equation 5.19). Hence, the optimal value function $V^* : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ can be expressed as

$$V^*(x^o) \triangleq \inf_{u(\tau)\in U | \tau \in \mathbb{R}_{\geq t}} \int_t^{\infty} r\left(\phi^u(\tau; t, x^o), u(\tau)\right) d\tau, \quad (5)$$

where $U \subset \mathbb{R}^m$ is the action space. Assuming an optimal controller exists, the optimal value function is characterized by the corresponding Hamilton–Jacobi–Bellman (HJB) equation (Kirk, 2004, Section 3.11)

$$0 = \min_{u^o \in U} \left(\nabla V(x^o)\left(f(x^o) + g(x^o)u^o\right) + r(x^o, u^o)\right), \quad (6)$$

with the boundary condition $V(\mathbf{0}_{n\times 1}) = 0$, where $U \subset \mathbb{R}^m$ denotes the action space. Provided the HJB in (6) admits a continuously differentiable solution, it constitutes a necessary and sufficient condition for optimality (Liberzon, 2012, Section 5.1.4), (Kirk, 2004, Section 3.13). The optimal control policy $u^* : \mathbb{R}^n \to \mathbb{R}^m$ can be determined from (6) as

$$u^*(x^o) \triangleq -\frac{1}{2} R^{-1} g^T(x^o) \left(\nabla V^*(x^o)\right)^T. \quad (7)$$

[2] The notation $\binom{a}{b}$ denotes the combinatorial operation "a choose b".

[3] The notation $\mathbb{R}_{\geq a}$ denotes the interval $[a, \infty)$, and the notation $\mathbb{R}_{>a}$ denotes the interval (a, ∞).

[4] The notation $\mathbf{0}_{n\times m}$ denotes an $n \times m$ matrix of zeros.

[5] In (4) and in the reminder of this paper, the notation $(\cdot)^o \in A$ is used to denote an arbitrary element of the set A.

3.3. Value function approximation

An analytical solution of the HJB equation is generally infeasible; hence, an approximate solution is sought. In an approximate actor-critic-based solution, the optimal value function V^* is approximated using a parametric estimate. The expression for the optimal policy in (7) indicates that, to compute the optimal action when the system is at any given state x^o, one only needs to evaluate the gradient ∇V^* at x^o. Hence, to compute the optimal policy at x^o, one only needs to approximate the value function over a small neighborhood around x^o. Furthermore, as established in Theorem 1, the number of basis functions required to approximate the value function is smaller if the approximation space is smaller (with respect to the ordering induced by set containment). Hence, in this result, the aim is to obtain a uniform approximation of the value function over a small neighborhood around the current system state.

StaF kernels are employed to achieve the aforementioned objective. To facilitate the development, let $\chi \subset \mathbb{R}^n$ be compact and let x^o be in the interior of χ. Then, for all $\epsilon > 0$, there exists a function $\overline{V}^* \in H_{x^o,r}$ such that $\sup_{y^o \in B_r(x^o)} \left| V^*\left(y^o\right) - \overline{V}^*\left(y^o\right) \right| < \epsilon$, where $H_{x^o,r}$ is a restriction of a universal RKHS, H, introduced in Section 2, to $B_r\left(x^o\right)$. In the developed StaF-based method, a small compact set $B_r\left(x^o\right)$ around the current state x^o is selected for value function approximation by selecting the centers $C \in B_r\left(x^o\right)$ such that $C = c\left(x^o\right)$ for some continuously differentiable function $c : \chi \to \chi^L$. Using StaF kernels centered at a point x^o, the value function can be represented as

$$V^*\left(y^o\right) = W\left(x^o\right)^T \sigma\left(y^o, c\left(x^o\right)\right) + \varepsilon\left(x^o, y^o\right),$$

$y^o \in B_r\left(x^o\right)$, where $\varepsilon\left(x^o, y^o\right)$ denotes the function approximation error.

Since the centers of the kernel functions change as the system state changes, the ideal weights also change as the system state changes. The state-dependent nature of the ideal weights differentiates this approach from state-of-the-art ADP methods in the sense that the stability analysis needs to account for changing ideal weights. Based on Theorem 2, it can be established that the ideal weight function $W : \chi \to \mathbb{R}^L$ defined as $W\left(x^o\right) \triangleq W_{H_{x^o,r}}\left(c\left(x^o\right)\right)$, where $W_{H_{x^o,r}}$ was introduced in (1), is continuously differentiable, provided the functions σ and c are continuously differentiable.

The approximate value function $\hat{V} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^L \to \mathbb{R}$ and the approximate policy $\hat{u} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^L \to \mathbb{R}^m$, evaluated at a point $y^o \in B_r\left(x^o\right)$, using StaF kernels centered at x^o, can then be expressed as

$$\hat{V}\left(y^o, x^o, \hat{W}_c\right) \triangleq \hat{W}_c^T \sigma\left(y^o, c\left(x^o\right)\right),$$
$$\hat{u}\left(y^o, x^o, \hat{W}_a\right) \triangleq -\frac{1}{2} R^{-1} g^T\left(y^o\right) \nabla\sigma\left(y^o, c\left(x^o\right)\right)^T \hat{W}_a, \tag{8}$$

where σ denotes the vector of basis functions, introduced in Section 2.

The objective of the critic is to learn the ideal parameters $W\left(x^o\right)$, and the objective of the actor is to implement a stabilizing controller based on the parameters learned by the critic. Motivated by the stability analysis, the actor and the critic maintain separate estimates $\hat{W}_a$ and $\hat{W}_c$, respectively, of the ideal parameters $W\left(x^o\right)$. Using the estimates $\hat{V}$ and $\hat{u}$ for V^* and u^*, respectively, a residual error $\delta : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^L \times \mathbb{R}^L \to \mathbb{R}$, called the BE, is computed as

$$\delta\left(y^o, x^o, \hat{W}_c, \hat{W}_a\right) \triangleq r\left(y^o, \hat{u}\left(y^o, x^o, \hat{W}_a\right)\right) + \nabla\hat{V}\left(y^o, x^o, \hat{W}_c\right)\left(f\left(y^o\right) + g\left(y^o\right)\hat{u}\left(y^o, x^o, \hat{W}_a\right)\right). \tag{9}$$

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 $\delta\left(y^o, x^o, \hat{W}_c, \hat{W}_a\right) = 0, \forall x^o \in \mathbb{R}^n, \forall y^o \in B_r\left(x^o\right)$. Since an exact basis for value function approximation is generally not available, an approximate set of parameters that minimizes the BE is sought.

3.4. Online learning based on simulation of experience

To learn the ideal parameters online, the critic evaluates a form $\delta_t : \mathbb{R}_{\geq t_0} \to \mathbb{R}$ of the BE at each time instance t as

$$\delta_t(t) \triangleq \delta\left(x(t), x(t), \hat{W}_c(t), \hat{W}_a(t)\right), \tag{10}$$

where $\hat{W}_a(t)$ and $\hat{W}_c(t)$ denote the estimates of the actor and the critic weights, respectively, at time t, and the notation $x(t)$ is used to denote the state the system in (2), at time t, when starting from initial time t_0, initial state x_0, and under the feedback controller

$$u(t) = \hat{u}\left(x(t), x(t), \hat{W}_a(t)\right). \tag{11}$$

Since (6) constitutes a necessary and sufficient condition for optimality, the BE serves as an indirect measure of how close the critic parameter estimates $\hat{W}_c$ are to their ideal values; hence, in RL literature, each evaluation of the BE is interpreted as gained experience. Since the BE in (10) is evaluated along the system trajectory, the experience gained is along the system trajectory.

Learning based on simulation of experience is achieved by extrapolating the BE to unexplored areas of the state space. The critic selects a set of functions $\left\{x_i : \mathbb{R}^n \times \mathbb{R}_{\geq t_0} \to \mathbb{R}^n\right\}_{i=1}^N$ such that each x_i maps the current state $x(t)$ to a point $x_i(x(t), t) \in B_r(x(t))$.

The critic then evaluates a form $\delta_{ti} : \mathbb{R}_{\geq t_0} \to \mathbb{R}$ of the BE for each x_i as

$$\delta_{ti}(t) = \delta\left(x_i(x(t), t), x(t), \hat{W}_c(t), \hat{W}_a(t)\right). \tag{12}$$

The critic then uses the BEs from (10) and (12) to improve the estimate $\hat{W}_c(t)$ using the recursive least-squares-based update law

$$\dot{\hat{W}}_c(t) = -k_{c1}\Gamma(t)\frac{\omega(t)}{\rho(t)}\delta_t(t) - \frac{k_{c2}}{N}\Gamma(t)\sum_{i=1}^{N}\frac{\omega_i(t)}{\rho_i(t)}\delta_{ti}(t), \tag{13}$$

where $\rho_i(t) \triangleq \sqrt{1 + \gamma_1 \omega_i^T(t)\omega_i(t)}$, $\rho(t) \triangleq \sqrt{1 + \gamma_1 \omega^T(t)\omega(t)}$, $k_{c1}, k_{c2}, \gamma_1 \in \mathbb{R}_{>0}$ are constant learning gains,

$$\omega(t) \triangleq \nabla\sigma(x(t), c(x(t)))f(x(t)) + \nabla\sigma(x(t), c(x(t)))g(x(t))\hat{u}\left(x(t), x(t), \hat{W}_a(t)\right),$$

and

$$\omega_i(t) \triangleq \nabla\sigma(x_i(x(t)), c(x(t)))f(x_i(x(t), t)) + \nabla\sigma(x_i(x(t)), c(x(t)))g(x_i(x(t), t))\hat{u}\left(x_i(x(t), t), x(t), \hat{W}_a(t)\right).$$

In (13), $\Gamma(t)$ denotes the least-square learning gain matrix updated according to

$$\dot{\Gamma}(t) = \beta\Gamma(t) - k_{c1}\Gamma(t)\frac{\omega(t)\omega^T(t)}{\rho^2(t)}\Gamma(t) - \frac{k_{c2}}{N}\Gamma(t)\sum_{i=1}^{N}\frac{\omega_i(t)\omega_i^T(t)}{\rho_i^2(t)}\Gamma(t), \quad \Gamma(t_0) = \Gamma_0, \tag{14}$$

where $\beta \in \mathbb{R}_{>0}$ is a constant forgetting factor.

Motivated by a Lyapunov-based stability analysis, the update law for the actor is designed as

$$\dot{\hat{W}}_a(t) = -k_{a1}\left(\hat{W}_a(t) - \hat{W}_c(t)\right) - k_{a2}\hat{W}_a(t) + \frac{k_{c1}G_\sigma^T(t)\hat{W}_a(t)\omega(t)^T}{4\rho(t)}\hat{W}_c(t) + \sum_{i=1}^{N}\frac{k_{c2}G_{\sigma i}^T(t)\hat{W}_a(t)\omega_i^T(t)}{4N\rho_i(t)}\hat{W}_c(t), \tag{15}$$

where $k_{a1}, k_{a2} \in \mathbb{R}_{>0}$ are learning gains,

$$G_\sigma(t) \triangleq \nabla\sigma(x(t), c(x(t)))\, g(x(t))\, R^{-1} g^T(x(t)) \cdot \nabla\sigma^T(x(t), c(x(t))),$$

and

$$G_{\sigma i}(t) \triangleq \nabla\sigma(x_i(x(t), t), c(x(t)))\, g(x_i(x(t), t)) \cdot R^{-1} g^T(x_i(x(t), t))\, \nabla\sigma^T(x_i(x(t), t), c(x(t))).$$

4. Analysis

4.1. Computational complexity

The computational cost associated with the implementation of the developed method can be computed to be $O\left(N\left(L^3 + mnL + Lm^2 + n^2 + m^2\right)\right)$. Since local approximation is targeted, the StaF kernels result in a reduction in the number of required basis functions (i.e., L). Since the computational cost has a cubic relationship with the number of basis functions, the StaF methodology results in a significant computational benefit. The computational cost grows linearly with the number of extrapolation points (i.e., N). If the points are selected using grid-based methods employed in results such as Kamalapurkar et al. (2013), the number N increases geometrically with respect to the state dimension, n. On the other hand, if the extrapolation points are selected to be time varying, then as few as a single point can be sufficient, provided the time-trajectory of the point contains enough information to satisfy the subsequent Assumption 1.

In the following, Assumption 1 formalizes the conditions under which the trajectories of the closed-loop system can be shown to be ultimately bounded, and Lemma 1 facilitates the analysis of the closed-loop system when time-varying extrapolation trajectories are utilized.

4.2. Excitation conditions

For notational brevity, time-dependence of all the signals is suppressed hereafter. Let $B_\zeta \subset \mathbb{R}^{n+2L}$ denote a closed ball with radius ζ centered at the origin. Let $\chi \triangleq B_\zeta \cap \mathbb{R}^n$. Let the notation $\overline{\|(\cdot)\|}$ be defined as $\overline{\|h\|} \triangleq \sup_{\xi\in\chi}\|h(\xi)\|$, for some continuous function $h : \mathbb{R}^n \to \mathbb{R}^k$. To facilitate the subsequent stability analysis, the BEs in (10) and (12) are expressed in terms of the weight estimation errors $\tilde{W}_c \triangleq W - \hat{W}_c$ and $\tilde{W}_a = W - \hat{W}_a$ as

$$\delta_t = -\omega^T\tilde{W}_c + \frac{1}{4}\tilde{W}_a G_\sigma \tilde{W}_a + \Delta(x),$$
$$\delta_{ti} = -\omega_i^T\tilde{W}_c + \frac{1}{4}\tilde{W}_a^T G_{\sigma i}\tilde{W}_a + \Delta_i(x), \tag{16}$$

where the functions $\Delta, \Delta_i : \mathbb{R}^n \to \mathbb{R}$ are uniformly bounded over χ such that the bounds $\overline{\|\Delta\|}$ and $\overline{\|\Delta_i\|}$ decrease with decreasing $\overline{\|\nabla\varepsilon\|}$ and $\overline{\|\nabla W\|}$. Let a candidate Lyapunov function $V_L : \mathbb{R}^{n+2L} \times \mathbb{R}_{\geq t_0} \to \mathbb{R}$ be defined as

$$V_L(Z, t) \triangleq V^*(x) + \frac{1}{2}\tilde{W}_c^T\Gamma^{-1}(t)\tilde{W}_c + \frac{1}{2}\tilde{W}_a^T\tilde{W}_a,$$

where V^* is the optimal value function, and

$$Z = \left[x^T, \tilde{W}_c^T, \tilde{W}_a^T\right]^T.$$

To facilitate learning, the system states x and the selected functions x_i are assumed to satisfy the following.

Assumption 1. There exist constants $T \in \mathbb{R}_{>0}$ and $\underline{c}_1, \underline{c}_2, \underline{c}_3 \in \mathbb{R}_{\geq 0}$, such that

$$\underline{c}_1 I_L \leq \int_t^{t+T}\left(\frac{\omega(\tau)\omega^T(\tau)}{\rho^2(\tau)}\right)d\tau, \quad \forall t \in \mathbb{R}_{\geq t_0},$$
$$\underline{c}_2 I_L \leq \inf_{t\in\mathbb{R}_{\geq t_0}}\left(\frac{1}{N}\sum_{i=1}^{N}\frac{\omega_i(t)\omega_i^T(t)}{\rho_i^2(t)}\right),$$
$$\underline{c}_3 I_L \leq \frac{1}{N}\int_t^{t+T}\left(\sum_{i=1}^{N}\frac{\omega_i(\tau)\omega_i^T(\tau)}{\rho_i^2(\tau)}\right)d\tau, \quad \forall t \in \mathbb{R}_{\geq t_0},$$

where, at least one of the constants $\underline{c}_1$, $\underline{c}_2$, and $\underline{c}_3$ is strictly positive.

Unlike typical ADP literature that assumes ω is PE, Assumption 1 only requires either the regressor ω or the regressor ω_i to be persistently exciting. The regressor ω is completely determined by the system state x, and the weights $\hat{W}_a$. Hence, excitation in ω vanishes as the system states and the weights converge. Hence, in general, it is unlikely that $\underline{c}_1 > 0$. However, the regressor ω_i depends on x_i, which can be designed independent of the system state x. Hence, $\underline{c}_3$ can be made strictly positive if the signal x_i contains enough frequencies. The constant $\underline{c}_2$ can be made strictly positive by selecting a sufficient number of extrapolation functions.

Intuitively, selection of a single time-varying BE extrapolation function results in virtual excitation. That is, instead of using input–output data from a persistently excited system, the dynamic model is used to simulate persistent excitation to facilitate parameter convergence. The performance of the developed extrapolation method is demonstrated using comparative simulations in Section 6.3, where it is demonstrated that the developed method using a single time-varying extrapolation point results in improved computational efficiency when compared to a large number of fixed extrapolation functions.

4.3. Boundedness of the least-squares gain under persistent excitation

The following lemma facilitates the stability analysis by establishing upper and lower bounds on the eigenvalues of the least-squares learning gain matrix, Γ.

Lemma 1. *Provided Assumption 1 holds and $\lambda_{\min}\left\{\Gamma_0^{-1}\right\} > 0$, the update law in (14) ensures that the least squares gain matrix satisfies*

$$\underline{\Gamma} I_L \leq \Gamma(t) \leq \overline{\Gamma} I_L, \tag{17}$$

where $\overline{\Gamma} = \frac{1}{\min\left\{k_{c1}\underline{c}_1 + k_{c2}\max\left\{\underline{c}_2 T, \underline{c}_3\right\}, \lambda_{\min}\left\{\Gamma_0^{-1}\right\}\right\}e^{-\beta T}}$ and $\underline{\Gamma} = \frac{1}{\lambda_{\max}\left\{\Gamma_0^{-1}\right\} + \frac{(k_{c1}+k_{c2})}{\beta\gamma_1}}$. Furthermore, $\overline{\Gamma} > 0$.

Proof. The proof closely follows the proof of Corollary 4.3.2 in Ioannou and Sun (1996). The update law in (14) implies that $\frac{d}{dt}\Gamma^{-1}(t) = -\beta\Gamma^{-1}(t) + k_{c1}\frac{\omega(t)\omega^T(t)}{\rho^2(t)} + \frac{k_{c2}}{N}\sum_{i=1}^{N}\frac{\omega_i(t)\omega_i^T(t)}{\rho_i^2(t)}$. Hence,

$$\Gamma^{-1}(t) = e^{-\beta t}\Gamma_0^{-1} + k_{c1}\int_0^t e^{-\beta(t-\tau)}\frac{\omega(\tau)\omega^T(\tau)}{\rho^2(\tau)}d\tau + \frac{k_{c2}}{N}\int_0^t e^{-\beta(t-\tau)}\sum_{i=1}^{N}\frac{\omega_i(\tau)\omega_i^T(\tau)}{\rho_i^2(\tau)}d\tau.$$

To facilitate the proof, let $t < T$. Then,

$$\Gamma^{-1}(t) \geq e^{-\beta t}\Gamma_0^{-1} \geq e^{-\beta T}\Gamma_0^{-1} \geq \lambda_{\min}\left\{\Gamma_0^{-1}\right\} e^{-\beta T} I_L.$$

Since the integrands are positive, it follows that if $t \geq T$, then Γ^{-1} can be bounded as

$$\Gamma^{-1}(t) \geq k_{c1}\int_{t-T}^{t} e^{-\beta(t-\tau)}\frac{\omega(\tau)\omega^T(\tau)}{\rho^2(\tau)}d\tau + \frac{k_{c2}}{N}\int_{t-T}^{t} e^{-\beta(t-\tau)}\sum_{i=1}^{N}\frac{\omega_i(\tau)\omega_i^T(\tau)}{\rho_i^2(\tau)}d\tau.$$

Hence,

$$\Gamma^{-1}(t) \geq k_{c1}e^{-\beta T}\int_{t-T}^{t}\frac{\omega(\tau)\omega^T(\tau)}{\rho^2(\tau)}d\tau + \frac{k_{c2}}{N}e^{-\beta T}\int_{t-T}^{t}\sum_{i=1}^{N}\frac{\omega_i(\tau)\omega_i^T(\tau)}{\rho_i^2(\tau)}d\tau.$$

Using Assumption 1,

$$\frac{1}{N}\int_{t-T}^{t}\sum_{i=1}^{N}\frac{\omega_i(\tau)\omega_i^T(\tau)}{\rho_i^2(\tau)}d\tau \geq \max\left\{\underline{c}_2 T, \underline{c}_3\right\} I_L,$$

$$\int_{t-T}^{t}\frac{\omega(\tau)\omega^T(\tau)}{\rho^2(\tau)}d\tau \geq \underline{c}_1 I_L.$$

Hence a lower bound for Γ^{-1} is obtained as,

$$\Gamma^{-1}(t) \geq \min\left\{k_{c1}\underline{c}_1 + k_{c2}\max\left\{\underline{c}_2 T, \underline{c}_3\right\}, \lambda_{\min}\left\{\Gamma_0^{-1}\right\}\right\} e^{-\beta T} I_L. \tag{18}$$

Provided Assumption 1 holds, the lower bound in (18) is strictly positive. Furthermore, using the facts that $\frac{\omega(t)\omega^T(t)}{\rho^2(t)} \leq \frac{1}{\gamma_1}$ and $\frac{\omega_i(t)\omega_i^T(t)}{\rho_i^2(t)} \leq \frac{1}{\gamma_1}$ for all $t \in \mathbb{R}_{\geq t_0}$,

$$\Gamma^{-1}(t) \leq \int_0^t e^{-\beta(t-\tau)}\left(k_{c1}\frac{1}{\gamma_1} + \frac{k_{c2}}{N}\sum_{i=1}^{N}\frac{1}{\gamma_1}\right) I_L d\tau + e^{-\beta t}\Gamma_0^{-1} \leq \left(\lambda_{\max}\left\{\Gamma_0^{-1}\right\} + \frac{(k_{c1}+k_{c2})}{\beta\gamma_1}\right) I_L.$$

Since the inverse of the lower and upper bounds on Γ^{-1} are the upper and lower bounds on Γ, respectively, the proof is complete.

4.4. Main result

Since the optimal value function is positive definite, (17) and Khalil (2002, Lemma 4.3) can be used to show that the candidate Lyapunov function satisfies the following bounds

$$\underline{v_l}\left(\left\|Z^o\right\|\right) \leq V_L\left(Z^o, t\right) \leq \overline{v_l}\left(\left\|Z^o\right\|\right), \tag{19}$$

for all $t \in \mathbb{R}_{\geq t_0}$ and for all $Z^o \in \mathbb{R}^{2+2L}$. In (19), $\underline{v_l}, \overline{v_l} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class $\mathcal{K}$ functions. To facilitate the analysis, let $\underline{c} \in \mathbb{R}_{>0}$ be a constant defined as

$$\underline{c} \triangleq \frac{\beta}{2\overline{\Gamma}k_{c2}} + \frac{\underline{c}_2}{2}, \tag{20}$$

and let $\iota \in \mathbb{R}_{>0}$ be a constant defined as

$$\iota \triangleq \frac{3\left(\frac{(k_{c1}+k_{c2})\overline{\|\Delta\|}}{\sqrt{\nu}} + \frac{\overline{\|\nabla W f\|}}{\underline{\Gamma}} + \frac{\overline{\left\|\Gamma^{-1}G_{W\sigma}W\right\|}}{2}\right)^2}{4k_{c2}\underline{c}} + \frac{1}{(k_{a1}+k_{a2})}\left(\frac{\overline{\|G_{W\sigma}W\|} + \overline{\|G_{V\sigma}\|}}{2} + k_{a2}\overline{\|W\|} + \overline{\|\nabla W f\|} + \frac{(k_{c1}+k_{c2})\,\overline{\|G_\sigma\|\|W\|}^2}{4\sqrt{\nu}}\right)^2 + \frac{1}{2}\overline{\|G_{VW\sigma}\|} + \frac{1}{2}\overline{\|G_{V\varepsilon}\|},$$

where $G_{W\sigma} \triangleq \nabla W G \nabla\sigma^T$, $G_{V\sigma} \triangleq \nabla V^* G \nabla\sigma^T$, $G_{VW} \triangleq \nabla V^* G \nabla W^T$, and $G_{V\epsilon} \triangleq \nabla V^* G \nabla\epsilon^T$. Let $v_l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a class $\mathcal{K}$ function such that

$$v_l(\|Z\|) \leq \frac{Q(x)}{2} + \frac{k_{c2}\underline{c}}{6}\left\|\tilde{W}_c\right\|^2 + \frac{(k_{a1}+k_{a2})}{8}\left\|\tilde{W}_a\right\|^2.$$

The sufficient conditions for the subsequent Lyapunov-based stability analysis are given by

$$\frac{k_{c2}\underline{c}}{3} \geq \frac{\left(\frac{\overline{\|G_{W\sigma}\|}}{2\underline{\Gamma}} + \frac{(k_{c1}+k_{c2})\overline{\|W^T G_\sigma\|}}{4\sqrt{\nu}} + k_{a1}\right)^2}{(k_{a1}+k_{a2})}, \tag{21}$$

$$\frac{(k_{a1}+k_{a2})}{4} \geq \left(\frac{\overline{\|G_{W\sigma}\|}}{2} + \frac{(k_{c1}+k_{c2})\overline{\|W\|\|G_\sigma\|}}{4\sqrt{\nu}}\right), \tag{22}$$

$$v_l^{-1}(\iota) < \overline{v_l}^{-1}\left(\underline{v_l}(\zeta)\right). \tag{23}$$

The sufficient condition in (21) can be satisfied provided the points for BE extrapolation are selected such that the minimum eigenvalue $\underline{c}$, introduced in (20) is large enough. The sufficient condition in (22) can be satisfied without affecting (21) by increasing the gain k_{a2}. The sufficient condition in (23) can be satisfied provided $\underline{c}$, k_{a2}, and the state penalty $Q(x)$ are selected to be sufficiently large and the StaF kernels for value function approximation are selected such that $\overline{\|\nabla W\|}$, $\overline{\|\varepsilon\|}$, and $\overline{\|\nabla\varepsilon\|}$ are sufficiently small.[6] To improve computational efficiency, the size of the domain around the current state where the StaF kernels provide good approximation of the value function is desired to be small. Smaller approximation domain results in almost identical extrapolated points, which in turn, results in smaller $\underline{c}$. Hence, the approximation domain cannot be selected to be arbitrarily small and needs to be large enough to meet the sufficient conditions in (21)–(23).

Theorem 3. *Provided Assumption 1 holds and the sufficient gain conditions in (21)–(23) are satisfied, the controller in (11) and the update laws in (13)–(15) ensure that the state x and the weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$ are ultimately bounded.*

Proof. The time-derivative of the candidate Lyapunov function is given by

$$\dot{V}_L = \dot{V}^* + \tilde{W}_c^T\Gamma^{-1}\left(\dot{W} - \dot{\hat{W}}_c\right) + \frac{1}{2}\tilde{W}_c^T\dot{\Gamma}^{-1}\tilde{W}_c + \tilde{W}_a^T\left(\dot{W} - \dot{\hat{W}}_a\right).$$

[6] Similar to NN-based approximation methods such as Al-Tamimi et al. (2008), Dierks et al. (2009), Doya (2000), Lewis and Vrabie (2009), Mehta and Meyn (2009), Padhi et al. (2006), Vamvoudakis and Lewis (2010) and Zhang et al. (2011), the function approximation error, ε, is unknown, and in general, infeasible to compute for a given function, since the ideal NN weights are unknown. Since a bound on ε is unavailable, the gain conditions in (21)–(23) cannot be formally verified. However, they can be met using trial and error by increasing the gain k_{a2}, the number of StaF basis functions, and $\underline{c}$, by selecting more points to extrapolate the bellman error.

Using Theorem 2, the time derivative of the ideal weights can be expressed as

$$\dot{W} = \nabla W(x)\left(f(x) + g(x)u\right). \tag{24}$$

Using (13)–(16) and (24), the time derivative of the candidate Lyapunov function is expressed as

$$\begin{aligned}\dot{V}_L =\ & \nabla V^*(x)\left(f(x)+g(x)u\right)\\ &+\tilde{W}_c^T\Gamma^{-1}\nabla W(x)\left(f(x)+g(x)u\right)\\ &-\tilde{W}_c^T\Gamma^{-1}\left(-k_{c1}\Gamma\frac{\omega}{\rho}\left(-\omega^T\tilde{W}_c+\frac{1}{4}\tilde{W}_aG_\sigma\tilde{W}_a+\Delta(x)\right)\right)\\ &-\tilde{W}_c^T\Gamma^{-1}\left(-\frac{k_{c2}}{N}\Gamma\sum_{i=1}^{N}\frac{\omega_i}{\rho_i}\frac{1}{4}\tilde{W}_a^TG_{\sigma i}\tilde{W}_a\right)\\ &-\tilde{W}_c^T\Gamma^{-1}\left(-\frac{k_{c2}}{N}\Gamma\sum_{i=1}^{N}\frac{\omega_i}{\rho_i}\left(-\omega_i^T\tilde{W}_c+\Delta_i(x)\right)\right)\\ &-\frac{1}{2}\tilde{W}_c^T\Gamma^{-1}\left(\beta\Gamma-k_{c1}\Gamma\frac{\omega\omega^T}{\rho}\Gamma\right)\Gamma^{-1}\tilde{W}_c\\ &-\frac{1}{2}\tilde{W}_c^T\Gamma^{-1}\left(-\frac{k_{c2}}{N}\Gamma\sum_{i=1}^{N}\frac{\omega_i\omega_i^T}{\rho_i}\Gamma\right)\Gamma^{-1}\tilde{W}_c\\ &+\tilde{W}_a^T\left(\nabla W(x)\left(f(x)+g(x)u\right)-\dot{\hat{W}}_a\right).\end{aligned}$$

Provided the sufficient conditions in (21)–(23) hold, the time derivative of the candidate Lyapunov function can be bounded as

$$\dot{V}_L \le -v_l(\|Z\|), \quad \forall \zeta > \|Z\| > v_l^{-1}(\iota). \tag{25}$$

Using (19), (23), and (25), Khalil (2002, Theorem 4.18) can be invoked to conclude that Z is ultimately bounded, in the sense that

$$\limsup_{t\to\infty}\|Z(t)\| \le \underline{v_l}^{-1}\left(\overline{v_l}\left(v_l^{-1}(\iota)\right)\right).$$

Since $Z \in \mathcal{L}_\infty$, x, $\tilde{W}_a$, and $\tilde{W}_c \in \mathcal{L}_\infty$. Since $x \in \mathcal{L}_\infty$ and since W is a continuous function of x, $W \circ x \in \mathcal{L}_\infty$. Hence, $\hat{W}_a$ and $\hat{W}_c \in \infty$, which implies $u \in \mathcal{L}_\infty$.

5. Extension to systems with uncertain drift dynamics

If the drift dynamics are uncertain, a parametric approximation of the dynamics can be employed for BE extrapolation. On any compact set $\mathcal{C} \subset \mathbb{R}^n$ the function f can be represented using a NN as $f(x^o) = \theta^T\sigma_f\left(Y^Tx_1(x^o)\right) + \varepsilon_\theta(x)$, where $x_1(x^o) \triangleq \begin{bmatrix}1 & x^{oT}\end{bmatrix}^T \in \mathbb{R}^{n+1}$, $\theta \in \mathbb{R}^{p+1\times n}$ 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, $\varepsilon_\theta : \mathbb{R}^n \to \mathbb{R}^n$ 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 (cf. Sadegh, 1993), there exist constant ideal weights θ and known constants $\overline{\theta}$, $\overline{\varepsilon_\theta}$, and $\overline{\varepsilon'_\theta} \in \mathbb{R}$ such that $\|\theta\| \le \overline{\theta} < \infty$, $\sup_{x^o\in\mathcal{C}}\|\varepsilon_\theta(x^o)\| \le \overline{\varepsilon_\theta}$, and $\sup_{x^o\in\mathcal{C}}\|\nabla_{x^o}\varepsilon_\theta(x^o)\| \le \overline{\varepsilon'_\theta}$. 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 $\hat{f} : \mathbb{R}^n \times \mathbb{R}^{p+1\times n} \to \mathbb{R}^n$ defined as $\hat{f}\left(x^o, \hat{\theta}\right) \triangleq \hat{\theta}^T\sigma_\theta(x^o)$, where $\sigma_\theta : \mathbb{R}^n \to \mathbb{R}^{p+1}$ is defined as $\sigma_\theta(x^o) = \sigma_f\left(Y^T\begin{bmatrix}1 & x^{oT}\end{bmatrix}^T\right)$. Using $\hat{f}$, the BE in (9) can be approximated by $\hat{\delta} : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^L \times \mathbb{R}^{p+1\times n} \to \mathbb{R}^n$ as

$$\begin{aligned}\hat{\delta}\left(y^o, x^o, \hat{W}_c, \hat{W}_a, \hat{\theta}\right) \triangleq\ & r\left(y^o, \hat{u}\left(y^o, x^o, \hat{W}_a\right)\right)\\ &+\nabla\hat{V}\left(y^o, x^o, \hat{W}_c\right)\left(\hat{f}\left(y^o, \hat{\theta}\right)+g\left(y^o\right)\hat{u}\left(y^o, x^o, \hat{W}_a\right)\right).\end{aligned} \tag{26}$$

Using $\hat{\delta}$, the instantaneous BEs in (10) and (12) are redefined as

$$\delta_t(t) \triangleq \hat{\delta}\left(x(t), x(t), \hat{W}_c(t), \hat{W}_a(t), \hat{\theta}(t)\right), \tag{27}$$

and

$$\delta_{ti}(t) = \hat{\delta}\left(x_i(x(t),t), x(t), \hat{W}_c(t), \hat{W}_a(t), \hat{\theta}(t)\right), \tag{28}$$

respectively, where ω and ω_i are redefined as

$$\begin{aligned}\omega(t) \triangleq\ & \nabla\sigma(x(t), c(x(t)))\hat{f}\left(x(t), \hat{\theta}(t)\right)\\ &+\nabla\sigma(x(t), c(x(t)))g(x(t))\hat{u}\left(x(t), x(t), \hat{W}_a(t)\right),\end{aligned} \tag{29}$$

and

$$\begin{aligned}\omega_i(t) \triangleq\ & \nabla\sigma(x_i(x(t),t), c(x(t)))f\left(x_i(x(t),t), \hat{\theta}(t)\right)\\ &+\nabla\sigma(x_i(x(t),t), c(x(t)))g(x_i(x(t),t))\\ &\cdot\hat{u}\left(x_i(x(t),t), x(t), \hat{W}_a(t)\right).\end{aligned} \tag{30}$$

The following assumption describes the characteristic of a parameter estimator required to achieve closed-loop stability.

Assumption 2 (*Kamalapurkar, Walters, & Dixon, 2016*)**.** A compact set $\Theta \subset \mathbb{R}^p$ that contains the unknown parameter vector θ is known a priori. The estimates $\hat{\theta} : \mathbb{R}_{\ge t_0} \to \mathbb{R}^p$ are updated based on a switched update law of the form

$$\dot{\hat{\theta}}(t) = f_{\theta s}\left(\hat{\theta}(t), t\right), \tag{31}$$

$\hat{\theta}(t_0) = \hat{\theta}_0 \in \Theta$, where $s \in \mathbb{N}$ denotes the switching index and $\left\{f_{\theta s} : \mathbb{R}^{p+1\times n} \times \mathbb{R}_{\ge 0} \to \mathbb{R}^{p+1\times n}\right\}_{s\in\mathbb{N}}$ denotes a family of continuously differentiable functions. There exists a continuously differentiable function $V_\theta : \mathbb{R}^{p+1\times n} \times \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ that satisfies

$$\underline{v}_\theta\left(\left\|\tilde{\theta}^o\right\|\right) \le V_\theta\left(\tilde{\theta}^o, t\right) \le \overline{v}_\theta\left(\left\|\tilde{\theta}^o\right\|\right), \tag{32}$$

$$\begin{aligned}&\nabla V_\theta\left(\tilde{\theta}^o, t\right)\left(-f_{\theta s}\left(\theta-\tilde{\theta}^o, t\right)\right)+\frac{\partial V_\theta\left(\tilde{\theta}^o, t\right)}{\partial t}\\ &\le -K\left\|\tilde{\theta}^o\right\|^2+D\left\|\tilde{\theta}^o\right\|,\end{aligned} \tag{33}$$

for all $s \in \mathbb{N}$, $t \in \mathbb{R}_{\ge t_0}$, and $\tilde{\theta}^o \in \mathbb{R}^{p+1\times n}$, where $\underline{v}_\theta$, $\overline{v}_\theta : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ are class $\mathcal{K}$ functions, $K \in \mathbb{R}_{>0}$ is an adjustable parameter, and $D \in \mathbb{R}_{>0}$ is a positive constant (possibly dependent on K). Furthermore, the ratio $\frac{D}{K}$ is sufficiently small.

Assumption 2 implies that the function V_θ can be used as a candidate Lyapunov function to establish convergence of the parameter estimation error, $\tilde{\theta}$, to a neighborhood of the origin. The function $V_\theta + V_L$ is used as a candidate Lyapunov function to prove Theorem 4. CL (cf. Chowdhary, 2010, Chowdhary & Johnson, 2011b, and Chowdhary et al., 2013) can be used to design parameter estimators that satisfy Assumption 2. Examples of CL-based parameter estimators that satisfy Assumption 2 are available in Kamalapurkar (2014, Section 6.2) for nonlinearly parameterized uncertainty and Kamalapurkar et al. (2016, Appendix A) for linearly parameterized uncertainty. The main result for uncertain drift dynamics is summarized in the following theorem.

Theorem 4. *Provided a parameter estimator that satisfies Assumption 2 is available, the StaF kernels and the basis functions for system identification are selected such that ∇W and the approximation errors ε, $\nabla\varepsilon$, ε_θ and $\nabla\varepsilon_\theta$ are sufficiently small, and provided the points for BE extrapolation are selected such that the minimum eigenvalue $\underline{c}$, introduced in (20) is sufficiently large, then the update laws given by (13)–(15), with the renewed definitions in (26)–(30) ensure that the state x and the weight estimation errors $\tilde{\theta}$, $\tilde{W}_c$, and $\tilde{W}_a$ are ultimately bounded.*

Proof. The proof is a trivial combination of the proof of Theorem 3 and Kamalapurkar et al. (2016, Theorem 1), and hence, is omitted.

6. Simulation

6.1. Optimal regulation problem with exact model knowledge

6.1.1. Simulation parameters

To demonstrate the effectiveness of the StaF kernels, simulations are performed on a two-state nonlinear dynamical system. The system dynamics are given by (2), where $x^o = [x_1^o, x_2^o]^T$,

$$f\left(x^o\right) = \begin{bmatrix} -x_1^o + x_2^o \\ -\frac{1}{2}x_1^o - \frac{1}{2}x_2^o\left(\cos\left(2x_1^o\right)+2\right)^2 \end{bmatrix},$$

$$g\left(x^o\right) = \begin{bmatrix} 0 \\ \cos\left(2x_1^o\right)+2 \end{bmatrix}. \tag{34}$$

The control objective is to minimize the cost

$$\int_0^\infty \left(x^T(\tau)\,x(\tau) + u^2(\tau)\right)d\tau. \tag{35}$$

The system in (34) and the cost in (35) are selected because the corresponding optimal control problem has a known analytical solution. The optimal value function is $V^*(x^o) = \frac{1}{2}x_1^{o2} + x_2^{o2}$, and the optimal control policy is $u^*(x^o) = -(\cos(2x_1^o)+2)x_2^o$ (cf. Vamvoudakis & Lewis, 2010).

To apply the developed technique to this problem, the value function is approximated using three exponential StaF kernels, i.e., $\sigma(x^o, C) = [\sigma_1(x^o, c_1), \sigma_2(x^o, c_2), \sigma_3(x^o, c_3)]^T$. The kernels are selected to be $\sigma_i(x^o, c_i) = e^{x^{oT}c_i} - 1$, $i = 1, \ldots, 3$. The centers c_i are selected to be on the vertices of a shrinking equilateral triangle around the current state, i.e., $c_i = x^o + d_i(x^o)$, $i = 1, \ldots, 3$, where $d_1(x^o) = 0.7\nu^o(x^o)\cdot[0, 1]^T$, $d_2(x^o) = 0.7\nu^o(x^o)\cdot[0.87, -0.5]^T$, and $d_3(x^o) = 0.7\nu^o(x^o)\cdot[-0.87, -0.5]^T$, and $\nu^o(x^o) \triangleq \left(\frac{x^{oT}x^o+0.01}{1+\gamma_2 x^{oT}x^o}\right)$ denotes the shrinking function, where $\gamma_2 \in \mathbb{R}_{>0}$ is a constant normalization gain. To ensure sufficient excitation, a single point for BE extrapolation is selected at random from a uniform distribution over a $2.1\nu^o(x(t)) \times 2.1\nu^o(x(t))$ square centered at the current state $x(t)$ so that the function x_i is of the form $x_i(x^o, t) = x^o + a_i(t)$ for some $a_i(t) \in \mathbb{R}^2$.[7]

The system is initialized at $t_0 = 0$ and the initial conditions[8]

$$x(0) = [-1, 1]^T, \qquad \hat{W}_c(0) = 0.4 \times \mathbf{1}_{3\times 1},$$
$$\Gamma(0) = 500 I_3, \qquad \hat{W}_a(0) = 0.7\hat{W}_c(0),$$

and the learning gains are selected as

$$k_{c1} = 0.001, \quad k_{c2} = 0.25, \quad k_{a1} = 1.2, \quad k_{a2} = 0.01,$$
$$\beta = 0.003, \quad \gamma_1 = 0.05, \quad \gamma_2 = 1.$$

[7] For a general problem with an n-dimensional state, exponential kernels can be utilized with the centers placed at the vertices of an n-dimensional simplex with the current state as the centroid. The extrapolation point can be sampled at each iteration from a uniform distribution over an n-dimensional hypercube centered at the current state.

[8] The notation I_n denotes an $n \times n$ identity matrix and the notation $\mathbf{1}_{n\times m}$ denotes an $n \times m$ matrices of ones.

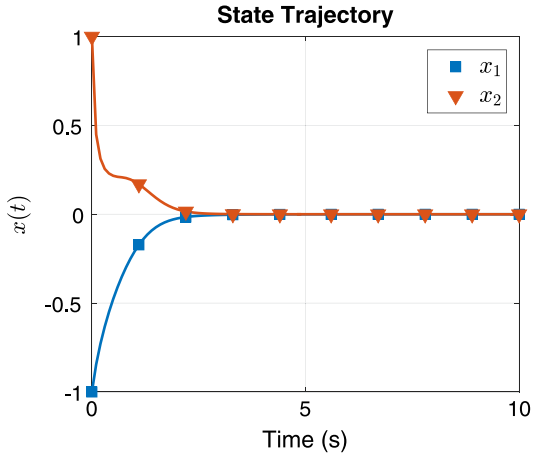

Fig. 1. State trajectories generated using StaF kernel-based ADP.

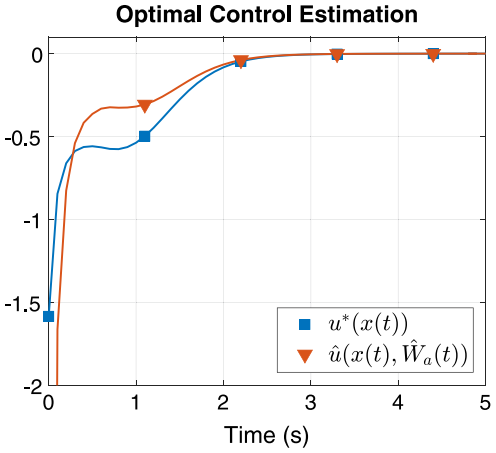

Fig. 2. Control trajectory generated using StaF kernel-based ADP compared with the optimal control trajectory.

6.1.2. Results

Fig. 1 shows that the developed StaF-based controller drives the system states to the origin while maintaining system stability. Fig. 2 shows the implemented control signal compared with the optimal control signal. It is clear that the implemented control converges to the optimal controller. Figs. 3 and 4 show that the weight estimates for the StaF-based value function and policy approximation remain bounded and converge as the state converges to the origin. Since the ideal values of the weights are unknown, the weights cannot directly be compared with their ideal values. However, since the optimal solution is known, the value function estimate corresponding to the weights in Fig. 3 can be compared to the optimal value function at each time t. Fig. 5 shows that the error between the optimal and the estimated value functions rapidly decays to zero.

6.2. Optimal tracking problem with parametric uncertainties in the drift dynamics

6.2.1. Simulation parameters

This simulation demonstrates the effectiveness of the extension developed in Section 5. The drift dynamics in the two-state nonlinear dynamical system in (34) are assumed to be linearly parameterized as

$$f\left(x^o\right) = \underbrace{\begin{bmatrix}\theta_1 & \theta_2 & \theta_3 \\ \theta_4 & \theta_5 & \theta_6\end{bmatrix}}_{\theta^T} \underbrace{\begin{bmatrix} x_1^o \\ x_2^o \\ x_2^o\left(\cos\left(2x_1^o\right)+2\right)\end{bmatrix}}_{\sigma_\theta(x^o)},$$

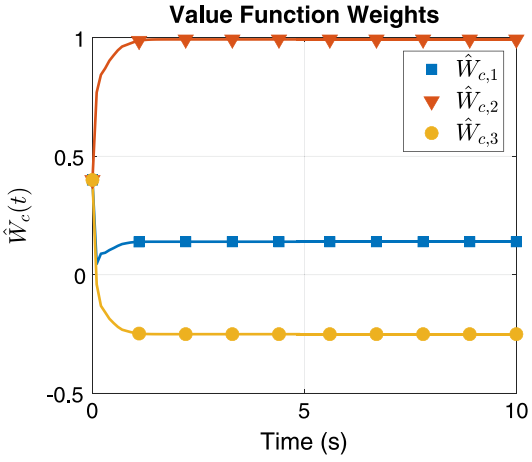

Fig. 3. Trajectories of the estimates of the unknown parameters in the value function generated using StaF kernel-based ADP. The ideal weights are unknown and time-varying; hence, the obtained weights cannot be compared with their ideal weights.

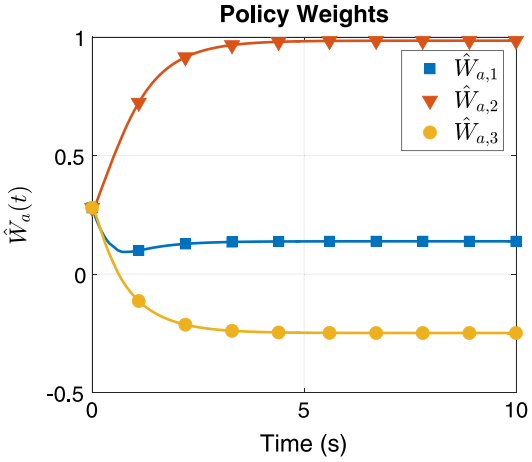

Fig. 4. Trajectories of the estimates of the unknown parameters in the policy generated using StaF kernel-based ADP. The ideal weights are unknown and time-varying; hence, the obtained weights cannot be compared with their ideal weights.

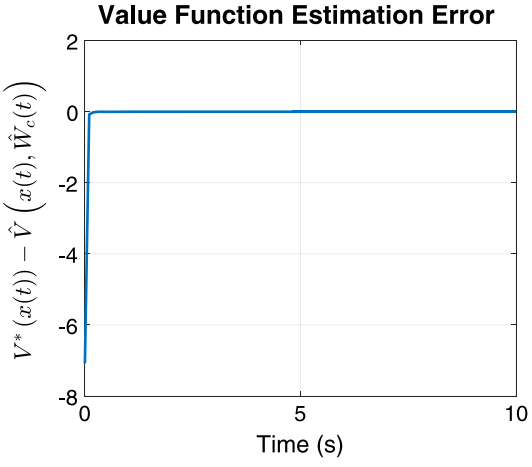

Fig. 5. The error between the optimal and the estimated value function.

where $\theta \in \mathbb{R}^{3\times 2}$ is the matrix of unknown parameters and σ_θ is the known vector of basis functions. The ideal values of the unknown parameters are $\theta_1 = -1$, $\theta_2 = 1$, $\theta_3 = 0$, $\theta_4 = -0.5$, $\theta_5 = 0$, and $\theta_6 = -0.5$. Let $\hat{\theta}$ denote an estimate of the unknown matrix θ. The control objective is to drive the estimate $\hat{\theta}$ to the ideal matrix θ, and to drive the state x to follow a desired trajectory x_d. The desired trajectory is selected to be solution of the initial value problem

$$\dot{x}_d(t) = \begin{bmatrix} -1 & 1 \\ -2 & 1 \end{bmatrix} x_d(t), \quad x_d(0) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \tag{36}$$

and the cost functional is selected to be $\int_0^\infty \left(e^T(t)\,\mathrm{diag}(10, 10)\,e(t) + (\mu(t))^2\right) dt$, where $e(t) = x(t) - x_d(t)$, μ is an auxiliary controller designed using the developed method, and the tracking controller is designed as

$$u(t) = g^+(x_d(t)) \left(\begin{bmatrix} -1 & 1 \\ -2 & 1 \end{bmatrix} x_d(t) - \hat{\theta}^T \sigma_\theta(x_d(t)) \right) + \mu(t),$$

where $g^+(x^o)$ denotes the pseudoinverse of $g(x^o)$.

The value function is a function of the concatenated state $\zeta \triangleq \begin{bmatrix} e^T & x_d^T \end{bmatrix}^T \in \mathbb{R}^4$. The value function is approximated using five exponential StaF kernels given by $\sigma_i(\zeta^o, C)$, where the five centers are selected according to $c_i = \zeta^o + d_i(\zeta^o)$ to form a regular five dimensional simplex around the current state with $\nu^o(\zeta^o) \equiv 1$. Learning gains for system identification and value function approximation are selected as

$$k_{c1} = 0.001, \quad k_{c2} = 2, \quad k_{a1} = 2, \quad k_{a2} = 0.001,$$
$$\beta = 0.01, \quad \gamma_1 = 0.1, \quad \gamma_2 = 1, \quad k = 500,$$
$$\Gamma_\theta = I_3, \quad \Gamma(0) = 50 I_5, \quad k_\theta = 20.$$

Sufficient excitation is ensured by selecting a single state trajectory $\zeta_i(\zeta^o, t) \triangleq \zeta^o + a_i(t)$ for BE extrapolation, where $a_i(t)$ is sampled at each t from a uniform distribution over the $2.1 \times 2.1 \times 2.1 \times 2.1$ hypercube centered at the origin. The history stack required for CL contains ten points, and is recorded online using a singular value maximizing algorithm (cf. Chowdhary et al., 2013), and the required state derivatives are computed using a fifth order Savitzky–Golay smoothing filter (cf. Savitzky & Golay, 1964).

The initial values for the state and the state estimate are selected to be $x(0) = [0, 0]^T$ and $\hat{x}(0) = [0, 0]^T$, respectively. The initial values for the NN weights for the value function, the policy, and the drift dynamics are selected to be $0.025 \times \mathbf{1}_{5\times 1}$, $0.025 \times \mathbf{1}_{5\times 1}$, and $\mathbf{0}_{3\times 2}$, respectively. Since the system in (34) has no stable equilibria, the initial policy $\hat{\mu}(\zeta, \mathbf{0}_{3\times 2})$ is not stabilizing. The stabilization demonstrated in Fig. 6 is achieved via fast simultaneous learning of the system dynamics and the value function.

6.2.2. Results

Figs. 6 and 7 demonstrate that the controller remains bounded and the tracking error is regulated to the origin. The NN weights are functions of the system state ζ. Since ζ converges to a periodic orbit, the NN weights also converge to a periodic orbit (within the bounds of the excitation introduced by the BE extrapolation signal), as demonstrated in Figs. 8 and 9. Fig. 10 demonstrates that the unknown parameters in the drift dynamics, represented by solid lines, converge to their ideal values, represented by dashed lines.

6.3. Comparison

The developed technique is compared with the model-based RL method developed in Kamalapurkar et al. (2013) for regulation and Kamalapurkar, Andrews et al. (2014) for tracking, respectively. The simulations are performed in MATLAB® Simulink® at 1000 Hz on the same machine. The simulations run for 100 s of simulated time. Since the objective is to compare computational efficiency of the model-based RL method, exact knowledge of the system model is used. Table 1 shows that the developed controller requires significantly fewer computational resources than the controllers from Kamalapurkar, Andrews et al. (2014) and Kamalapurkar

Table 1
Simulation results for 2, 3 and 4 dimensional nonlinear systems.

Problem description	Regulation (2-state system)		Regulation (3-state system)		Tracking (4-state system)	
Controller	StaF	Controller in Kamalapurkar et al. (2013)	StaF	Controller in Kamalapurkar et al. (2013)	StaF	Controller in Kamalapurkar, Andrews et al. (2014)
Running time (s)	6.5	17	9.5	62	12	260
Total cost	2.8	1.8	9.3	12.3	3.9	3.4
RMS steady-state error	2.5×10^{-6}	0	4.3×10^{-6}	4.5×10^{-6}	3.5×10^{-4}	2.5×10^{-4}

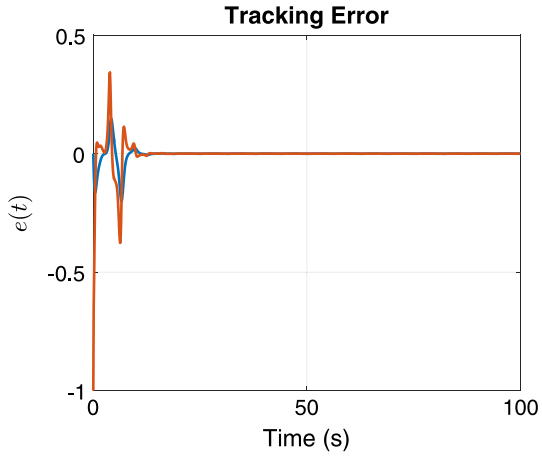

Fig. 6. Tracking error trajectories generated using the proposed method for the trajectory tracking problem.

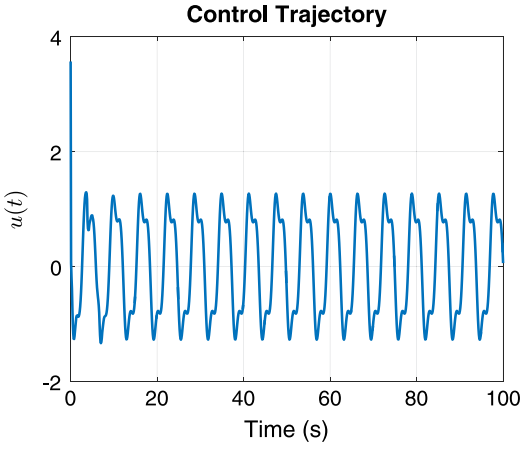

Fig. 7. Control signal generated using the proposed method for the trajectory tracking problem.

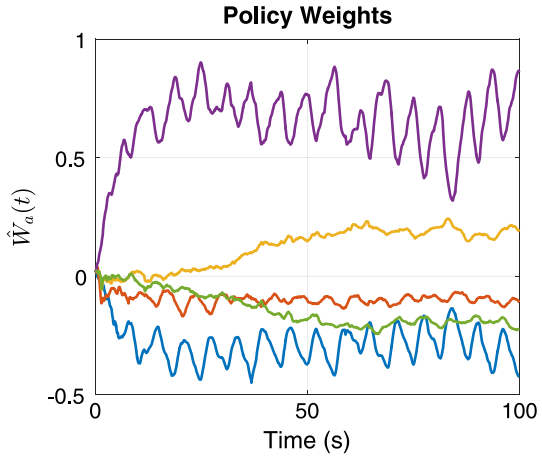

Fig. 8. Policy weight trajectories generated using the proposed method for the trajectory tracking problem. The weights do not converge to a steady-state value because the ideal weights are functions of the time-varying system state. Since an analytical solution of the optimal tracking problem is not available, weights cannot be compared against their ideal values.

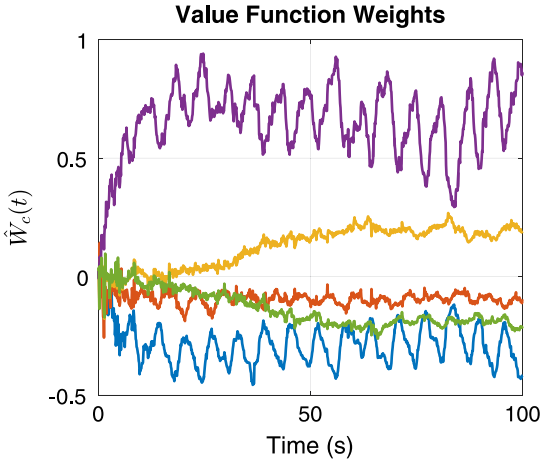

Fig. 9. Value function weight trajectories generated using the proposed method for the trajectory tracking problem. The weights do not converge to a steady-state value because the ideal weights are functions of the time-varying system state. Since an analytical solution of the optimal tracking problem is not available, weights cannot be compared against their ideal values.

et al. (2013). Furthermore, as the system dimension increases, the developed controller significantly outperforms the controllers from Kamalapurkar, Andrews et al. (2014) and Kamalapurkar et al. (2013) in terms of computational efficiency.

Since the optimal solution for the regulation problem is known to be quadratic, the model-based RL method from Kamalapurkar et al. (2013) is implemented using three quadratic basis functions. Since the basis used is exact, the method from Kamalapurkar et al. (2013) yields a smaller steady-state error than the developed method, which uses three inexact, but generic StaF kernels. For the 3-state regulation problem and the tracking problem, the methods from Kamalapurkar, Andrews et al. (2014) and Kamalapurkar et al. (2013) are implemented using polynomial basis functions selected based on a trial-and-error approach. The developed technique is implemented using generic StaF kernels. In this case, since the optimal solution is unknown, both the methods use inexact basis functions, resulting in similar steady-state errors.

The two main advantages of StaF kernels are that they are universal, in the sense that they can be used to approximate a large class of value functions, and that they target local approximation, resulting in a smaller number of required basis functions. However, the StaF kernels trade optimality for universality and computational efficiency. The kernels are inexact, and the weight estimates need to be continually adjusted based on the system trajectory. Hence, as shown in Table 1, the developed technique results in a higher total cost than state-of-the-art model-based RL techniques.

7. Conclusion

In this paper an infinite horizon optimal control problem is solved using a new approximation methodology called the StaF

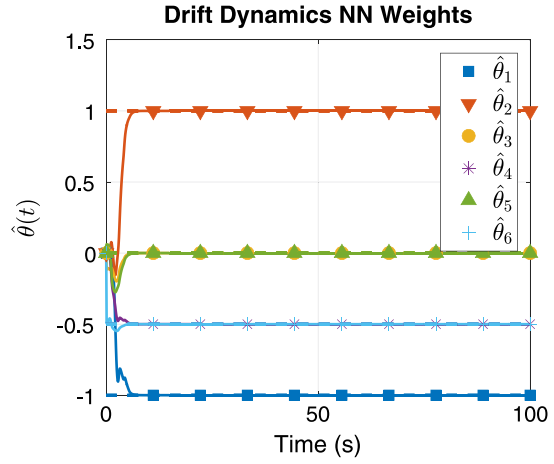

Fig. 10. Trajectories of the unknown parameters in the system drift dynamics for the trajectory tracking problem. The dotted lines represent the true values of the parameters.

kernel method. Motivated by the fact that a smaller number of basis functions is required to approximate functions on smaller domains, the StaF kernel method aims to maintain good approximation of the value function over a small neighborhood of the current state. Computational efficiency of model-based RL is improved by allowing selection of fewer time-varying extrapolation trajectories instead of a large number of autonomous extrapolation functions. Simulation results are presented that solve the infinite horizon optimal regulation and tracking problems online for a two state system using only three and five basis functions, respectively, via the StaF kernel method.

State-of-the-art solutions to solve infinite horizon optimal control problems online aim to approximate the value function over the entire operating domain. Since the approximate optimal policy is completely determined by the value function estimate, state-of-the-art solutions generate, often at an intractable computational cost, policies that are valid over the entire state space. Since the StaF kernel method aims at maintaining local approximation of the value function around the current system state, the StaF kernel method lacks memory, in the sense that the information about the ideal weights over a region of interest is lost when the state leaves the region of interest. Thus, unlike existing techniques, the StaF method trades global optimality for computational efficiency to generate a policy that is near-optimal only over a small neighborhood of the origin. A memory-based modification to the StaF technique that retains and reuses past information is a subject for future research. The control design in (8) exploits the fact that given a basis σ for approximation of the value function, the basis $\frac{1}{2}R^{-1}g^T\nabla\sigma^T$ approximates the optimal controller, provided the dynamics control-affine. As a part of future research, possible extensions to nonaffine systems could potentially be explored by approximating the controller using an independent basis (cf. Bian, Jiang, & Jiang, 2014, Ge & Zhang, 2003, Kiumarsi, Kang, & Lewis, 2016, Liu et al., 2013, Song, Wei, & Xiao, 2016, Wang, Liu, Wei, Zhao, & Jin, 2012, Yang, Liu, Wei, & Wang, 2015 and Zhang, Zhang, Sun, & Luo, 2012).

Acknowledgments

This research is supported in part by NSF award numbers 1161260 and 1217908, ONR grant numbers N00014-13-1-0151 and N00014-16-1-2091, and a contract with the AFRL Mathematical Modeling and Optimization Institute. 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.

References

Al-Tamimi, A., Lewis, F. L., & Abu-Khalaf, M. (2008). Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *38*, 943–949.

Bertsekas, D. (2007). *Dynamic programming and optimal control, Vol. 2* (3rd ed.). Belmont, MA: Athena Scientific.

Bertsekas, D., & Tsitsiklis, J. (1996). *Neuro-dynamic programming*. Athena Scientific.

Bhasin, S., Kamalapurkar, R., Johnson, M., Vamvoudakis, K. G., Lewis, F. L., & Dixon, W. E. (2013). A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems. *Automatica*, *49*(1), 89–92.

Bian, T., Jiang, Y., & Jiang, Z.-P. (2014). Adaptive dynamic programming and optimal control of nonlinear nonaffine systems. *Automatica*, *50*(10), 2624–2632.

Chowdhary, G. (2010). *Concurrent learning for convergence in adaptive control without persistency of excitation*. (Ph.D. dissertation), Georgia Institute of Technology.

Chowdhary, G., & Johnson, E. (2011a). A singular value maximizing data recording algorithm for concurrent learning. In *Proc. American control conf.* (pp. 3547–3552).

Chowdhary, G. V., & Johnson, E. N. (2011b). Theory and flight-test validation of a concurrent-learning adaptive controller. *Journal of Guidance, Control, and Dynamics*, *34*(2), 592–607.

Chowdhary, G., Yucelen, T., Mühlegg, M., & Johnson, E. N. (2013). Concurrent learning adaptive control of linear systems with exponentially convergent bounds. *International Journal of Adaptive Control and Signal Processing*, *27*(4), 280–301.

Deisenroth, M.P., & Rasmussen, C.E. (2011). Pilco: A model-based and data-efficient approach to policy search. In *Int. conf. mach. learn.* (pp. 465–472).

Dierks, T., Thumati, B., & Jagannathan, S. (2009). Optimal control of unknown affine nonlinear discrete-time systems using offline-trained neural networks with proof of convergence. *Neural Networks*, *22*(5–6), 851–860.

Doya, K. (2000). Reinforcement learning in continuous time and space. *Neural Computation*, *12*(1), 219–245.

Ge, S. S., & Zhang, J. (2003). Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback. *IEEE Transactions on Neural Networks*, *14*(4), 900–918.

Heydari, A., & Balakrishnan, S. (2013). Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics. *IEEE Transactions on Neural Networks and Learning Systems*, *24*(1), 145–157.

Ioannou, P., & Sun, J. (1996). *Robust adaptive control*. Prentice Hall.

Kamalapurkar, R. (2014). *Model-based reinforcement learning for online approximate optimal control*. (Ph.D. dissertation), University of Florida.

Kamalapurkar, R., Andrews, L., Walters, P., & Dixon, W.E. (2014). Model-based reinforcement learning for infinite-horizon approximate optimal tracking. In *Proc. IEEE conf. decis. control, Los Angeles, CA, Dec.* (pp. 5083–5088).

Kamalapurkar, R., Klotz, J., & Dixon, W. E. (2014). Concurrent learning-based online approximate feedback Nash equilibrium solution of N-player nonzero-sum differential games. *IEEE/CAA Journal of Automatica Sinica*, *1*(3), 239–247.

Kamalapurkar, R., Rosenfeld, J.A., & Dixon, W.E. (2015). State following (StaF) Kernel functions for function approximation Part II: Adaptive dynamic programming. In *Proc. Am. control conf.* (pp. 521–526).

Kamalapurkar, R., Walters, P., & Dixon, W.E. (2013). Concurrent learning-based approximate optimal regulation. In *Proc. IEEE conf. decis. control, Florence,IT, Dec.* (pp. 6256–6261).

Kamalapurkar, R., Walters, P., & Dixon, W. E. (2016). Model-based reinforcement learning for approximate optimal regulation. *Automatica*, *64*, 94–104.

Khalil, H. K. (2002). *Nonlinear systems* (3rd ed.). Upper Saddle River, NJ: Prentice Hall.

Kirk, D. (2004). *Optimal control theory: An introduction*. Mineola, NY: Dover.

Kiumarsi, B., Kang, W., & Lewis, F. L. (2016). H-∞ control of nonaffine aerial systems using off-policy reinforcement learning. *Unmanned Systems*, *4*(1), 1–10.

Kiumarsi, B., Lewis, F. L., Modares, H., Karimpour, A., & Naghibi-Sistani, M.-B. (2014). Reinforcement Q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics. *Automatica*, *50*(4), 1167–1175.

Konda, V., & Tsitsiklis, J. (2004). On actor-critic algorithms. *SIAM Journal on Control and Optimization*, *42*(4), 1143–1166.

Lewis, F. L., & Vrabie, D. (2009). Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circuits and Systems Magazine*, *9*(3), 32–50.

Liberzon, D. (2012). *Calculus of variations and optimal control theory: a concise introduction*. Princeton University Press.

Liu, D., Huang, Y., Wang, D., & Wei, Q. (2013). Neural-network-observer-based optimal control for unknown nonlinear systems using adaptive dynamic programming. *International Journal of Control*, *86*(9), 1554–1566.

Lorentz, G. G. (1986). *Bernstein polynomials* (2nd ed.). New York: Chelsea Publishing Co..

Luo, B., Wu, H.-N., Huang, T., & Liu, D. (2014). Data-based approximate policy iteration for affine nonlinear continuous-time optimal control design. *Automatica*,.

Mehta, P., & Meyn, S. (2009). Q-learning and pontryagin's minimum principle. In *Proc. IEEE conf. decis. control Dec.* (pp. 3598–3605).

Modares, H., Lewis, F. L., & Naghibi-Sistani, M.-B. (2014). Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems. *Automatica*, *50*(1), 193–202.
Padhi, R., Unnikrishnan, N., Wang, X., & Balakrishnan, S. (2006). A single network adaptive critic (SNAC) architecture for optimal control synthesis for a class of nonlinear systems. *Neural Networks*, *19*(10), 1648–1660.
Rosenfeld, J.A., Kamalapurkar, R., & Dixon, W.E. (2015). State following (StaF) Kernel functions for function approximation Part I: Theory and motivation. In *Proc. Am. control conf.* (see also arXiv: 1503.04854), (pp. 1217–1222).
Sadegh, N. (1993). A perceptron network for functional identification and control of nonlinear systems. *IEEE Transactions on Neural Networks*, *4*(6), 982–988.
Savitzky, A., & Golay, M. J. E. (1964). Smoothing and differentiation of data by simplified least squares procedures. *Analytical Chemistry*, *36*(8), 1627–1639.
Song, R., Wei, Q., & Xiao, W. (2016). Off-policy neuro-optimal control for unknown complex-valued nonlinear systems based on policy iteration. *Neural Computing and Applications*, *46*(1), 85–95.
Steinwart, I., & Christmann, A. (2008). *Information science and statistics*, *Support vector machines*. New York: Springer.
Sutton, R. S., & Barto, A. G. (1998). *Reinforcement learning: An introduction*. Cambridge, MA, USA: MIT Press.
Szepesvári, C. (2010). *Synthesis lectures on artificial intelligence and machine learning*, *Algorithms for reinforcement learning*. Morgan & Claypool Publishers.
Vamvoudakis, K., & Lewis, F. (2009). Online synchronous policy iteration method for optimal control. In W. Yu (Ed.), *Recent advances in intelligent control systems* (pp. 357–374). Springer.
Vamvoudakis, K., & Lewis, F. (2010). Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, *46*(5), 878–888.
Wang, D., Liu, D., Wei, Q., Zhao, D., & Jin, N. (2012). Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming. *Automatica*, *48*(8), 1825–1832.
Yang, X., Liu, D., & Wang, D. (2014). Reinforcement learning for adaptive optimal control of unknown continuous-time nonlinear systems with input constraints. *International Journal of Control*, *87*(3), 553–566.
Yang, X., Liu, D., & Wei, Q. (2014). Online approximate optimal control for affine non-linear systems with unknown internal dynamics using adaptive dynamic programming. *IET Control Theory & Applications*, *8*(16), 1676–1688.
Yang, X., Liu, D., Wei, Q., & Wang, D. (2015). Direct adaptive control for a class of discrete-time unknown nonaffine nonlinear systems using neural networks. *International Journal of Robust and Nonlinear Control*, *25*(12), 1844–1861.
Zhang, H., Cui, L., & Luo, Y. (2013). Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network adp. *IEEE Transactions Cybernetics*, *43*(1), 206–216.
Zhang, H., Cui, L., Zhang, X., & Luo, Y. (2011). Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method. *IEEE Transactions on Neural Networks*, *22*(12), 2226–2236.
Zhang, H., Liu, D., Luo, Y., & Wang, D. (2013). *Communications and control engineering*, *Adaptive dynamic programming for control algorithms and stability*. London: Springer-Verlag.
Zhang, X., Zhang, H., Sun, Q., & Luo, Y. (2012). Adaptive dynamic programming-based optimal control of unknown nonaffine nonlinear discrete-time systems with proof of convergence. *Neurocomputing*, *91*, 48–55.

Rushikesh Kamalapurkar received his M.S. and his Ph.D. degrees in 2011 and 2014, respectively, from the Mechanical and Aerospace Engineering Department at the University of Florida. After working for an year as a postdoctoral research fellow with Dr. Warren E. Dixon, he was selected as the 2015–16 MAE postdoctoral teaching fellow at the University of Florida. In 2016 he joined the School of Mechanical and Aerospace Engineering at the Oklahoma State University as an Assistant professor. His primary research interest has been intelligent, learning-based control of uncertain nonlinear dynamic systems. He has published 2 chapters, 14 journal papers and 18 conference papers. His work has been recognized by the 2015 University of Florida Department of Mechanical and Aerospace Engineering Best Dissertation Award, and the 2014 University of Florida Department of Mechanical and Aerospace Engineering Outstanding Graduate Research Award.

Joel A. Rosenfeld received his Ph.D. in Mathematics at the University of Florida in 2013, under the advisement of Dr. Michael T. Jury. He joined the Nonlinear Controls and Robotics group in 2013 as a postdoctoral researcher in the department of Mechanical Engineering at the University of Florida focusing on approximation problems in control theory.

Warren E. Dixon received his Ph.D. in 2000 from the Department of Electrical and Computer Engineering from Clemson University. He was selected as a Eugene P. Wigner Fellow at Oak Ridge National Laboratory (ORNL). In 2004, he joined the University of Florida in the Mechanical and Aerospace Engineering Department. His main research interest has been the development and application of Lyapunov-based control techniques for uncertain nonlinear systems. He has published 3 books, an edited collection, 12 chapters, and over 120 journal and 220 conference papers. His work has been recognized by the 2015 & 2009 American Automatic Control Council (AACC) O. Hugo Schuck (Best Paper) Award, the 2013 Fred Ellersick Award for Best Overall MILCOM Paper, a 2012–2013 University of Florida College of Engineering Doctoral Dissertation Mentoring Award, the 2011 American Society of Mechanical Engineers (ASME) Dynamics Systems and Control Division Outstanding Young Investigator Award, the 2006 IEEE Robotics and Automation Society (RAS) Early Academic Career Award, an NSF CAREER Award (2006–2011), the 2004 Department of Energy Outstanding Mentor Award, and the 2001 ORNL Early Career Award for Engineering Achievement. He is an ASME & IEEE Fellow, an IEEE Control Systems Society (CSS) Distinguished Lecturer, and served as the Director of Operations for the Executive Committee of the IEEE CSS Board of Governors (2012–2015). He currently serves as a member of the US Air Force Science Advisory Board. He is currently or formerly an associate editor for ASME Journal of Dynamic Systems, Measurement and Control, Automatica, IEEE Control Systems Magazine, IEEE Transactions on Systems Man and Cybernetics: Part B Cybernetics, and the International Journal of Robust and Nonlinear Control.