A Safety Aware Model-Based Reinforcement Learning Framework for Systems with Uncertainties

S M Nahid Mahmud[1] Katrine Hareland[1] Scott A Nivison[2] Zachary I. Bell[2] Rushikesh Kamalapurkar[1]

***Abstract*— Safety awareness is critical in reinforcement learning when task restarts are not available and/or when the system is safety critical. Safety requirements are often expressed in terms of state and/or control constraints. In the past, model-based reinforcement learning approaches combined with barrier transformations have been used as an effective tool to learn the optimal control policy under state constraints for systems with fully known models. In this paper, a reinforcement learning technique is developed that utilizes a novel filtered concurrent learning method to realize simultaneous learning and control in the presence of model uncertainties for safety critical systems.**

I. Introduction

Deployment of unmanned systems in complex, high-risk tasks provides significant operational benefits. However, unmanned systems need the ability to simultaneously synthesize and execute control policies online and in real time, to improve robustness to parametric uncertainties. Over the past few decades, reinforcement learning (RL) has been established as an effective tool for optimal online policy synthesis for both known and uncertain dynamical systems with a finite number of states and action sequences [1], [2].

RL typically requires a large number of iterations due to sample inefficiency. Sample efficiency in RL can be improved using model-based reinforcement learning (MBRL); however, MBRL methods are prone to failure due to inaccurate models. Online MBRL methods that handle modeling uncertainties are motivated by tasks that require systems to operate in dynamic environments with changing objectives, and accurate models of the system and environment are generally not available in complex tasks due to sparsity of data. In the past, MBRL techniques under the umbrella of approximate dynamic programming (ADP) have been successfully utilized to solve reinforcement learning problems online with model uncertainty. ADP utilizes parametric methods such as neural networks (NNs) to approximate the value function, and the system model online. By obtaining an approximation of both the value function and the system model, a stable closed loop adaptive control policy can be developed [3]–[10]. In addition to closed-loop stability, safety is also a significant consideration for online reinforcement learning problems. Real-world optimal control applications typically include constraints on states or inputs that are critical for safety [11]. ADP was successfully extended to address input constrained control problems in [12], [13]. The idea of transforming a state and input constrained nonlinear optimal control problem into an unconstrained one with a type of saturation function was introduced in [14], [15]. Recently, [16] applied ADP to both input and state constrained optimal control problems. The state constrained optimal control problem was transformed, using a barrier transformation (BT), into an equivalent, unconstrained optimization problem.

A MBRL approach to address the state-constrained optimal control problem appeared in [17], where the results in [16] are extended to soften the restrictive persistence of excitation requirement. However, the methods developed in [16] and [17] require fully known models, which are often difficult to obtain. In this paper, a novel filtered concurrent learning technique for parameter estimation is developed and integrated with the barrier transformation method to yield a novel MBRL technique for online state-constrained optimal feedback control. The developed MBRL method realizes simultaneous learning and control in the presence of parametric uncertainties for safety critical systems. The inclusion of filtered concurrent learning makes the feedback controller robust to modeling errors and guarantees closed-loop stability under a *finite* (as opposed to *persistent*) excitation condition. A Lypaunov-based analysis proves the developed MBRL technique is stable and guarantees safety requirements are satisfied. Simulation results are provided to demonstrate the performance of the developed MBRL approach compared to an existing optimal control method.

II. Problem Formulation

A. Control objective

Consider a continuous-time affine nonlinear dynamical system

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

where $x = [x_1; \ldots; x_n] \in \mathbb{R}^n$ with $i = 1, 2, \ldots, n$ is the system state, $\theta \in \mathbb{R}^p$ are the unknown parameters, $u \in \mathbb{R}^q$ is the control input, and the functions $f : \mathbb{R}^n \to \mathbb{R}^{n\times p}$ and $g : \mathbb{R}^n \to \mathbb{R}^{n\times q}$ are known, locally Lipschitz functions with $f(x) = [f_1(x); \cdots; f_n(x)]$ and $g(x) = [g_1(x); \cdots; g_n(x)]$. The notation $[a; b]$ denotes the vector $[a\ b]^T$.

The objective is to design a controller u for the system in (1) such that starting from a given feasible initial condition x^0, the trajectories $x(\cdot)$ decay to the origin and satisfy $x_i(t) \in (a_i, A_i), \forall t \geq 0$, where $a_i < 0 < A_i$. While MBRL

*This research was supported, in part, by the Air Force Research Laboratories under award number FA8651-19-2-0009. Any opinions, findings, or recommendations in this article are those of the author(s), and do not necessarily reflect the views of the sponsoring agencies.

[1]School of Mechanical and Aerospace Engineering, Oklahoma State University, email: {nahid.mahmud, katrine.hareland, rushikesh.kamalapurkar} @okstate.edu.

[2] Air Force Research Laboratories, email: {scott.nivison, zachary.bell.10} @us.af.mil.

methods such as those detailed in [10] guarantee stability of the closed-loop, state constraints are typically difficult to establish without extensive trial and error. Based on Lemma 1, a BT is used in the following to guarantee state constraints.

B. Barrier Transformation

The function $b : \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is referred to as Barrier Function (BF) if

$$b(z, a, A) := \log \frac{A(a - z)}{a(A - z)}, \quad z \in \mathbb{R}. \tag{2}$$

Consider the BF based state transformation

$$s_i := b(x_i, a_i, A_i), \quad x_i = b^{-1}(s_i, a_i, A_i). \tag{3}$$

In the following derivation, whenever clear from the context, the arguments a_i and A_i of the BF and its inverse are suppressed for brevity. The time derivative of the transformed state can then be computed using the chain rule as $\dot{s}_i = \dot{x}_i / \frac{\partial b^{-1}(z, a_i, A_i)}{\partial z}|_{z=s_i}$ which yields the transformed dynamics $\dot{s}_i = (f_i(x)\theta + g_i(x)u) / \frac{db^{-1}(z, a_i, A_i)}{dz}|_{z=s_i} = F_i(s)\theta + G_i(s)u$, $\forall i = 1, 2, \ldots, n,$, where

$$F_i(s) = \frac{a_i^2 e^{s_i} - 2a_i A_i + A_i^2 e^{-s_i}}{A_i a_i^2 - a_i A_i^2} f_i\big([b^{-1}(s_1); \ldots; b^{-1}(s_n)]\big), \tag{4}$$

$$G_i(s) = \frac{a_i^2 e^{s_i} - 2a_i A_i + A_i^2 e^{-s_i}}{A_i a_i^2 - a_i A_i^2} g_i\big([b^{-1}(s_1); \ldots; b^{-1}(s_n)]\big). \tag{5}$$

After using the BT, the dynamics of the transformed state $s = [s_1; \ldots; s_n]$ can be expressed as,

$$\dot{s} = F(s) + G(s)u = y(s)\theta + G(s)u, \tag{6}$$

where $y(s) := [F_1; \ldots; F_n] \in \mathbb{R}^{n \times p}$, and $G(s) := [G_1; \ldots; G_n] \in \mathbb{R}^{n \times q}$.

Continuous differentiability of b^{-1} implies that that F and G are locally Lipschitz continuous. Furthermore, $f(0) = 0$ along with the fact that $b^{-1}(0) = 0$ implies that $F(0) = 0$. As a result, for all compact sets $\Omega \subset \mathbb{R}^n$ containing the origin, G is bounded on Ω and there exists a positive constant a_f such that $\forall s \in \Omega$, $\|F(s)\| \leq a_f \|s\|$. The following lemma relates the solutions of the original system to the solutions of the transformed system.

Lemma 1. If $t \mapsto \Phi(t, b(x_0), \zeta)$ is a Carathéodory solution to (6) starting from the initial condition $b(x_0)$ under the feedback policy $(s, t) \mapsto \zeta(s, t)$ and if $t \mapsto \Lambda(t, x_0, \zeta)$ is a solution to (1) starting from the initial condition x_0 under the feedback policy $(x, t) \mapsto \zeta(b(x), t)$, then $\Lambda(t, x_0, \zeta) = b^{-1}(\Phi(t, b(x_0), \zeta))$ for almost all t.

Proof. See [18, Lemma 1]. □

It is immediate from the Lemma 1 that if the trajectories of (6) are bounded and decay to a neighborhood of the origin under a feedback policy $(s, t) \mapsto \zeta(s, t)$ then the feedback policy $(x, t) \mapsto \zeta(b(x), t)$, when applied to the original system in (1) achieves the control objective stated in Section II-A. To ensure the BT MBRL method is robust to modeling uncertainties, the following section develops a system identifier inspired by the filtered concurrent learning (FCL) method presented in [19].

III. Parameter Estimation

Estimates of the unknown parameters, $\hat{\theta} \in \mathbb{R}^p$ are generated using the filter

$$\dot{Y} = \begin{cases} y(s), & \|Y\| \leq \overline{Y} \\ 0, & \text{otherwise} \end{cases}, \quad Y(0) = 0, \tag{7}$$

$$\dot{Y}_f = \begin{cases} Y^T Y, & \|Y_f\| \leq \overline{Y_f} \\ 0, & \text{otherwise} \end{cases}, \quad Y_f(0) = 0, \tag{8}$$

$$\dot{G}_f = \begin{cases} G(s)u, & \|Y_f\| \leq \overline{Y_f} \\ 0, & \text{otherwise} \end{cases}, \quad G_f(0) = 0, \tag{9}$$

$$\dot{X}_f = \begin{cases} Y^T(s - s^0 - G_f), & \|Y_f\| \leq \overline{Y_f} \\ 0, & \text{otherwise} \end{cases}, \quad X_f(0) = 0, \tag{10}$$

where $s^0 = \left[b\left(x_1^0\right); \ldots; b\left(x_n^0\right)\right]$, and the update law

$$\dot{\hat{\theta}} = \beta_1 Y_f^T(t)(X_f(t) - Y_f(t)\hat{\theta}), \quad \hat{\theta}(0) = \theta^0, \tag{11}$$

where β_1 is a symmetric positive definite gain matrix and $\overline{Y_f}$ is a tunable upper bound on the filtered regressor Y_f.

Note that (8), expressed in the integral form

$$Y_f(t) = \int_0^{t_2} Y^T(\tau) Y(\tau) d\tau, \quad \forall t \geq 0, \tag{12}$$

where $t_2 := \inf_t \{t \geq 0 \mid \|Y_f(t)\| \leq \overline{Y_f}\}$, along with (10), expressed in the integral form

$$X_f(t) = \int_0^{t_2} Y^T(\tau)\big(s(\tau) - s^0 - G_f(\tau)\big) d\tau, \quad \forall t \geq 0, \tag{13}$$

and the fact that $s(\tau) - s^0 - G_f(\tau) = Y(\tau)\theta$, can be used to conclude that $X_f(t) = Y_f(t)\theta$, for all $t \geq 0$. As a result, a measure for the parameter estimation error can be obtained using known quantities as $Y_f \tilde{\theta} = X_f - Y_f \hat{\theta}$, where $\tilde{\theta} := \theta - \hat{\theta}$. The dynamics of the parameter estimation error can then be expressed as

$$\dot{\tilde{\theta}} = -\beta_1 Y_f^T Y_f \tilde{\theta}. \tag{14}$$

The filter design is thus motivated by the fact that if the matrix $Y_f^T Y_f$ is positive definite, uniformly in t, then the Lyapunov function $V_1(\tilde{\theta}) = \frac{1}{2} \tilde{\theta}^T \beta_1^{-1} \tilde{\theta}$ can be used to establish convergence of the parameter estimation error to the origin. Initially, $Y_f^T Y_f$ is a matrix of zeros. To ensure that there exists some finite time T such that $Y_f(t)^T Y_f(t)$ is positive definite, uniformly in t for all $t \geq T$, the following *finite* excitation condition is imposed.

Assumption 1. There exists a time instance $T > 0$ such that $Y_f(T)$ is full rank.

For $t_3 \leq t_4 \leq t_2$, $Y_f(t_4) = Y_f(t_3) + \int_{t_3}^{t_4} Y^T(\tau) Y(\tau) d\tau$ Since $Y_f(t_3)$ is positive semidefinite, and so is the integral, $\lambda_{\min}(Y_f(t_4)) \geq \lambda_{\min}(Y_f(t_4))$[1]. As a result, $t \mapsto \lambda_{\min}(Y_f(t))$ is non-decreasing. Therefore, if Assumption 1 is satisfied at $t = T$, then $Y_f(t)$ is also full rank for all $t \geq T$. Similar to other MBRL methods that rely on system

[1] $\lambda_{\min}$ denotes the minimum eigenvalue of a matrix.

identification (cf. [10, Chapter 4]) the following assumption is needed to ensure boundedness of the state trajectories over the interval $[0, T]$.

Assumption 2. A feedback controller $\psi : \mathbb{R}^n \to \mathbb{R}^q$ that keeps the trajectories of (6) inside a known bounded set over the interval $[0, T)$, without requiring the knowledge of θ, is known.

If a feedback controller that satisfies Assumption 2 is not known, then, under the additional assumption that the trajectories of (6) are exciting over the interval $[0, T)$, such a controller can be learned online using model-free reinforcement learning techniques such as [20]–[22].

IV. Model Based Reinforcement Learning

Lemma 1 implies that if a feedback controller that stabilizes the transformed system in (6) is designed, then the same feedback controller, applied to the original system by inverting the BT also achieves the control objective stated in Section II-A. In the following, the required controller is designed as an estimate of the controller that minimizes the infinite horizon cost[2]

$$J(u(\cdot)) := \int_0^\infty r(\phi(\tau, s^0, u(\cdot)), u(\tau))d\tau, \quad (15)$$

over the set $\mathcal{U}$ of piecewise continuous functions $t \mapsto u(t)$, subject to (6), where $\phi(\tau, s^0, u(\cdot))$ denotes the trajectory of (6), evaluated at time τ, starting from the state s^0 and under the controller $u(\cdot)$, $r(s, u) := s^T Q s + u^T R u$, and $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{q\times q}$ are symmetric positive definite (PD) matrices. Assuming that an optimal controller exists, let the optimal value function, denoted by $V^* : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{R}$, be defined as

$$V^*(s) := \min_{u(\cdot)\in\mathcal{U}_{t,\infty}} \int_t^\infty r(\phi(\tau, s, u(\cdot)), u(\cdot))d\tau, \quad (16)$$

where $\mathcal{U}_I$ is obtained by restricting the range of the functions in $\mathcal{U}$ to the interval I. Assuming that the optimal value function is continuously differentiable, it can be shown to be the unique positive definite solution of the Hamilton-Jacobi-Bellman (HJB) equation

$$\min_{u\in\mathbb{R}^q} \left(\nabla_s V(s)\left(F(s)+G(s)u\right)+s^T Q s+u^T R u\right)=0, \quad (17)$$

where $\nabla_s := \frac{\partial}{\partial s}$. Furthermore, the optimal controller is given by the feedback policy $u(t) = u^*(s(t))$ where $u^* : \mathbb{R}^n \to \mathbb{R}^q$ defined as $u^*(s) := -\frac{1}{2}R^{-1}G(s)^T(\nabla_s V^*(s))^T$.

A. Value function approximation

Since computation of analytical solutions of the HJB equation is generally infeasible, especially for systems with uncertainty, parametric approximation methods are used to approximate the value function V^* and the optimal policy u^*. The optimal value function is expressed as $V^*(s) = W^T\sigma(s) + \epsilon(s)$, where $W \in \mathbb{R}^L$ is an unknown vector of bounded weights, $\sigma : \mathbb{R}^n \to \mathbb{R}^L$ is a vector of continuously differentiable nonlinear activation functions such that $\sigma(0) = 0$ and $\nabla_s\sigma(0) = 0$, $L \in \mathbb{N}$ is the number of basis functions, and $\epsilon : \mathbb{R}^n \to \mathbb{R}$ is the reconstruction error.

Exploiting the universal function approximation property of single layer neural networks, it can be concluded that given any compact set $\chi \subset \mathbb{R}^n$ and a positive constant $\overline{\epsilon} \in \mathbb{R}$, there exists a number of basis functions $L \in \mathbb{N}$, and known positive constants $\bar{W}$ and $\overline{\sigma}$ such that $\|W\| \leq \bar{W}$, $\sup_{s\in\chi}\|\epsilon(s)\| \leq \overline{\epsilon}$ and $\sup_{s\in\chi}\|\nabla_s\epsilon(s)\| \leq \overline{\epsilon}$, $\sup_{s\in\chi}\|\sigma(s)\| \leq \overline{\sigma}$ and $\sup_{s\in\chi}\|\nabla_s\sigma(s)\| \leq \overline{\sigma}$ [24].

Using (17), a representation of the optimal controller using the same basis as the optimal value function is derived as

$$u^*(s) = -\frac{1}{2}R^{-1}G^T(s)\left(\nabla_s\sigma^T(s)W + \nabla_s\epsilon^T(s)\right).$$

Since the ideal weights, W, are unknown, an actor-critic approach is used in the following to estimate W. To that end, let the NN estimates $\hat{V} : \mathbb{R}^n \times \mathbb{R}^L \to \mathbb{R}$ and $\hat{u} : \mathbb{R}^n \times \mathbb{R}^L \to \mathbb{R}^q$ be defined as

$$\hat{V}\left(s, \hat{W}_c\right) := \hat{W}_c^T\sigma(s), \quad (18)$$

$$\hat{u}\left(s, \hat{W}_a\right) := -\frac{1}{2}R^{-1}G^T(s)\nabla_s\sigma^T(s)\hat{W}_a, \quad (19)$$

where the critic weights, $\hat{W}_c \in \mathbb{R}^L$ and actor weights, $\hat{W}_a \in \mathbb{R}^L$ are estimates of the ideal weights, W.

B. Bellman Error

Substituting (18) and (19) into (17) results in a residual term, $\hat{\delta} : \mathbb{R}^n \times \mathbb{R}^L \times \mathbb{R}^L \times \mathbb{R}^p \to \mathbb{R}$, which is referred to as Bellman Error (BE), defined as

$$\hat{\delta}(s, \hat{W}_c, \hat{W}_a, \hat{\theta}) := \nabla_s\hat{V}(s, \hat{W}_c)\left(y(s)\hat{\theta} + G(s)\hat{u}(s, \hat{W}_a)\right) + \hat{u}(s, \hat{W}_a)^T R\hat{u}(s, \hat{W}_a) + s^T Q s. \quad (20)$$

Traditionally, online RL methods require a persistence of excitation (PE) condition to be able learn the approximate control policy [12], [25], [26]. Guaranteeing PE a priori and verifying PE online are both typically impossible. However, using virtual excitation facilitated by model-based BE extrapolation, stability and convergence of online RL can established under a PE-like condition that, while impossible to guarantee a priori, can be verified online (by monitoring the minimum eigenvalue of a matrix in the subsequent Assumption 3).

Using the system model, the BE can be evaluated at any arbitrary point in the state space. Virtual excitation can then be implemented by selecting a set of states $\{s_k \mid k = 1, \cdots, N\}$ and evaluating the BE at this set of states to yield

$$\hat{\delta}_k(s_k, \hat{W}_c, \hat{W}_a, \hat{\theta}) := \nabla_s\hat{V}(s_k, \hat{W}_c)\left(y_k\hat{\theta} + G_k\hat{u}(s_k, \hat{W}_a)\right) + \hat{u}(s_k, \hat{W}_a)^T R\hat{u}(s_k, \hat{W}_a) + s_k^T Q s_k, \quad (21)$$

where, $y_k := y(s_k)$ and $G_k := G(s_k)$.

Defining the actor and critic weight estimation errors as $\tilde{W}_c := W - \hat{W}_c$ and $\tilde{W}_a := W - \hat{W}_a$ and substituting the

[2]For applications with bounded control inputs, a non-quadratic penalty function similar to [23, Eq. 17] can be incorporated in (15).

estimates (18) and (19) into (17), the BE can be expressed in terms of the weight estimation errors as[3]

$$\hat{\delta} = -\left(\omega^T \tilde{W}_c\right) + \left(\frac{1}{4}\tilde{W}_a^T G_\sigma \tilde{W}_a\right) - \left(W^T \nabla_s \sigma y \tilde{\theta}\right) + \Delta, \tag{22}$$

where $\Delta := \frac{1}{2}W^T\nabla_s\sigma G_R \nabla_s\epsilon^T + \frac{1}{4}G_\epsilon - \nabla_s\epsilon F$. $G_R := GR^{-1}G^T \in \mathbb{R}^{n\times n}$, $G_\epsilon := \nabla_s\epsilon G_R \nabla_s\epsilon^T \in \mathbb{R}$, $G_\sigma := \nabla_s\sigma G R^{-1} G^T \nabla_s\sigma^T \in \mathbb{R}^{L\times L}$, and $\omega := \nabla_s\sigma\left(y\hat{\theta} + G\hat{u}(s,\hat{W}_a)\right) \in \mathbb{R}^L$.

Similarly, (21) implies that

$$\hat{\delta}_k = -\left(\omega_k^T \tilde{W}_c\right) + \left(\frac{1}{4}\tilde{W}_a^T G_{\sigma_k} \tilde{W}_a\right) - \left(W^T \nabla_s \sigma_k y_k \tilde{\theta}\right) + \Delta_k, \tag{23}$$

where, $F_k := F(s_k)$, $\epsilon_k := \epsilon(s_k)$, $\sigma_k := \sigma(s_k)$, $\Delta_k := \frac{1}{2}W^T\nabla_s\sigma_k G_{R_k}\nabla_s\epsilon_k^T + \frac{1}{4}G_{\epsilon_k} - \nabla_s\epsilon_k F_k$, $G_{\epsilon_k} := \nabla_s\epsilon_k G_{R_k}\nabla_s\epsilon_k^T$, $\omega_k := \nabla_s\sigma_k\left(y_k\hat{\theta} + G_k\hat{u}(s_k,\hat{W}_a)\right) \in \mathbb{R}^L$, $G_{\sigma_k} := \nabla_s\sigma_k G_k R^{-1} G_k^T \nabla_s\sigma_k^T \in \mathbb{R}^{L\times L}$ and $G_{R_k} := G_k R^{-1} G_k^T \in \mathbb{R}^{n\times n}$.

Note that $\sup_{s\in\chi}|\Delta| \leq d\bar{\epsilon}$ and if $s_k \in \chi$ then $|\Delta_k| \leq d\bar{\epsilon}_k$, for some constant $d > 0$.

C. Update laws for Actor and Critic weights

The actor and the critic weights are held at their initial values over the interval $[0, T)$ and starting at $t = T$, using the instantaneous BE $\hat{\delta}$ from (20) and extrapolated BEs $\hat{\delta}_k$ from (21), the weights are updated according to

$$\dot{\hat{W}}_c = -k_{c_1}\Gamma\frac{\omega}{\rho}\hat{\delta} - \frac{k_{c_2}}{N}\Gamma\sum_{k=1}^{N}\frac{\omega_k}{\rho_k}\hat{\delta}_k, \tag{24}$$

$$\dot{\Gamma} = \beta\Gamma - k_{c_1}\Gamma\frac{\omega\omega^T}{\rho^2}\Gamma - \frac{k_{c_2}}{N}\Gamma\sum_{k=1}^{N}\frac{\omega_k\omega_k^T}{\rho_k^2}\Gamma, \tag{25}$$

$$\dot{\hat{W}}_a = -k_{a_1}\left(\hat{W}_a - \hat{W}_c\right) - k_{a_2}\hat{W}_a + \frac{k_{c_1}G_\sigma^T\hat{W}_a\omega^T}{4\rho}\hat{W}_c + \sum_{k=1}^{N}\frac{k_{c_2}G_{\sigma_k}^T\hat{W}_a\omega_k^T}{4N\rho_k}\hat{W}_c, \tag{26}$$

with $\Gamma(t_0) = \Gamma_0$, where $\Gamma : \mathbb{R}_{\geq t_0} \to \mathbb{R}^{L\times L}$ is a time-varying least-squares gain matrix, $\rho(t) := 1 + \gamma_1\omega^T(t)\omega(t)$, $\rho_k(t) := 1 + \gamma_1\omega_k^T(t)\omega_k(t)$, $\beta > 0 \in \mathbb{R}$ is a constant forgetting factor, and $k_{c_1}, k_{c_2}, k_{a_1}, k_{a_2} > 0 \in \mathbb{R}$ are constant adaptation gains. The control commands sent to the system are then computed using the actor weights as

$$u(t) = \begin{cases} \psi(s(t)), & 0 < t < T, \\ \hat{u}\left(s(t), \hat{W}_a(t)\right), & t \geq T. \end{cases} \tag{27}$$

The following verifiable PE-like rank condition is then utilized in the stability analysis.

[3]The dependence of various functions on the state, s, is omitted for brevity whenever it is clear from the context.

Assumption 3. There exists a constant $\underline{c}_3 > 0$ such that the set of points $\{s_k \in \mathbb{R}^n \mid k = 1, \ldots, N\}$ satisfies

$$\underline{c}_3 I_L \leq \inf_{t\in\mathbb{R}_{\geq T}}\left(\frac{1}{N}\sum_{k=1}^{N}\frac{\omega_k(t)\,\omega_k^T(t)}{\rho_k^2(t)}\right). \tag{28}$$

Since ω_k is a function of the weight estimates $\hat{\theta}$ and $\hat{W}_a$, Assumption 3 cannot be guaranteed a priori. However, unlike the PE condition, Assumption 3 can be verified online. Furthermore, since $\lambda_{\min}(\sum_{k=1}^{N}\frac{\omega_k(t)\omega_k^T(t)}{\rho_k^2(t)})$ is non-decreasing in the number of samples, N, Assumption 3 can be met, heuristically, by increasing the number of samples.

V. Stability Analysis

To facilitate the following analysis, let $Z := [s; \tilde{W}_c; \tilde{W}_a; \tilde{\theta}]$ denote the concatenated state of the closed-loop error system.

Theorem 1. Provided Assumptions 1-3 hold and the gains are selected large enough (see, e.g., [27], Algorithm A.2), then the system state s, weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$, and parameter estimation error $\tilde{\theta}$ are uniformly ultimately bounded.

Proof sketch. Under Assumption 1, the state trajectories are bounded over the interval $[0, T)$. Over the interval $[T, \infty)$, the developed parameter estimator satisfies the conditions of [27, Assumption 2]. Furthermore, the update laws developed in this paper are same to those developed in [18]. As a result, analysis of the controller developed in this arxiv paper follows the proof of [18, Theorem 1], and is omitted for brevity. □

VI. Simulation

To demonstrate the performance of the developed method, simulation results for a four-state dynamical system corresponding to a two-link planar robot manipulator are provided (see [10, page 114] for a complete description of the model.

The state x, that corresponds to angular positions and the angular velocities of the two links needs to satisfy the constraints, $x_1 \in (-7, 5)$, $x_2 \in (-7, 5)$, $x_3 \in (-5, 7)$ and $x_4 \in (-5, 7)$. The objective for the controller is to minimize the infinite horizon cost function in (15), with $Q = \text{diag}(1, 1, 1, 1)$ and $R = \text{diag}(1, 1)$ while identifying the unknown parameters $\theta \in \mathbb{R}^4$ that correspond to static and dynamic friction coefficients in the two links. The ideal values of the unknown parameters are $\theta_1 = 5.3$, $\theta_2 = 1.1$, $\theta_3 = 8.45$, and $\theta_4 = 2.35$.

The basis functions for value function approximation are selected as $\sigma(s) = [s_1s_3; s_2s_4; s_3s_2; s_4s_1; s_1s_2; s_4s_3; s_1^2; s_2^2; s_3^2; s_4^2]$. The initial conditions for the system and the initial guesses for the weights and parameters are selected as $x(0) = [-5; -5; 5; 5]$, $\hat{\theta}(0) = [5; 5; 5; 5]$, $\Gamma(0) = \text{diag}(10, 10, 10, 10, 10, 10, 10, 10, 10, 10)$, and $\hat{W}_a(0) = \hat{W}_c(0) = [60; 2; 2; 2; 2; 2; 40; 2; 2; 2]$. The ideal values of the actor and the critic weights are unknown. The simulation uses 100 fixed Bellman error extrapolation points in a 4x4 square around the origin of the $s-$coordinate system.

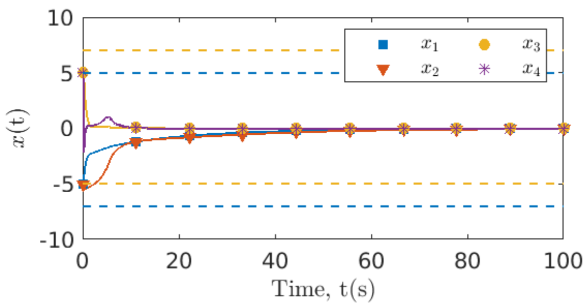

Fig. 1. State trajectories for the dynamical system using MBRL with FCL in the original coordinates. The dotted lines represent the user-selected safe set.

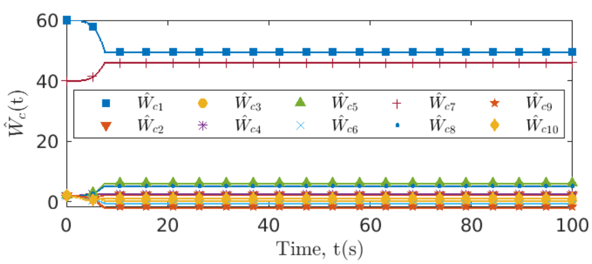

Fig. 2. Estimates of the critic weights under nominal gains.

A. Result

As seen from Fig. 1, the system state x stays within the user-specified safe set while converging to the origin. As demonstrated in Fig. 3 the parameter estimations converge to the true values.

Since the ideal actor and critic weights are unknown, the estimates cannot be directly compared against the ideal weights. To gauge the quality of the estimates, the trajectory generated by the controller $u(t) = \hat{u}\left(s(t), \hat{W}_c^*\right)$, where $\hat{W}_c^*$ is the final value of the critic weights obtained in Fig. 2, starting from a specific initial condition, is compared against the trajectory obtained using an *offline* numerical solution computed using the GPOPS II optimization software [28]. The total cost, generated by numerically integrating (15), is used as the metric for comparison. The results in Table I indicate that the two solution techniques generate slightly different trajectories in the state space (see Fig. 4) and the total cost of the trajectories is different. We hypothesize that the difference in costs is due to the basis for value function approximation being unknown.

In summary, the newly developed method can achieve online optimal feedback control thorough a BT MBRL approach while estimating the value of the unknown parameters in the system dynamics and ensuring safety guarantees in the original coordinates.

Remark 1. While the analysis of the developed technique dictates that a different stabilizing controller should be used over the time interval $[0, T)$, typically, as in this example, the transient response of the developed controller provides sufficient excitation so that T is small (of the order of $(10^{-6}$ in this example)), and the stabilizing controller is not needed.

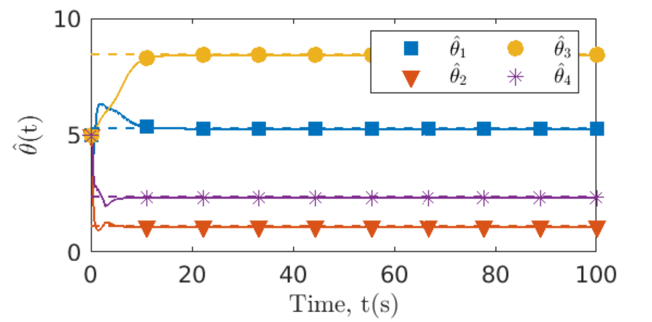

Fig. 3. Estimates of the unknown parameters in the system under the nominal gains.

TABLE I
COSTS FOR A SINGLE TRAJECTORY, OBTAINED USING THE DEVELOPED METHOD, AND USING PSEUDOSPECTRAL NUMERICAL OPTIMAL CONTROL SOFTWARE.

Method	Cost
BT MBRL with FCL	95.1490
GPOPS II [28]	57.8740

The following section details a one-at-a-time sensitivity analysis and study the sensitivity of the developed technique to changes in various tuning parameters.

B. Sensitivity Analysis

The parameters k_{c1}, k_{c2}, k_{a1}, k_{a2}, β, and v are selected for the sensitivity analysis. The costs of the trajectories, under the optimal feedback controller obtained using the developed method, are presented in Table II for 5 different values of each parameter. The parameters are varied in a neighborhood of the nominal values (selected through trial and error) $k_{c1} = 0.1$, $k_{c2} = 10$, $k_{a1} = 20$, $k_{a2} = 0.2$, $\beta = 0.8$, and $v = 100$. The value of β_1 is set to be $\text{diag}(100, 100, 100, 100)$. The results in Table II indicate that the developed method is sensitive to changes in the learning gains.

VII. Conclusion

This paper presents a novel online MBRL based controller which uses BFs, BE extrapolation and a novel FCL method. A known BF transformation is applied to a constrained optimal control problem to generate an unconstrained optimal control problem in the transformed coordinates. The

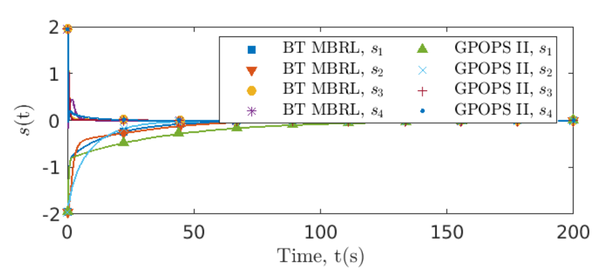

Fig. 4. Comparison of the optimal state trajectories obtained using GPOPS II and using BT MBRL with FCL and fixed optimal weights.

TABLE II

SENSITIVITY ANALYSIS

k_{c_1}=	0.01	0.05	0.1	0.5	1
Cost	95.91	95.4185	95.1490	94.1607	93.5487
k_{c_2}=	1	5	10	20	30
Cost	304.4	101.0786	95.1490	92.7148	93.729
k_{a_1}=	5	10	20	30	50
Cost	94.9464	95.1224	95.1490	95.1736	95.1974
k_{a_2}=	0.05	0.1	0.2	0.5	1
Cost	95.2750	95.2480	95.1490	94.9580	94.6756
β=	0.1	0.5	0.8	0.9	0.95
Cost	125.33	109.7721	95.1490	92.91	93.7231
υ=	50	70	100	125	150
Cost	92.2836	93.34	95.1490	96.1926	97.9870

system dynamics, if linear in the parameters in the original coordinates, are shown to be also linearly parameterized in the transformed coordinates. MBRL is used to solve the problem online in the transformed coordinates in conjunction with the novel FCL to learn the unknown model parameters. Regulation of the system states to a neighborhood of the origin and convergence of the estimated policy to a neighborhood of the optimal policy is determined using a Lyapunov-based stability analysis.

In the developed method, the cost function is selected to be quadratic in the transformed coordinates. However, a physically meaningful cost function is more likely to be available in the original coordinates. To our best knowledge, this is the first time, any one has developed a safety aware model based reinforcement learning method using BT for the system with parametric uncertainties. Techniques to transform cost functions from the original coordinates to the barrier coordinates to ensure that optimization in barrier coordinates also corresponds to optimization in the original coordinates is a topic for future research.

REFERENCES

[1] R. S. Sutton and A. G. Barto, *Reinforcement learning: an introduction*. Cambridge, MA, USA: MIT Press, 1998.

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

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

[4] D. P. Bertsekas, "Dynamic programming and optimal control 3rd edition, volume II," *Belmont, MA: Athena Scientific*, 2011.

[5] R. Kamalapurkar, L. Andrews, P. Walters, and W. E. Dixon, "Model-based reinforcement learning for infinite-horizon approximate optimal tracking," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 753–758, Mar. 2017.

[6] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, "An actor-critic-identifier architecture for adaptive approximate optimal control," in *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*, ser. IEEE Press Series on Computational Intelligence, F. L. Lewis and D. Liu, Eds. Wiley and IEEE Press, Feb. 2012, pp. 258–278.

[7] K. Vamvoudakis, D. Vrabie, and F. L. Lewis, "Online policy iteration based algorithms to solve the continuous-time infinite horizon optimal control problem," in *IEEE Symp. Adapt. Dyn. Program. Reinf. Learn.*, 2009, pp. 36–41.

[8] 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.

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

[10] R. Kamalapurkar, P. Walters, J. A. Rosenfeld, and W. E. Dixon, *Reinforcement learning for optimal feedback control: A Lyapunov-based approach*, ser. Communications and Control Engineering. Springer International Publishing, 2018.

[11] W. He, Z. Li, and C. L. P. Chen, "A survey of human-centered intelligent robots: issues and challenges," *IEEE/CAA J. Autom. Sin.*, vol. 4, no. 4, pp. 602–609, 2017.

[12] H. Modares, F. L. 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.

[13] K. G. Vamvoudakis, M. F. Miranda, and J. P. Hespanha, "Asymptotically stable adaptive–optimal control algorithm with saturating actuators and relaxed persistence of excitation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2386–2398, 2016.

[14] K. Graichen and N. Petit, "Incorporating a class of constraints into the dynamics of optimal control problems," *Optimal Control Applications and Methods*, vol. 30, no. 6, pp. 537–561, 2009.

[15] C. P. Bechlioulis and G. A. Rovithakis, "Adaptive control with guaranteed transient and steady state tracking error bounds for strict feedback systems," *Automatica*, vol. 45, no. 2, pp. 532 – 538, 2009.

[16] Y. Yang, K. G. Vamvoudakis, H. Modares, W. He, Y.-X. Yin, and D. Wunsch, "Safety-aware reinforcement learning framework with an actor-critic-barrier structure," in *Proc. Am. Control Conf.*, 2019, to appear.

[17] M. L. Greene, P. Deptula, S. Nivison, and W. E. Dixon, "Reinforcement learning with sparse bellman error extrapolation for infinite-horizon approximate optimal regulation," in *Proc. IEEE Conf. Decis. Control*, Nice, Fr, Dec. 2019, pp. 1959–1964.

[18] N. S. M. Mahmud, K. Hareland, S. A. Nivison, Z. I. Bell, and R. Kamalapurkar, "A safety aware model-based reinforcement learning framework for systems with uncertainties," arXiv:2007.12666, 2020, submitted to IEEE Control Systems Letters.

[19] S. B. Roy, S. Bhasin, and I. N. Kar, "Parameter convergence via a novel PI-like composite adaptive controller for uncertain Euler-Lagrange systems," in *Proc. IEEE Conf. Decis. Control*, Dec. 2016, pp. 1261–1266.

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

[21] D. Vrabie and F. Lewis, *Online Adaptive Optimal Control Based on Reinforcement Learning*. New York, NY: Springer New York, 2010, pp. 309–323.

[22] 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.

[23] Y. Yang, D.-W. Ding, H. Xiong, Y. Yin, and D. C. Wunsch, "Online barrier-actor-critic learning for h∞ control with full-state constraints and input saturation," *Journal of the Franklin Institute*, vol. 357, no. 6, pp. 3316 – 3344, 2020.

[24] F. L. Lewis, R. Selmic, and J. Campos, *Neuro-fuzzy control of industrial systems with actuator nonlinearities*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2002.

[25] R. Kamalapurkar, J. A. Rosenfeld, and W. E. Dixon, "State following (StaF) kernel functions for function approximation Part II: Adaptive dynamic programming," in *Proc. Am. Control Conf.*, Chicago, IL, USA, Jul. 2015, pp. 521–526.

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

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

[28] M. A. Patterson and A. V. Rao, "GPOPS-II: A MATLAB software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming," *ACM Trans. Math. Softw.*, vol. 41, no. 1, Oct. 2014.