Integration of Adaptive Control and Reinforcement Learning for Real-Time Control and Learning

Anuradha M. Annaswamy, Life Fellow, IEEE, Anubhav Guha, Yingnan Cui, Sunbochen Tang, Graduate Student Member, IEEE, Peter A. Fisher, and Joseph E. Gaudio

Abstract—This article considers the problem of real-time control and learning in dynamic systems subjected to parameteric uncertainties. We propose a combination of a reinforcement learning (RL)-based policy in the outer loop suitably chosen to ensure stability and optimality for the nominal dynamics, together with adaptive control (AC) in the inner loop so that in real-time AC contracts the closed-loop dynamics toward a stable trajectory out by RL. In total, two classes of nonlinear dynamic systems are considered, both of which are control affine. The first class of dynamic systems utilizes equilibrium points and a Lyapunov approach, whereas second class of nonlinear systems uses contraction theory. AC-RL controllers are proposed for both classes of systems and shown to lead to online policies that guarantee stability using a high-order tuner and accommodate parameteric uncertainties and magnitude limits on the input. In addition to establishing a stability guarantee with real-time control, the AC-RL controller is also shown to lead to parameter learning with persistent excitation for the first class of systems. Numerical validations of all algorithms are carried out using a quadrotor landing task on a moving platform.

Index Terms—Adaptive control, parameter learning, persistent excitation, reinforcement learning, stability.

I. INTRODUCTION

This article considers the problem of real-time control and learning of a class of dynamic systems with parameteric uncertainties using a combination of adaptive control (AC) and reinforcement learning (RL). Over the past seven decades, the field of AC has focused on ensuring global boundedness with asymptotic tracking, parameter learning with persistent excitation, and robustness to nonparameteric uncertainties, such as disturbances and unmodeled dynamics for a large class of linear and nonlinear dynamic systems [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11]. The field of RL has focused on the determination of a sequence of inputs that drives a dynamical system to minimize a suitable objective with minimal knowledge of the system model. The central structure of the RL solution revolves around a policy that is learned so as to maximize a desired reward [12], [13], [14], [15].

Both AC and RL-based control methods have addressed the problem of control in the presence of uncertainty with a component of learning, but with entirely different approaches. AC methods have been proven to be effective in the enforcement of objectives for specific classes of problems, such as regulation and tracking, in real time, with analytical guarantees. In addition to accommodating parameteric uncertainties, constraints on the control input magnitude [16], [17] and rate [18] can also be incorporated within an analytical framework. They are, however, unable to directly guarantee the realization of long-term optimality-based objectives. RL-trained policies, on the other hand, can handle a broad range of objectives [15], where the control policies are learned offline in simulation, allowing for a near infinite number of agent–environment interactions to allow the policy to become near optimal with guarantees of local stability [14], [19], [20], [21], [22], [23]. In practice, however, offline policies trained in simulation often exhibit degenerate performance when used for real-time control due to modeling errors that can occur online [24], [25], [26], [27], [28]. This article takes a first step toward an integration of the AC and RL approaches so as to reduce this “sim-to-real” gap by realizing a combined set of advantages of both approaches.

The AC-RL approach that we propose in this article uses AC-based components in the inner loop and RL-based components in the outer loop. The AC-RL controller is designed such that the inner loop AC accommodates the effect of parameteric uncertainties in real time and contracts the closed-loop dynamics toward a reference system through a nonlinear adaptive law. The RL policy in the outer loop, which is used to determine the control input to the reference system, is designed to lead to an almost-optimal control policy for the reference system, through training in simulation. An important point to note is...
that this policy is chosen so that local stability is ensured, which
can be accomplished by introducing additional components in
the underlying loss function (e.g., [23]). In total, two classes of
nonlinear dynamic systems are considered, denoted as Problems
1 and 2. The first class of dynamic systems utilizes equilibrium
points and expansion forms around these points and employs
a Lyapunov approach to establish convergence. The second
class of nonlinear systems uses contraction theory as the un-
derlying framework [29]. In each class, an AC-RL controller
is proposed for a class of parameteric uncertainties, and the
resulting closed-loop system is proved to be globally bounded
with asymptotic properties of tracking and cost. In both the
problems, the states of the dynamic system are assumed to be
accessible for measurement.

In the first class of systems, the RL component in the outer
loop is chosen so as to lead to near-optimality of a desired
cost function with analytical guarantees of local stability for a
nonlinear reference system [23]. With this RL, and an AC-based
design that employs a high-order tuner (HT) in the inner loop for
parameter estimation, we show that for a class of parameteric
uncertainties, the closed-loop system, with the integrated AC-RL
controller, can be guaranteed to be stable for all initial conditions
of the true system. We also quantify the asymptotic behavior
of its state, input, and overall cost. The advantages of the
AC-RL controller are illustrated using numerical experiments
of a quadrotor required to land on a moving platform using a
nonlinear high-fidelity model, illustrating the reduction in the
sim-to-real gap. Various extensions of the AC-RL approach
are also proposed, including a relaxation of the parameteric
uncertainties in the underlying nonlinearities, inclusion of mul-
tiple equilibrium points, nonaffine systems, and incorporation of
magnitude limits in the control input. In all cases except the
last extension, the AC-RL ensures global boundedness; in the
last case, all the initial conditions of the state and parameter
estimates that lie inside a bounded domain are shown to lead to
boundedness for an arbitrary nonlinear dynamic system.

In the second class of dynamic systems, denoted as Problem
2, we introduce parameteric uncertainties similar to the first
class and employ contraction theory to establish convergence,
which utilizes tools in Riemannian geometry to analyze the con-
vergence of differential dynamics by constructing differential
Lyapunov functions [29], [30]. In contrast to prior work [31], we
introduce multiplicative parameteric uncertainties in the control
input, employ an HT for parameter estimation, and an RL-based
outer loop as before, with analytical guarantees of local stability
to synthesize the reference system.

In both the classes, the central component of the AC algorithm
uses an HT, which was first proposed in [32], and later expanded
in [33] and [34] in an effort to develop stable low-order adaptive
controllers. Independently, HT has also been sought after in
the machine learning (ML) community, in an effort to obtain
accelerated convergence of an underlying cost function and
the associated accelerated learning of the minimizer of this
function (see, for example, [35], [36], [37], [38], and [39]).
The basic idea is to include momentum-based updates so as to
get a faster convergence of the performance error. The HT
algorithms in [32] have seen widespread applications in machine
learning [40]. They were shown in [41], [42], and [43] to lead
to stability with time-varying regressors, and in [42] to have
accelerated convergence with constant regressors in discrete
time. The first contribution of this article is the demonstration of
these properties of global boundedness and asymptotic behavior
of the overall AC-RL controller in Problems 1 and 2.

The second contribution of this article is the demonstration of
parameter learning in uncertain dynamic systems. We focus
our attention primarily on the first class of dynamic systems
with a scalar input. We show that under conditions of persistent
excitation [1], the AC-RL controller with the HT guarantees that
the parameter estimates converge to their true values, thereby
allowing parameter learning. Since the HT inserts additional fil-
tering actions, the proof of parameter convergence is a nontrivial
result. Earlier results related to parameter convergence with HT
required additional processing and time-scale transformations
on the underlying regressors [33], which are avoided in this arti-
cle. The main benefits of parameter learning are robustness [10]
and avoiding the bursting phenomenon [1], [44].

Efforts to combine AC and RL approaches have been high-
lighted in several recent works, which include [45], [46], [47],
and [48] in discrete time and [49], [50], [51], and [52] in con-
tinuous time. Richards et al. [50] proposed the use of AC
for nonlinear systems in a data-driven manner, but requires
offline trajectories from a target system and does not address
accelerated learning or magnitude saturation. The authors in [51]
and [52] studied the linear-quadratic-regulator (LQR) problem
and its AC variants from an optimization and machine learning
perspective. In [47], an RL approach is used to determine an
adaptive controller for an unknown system, whereas in [48],
principles from AC and Lyapunov analysis are used to adjust
and train a deep neural network. These approaches, including
our earlier work in [49], have not addressed a comprehensive
treatment of an integrated AC-RL approach with analytical
guarantees. While the use of an AC in the inner loop and
RL in the outer loop is a common feature, all other elements in
this article, including HT, magnitude saturation, multiple
equilibrium points, nonaffine systems, and contraction-theory
based AC design have been considered for the first time. None
of these prior papers have addressed the problem of parameter
convergence using the AC-RL controller, and this is also a key
contribution of this article.

In summary, the contributions of this article are the follow-
ing, which are applicable to two classes of nonlinear dynamic
systems with certain types of parameteric uncertainties.

1) An AC-RL controller approach ensures global stability
with the AC in the inner loop employing the HT and RL
components in the outer loop, with asymptotic quantifi-
cation of its state, input, and overall cost with respect
to a reference system. The RL-based control policies
are designed so that local stability is guaranteed and are
near-optimal for the reference system.

2) A guarantee of parameter learning in real-time using the
AC-RL controller with persistent excitation.

The rest of this article is organized as follows. In Section II,
we describe the problem statements. Section III includes overviews
of a range of underlying tools related to the RL and AC ap-
proaches. Sections IV and V describe the AC-RL controller
for Problem 1 and Section VI outlines the controller for Problem
2. Section VII includes numerical studies of the AC-RL controllers
for Problems 1 and 2. Proofs of Theorems 1 and 4–8 as well as all
lemmas can be found in the Appendixes, with additional details
in [53]. Proofs of Theorems 2 and 3 can be found in [30] and [54],
respectively.
II. PROBLEM STATEMENT

We consider two classes of problems in this article, for which we propose control and learning solutions in real time. In all problems, states are assumed to be accessible.

A. Problem 1: Real-Time Control and Learning for Systems With Equilibrium Points

Consider a continuous-time, deterministic nonlinear system described by the following dynamics:

$$\dot{X}(t) = F(X(t), U(t))$$  \(1\)

where \(X(t) \in \mathbb{R}^n\) and \(U(t) \in \mathbb{R}^m\). We define \(x = X - X_0\) and \(u = U - U_0\), where \((X_0, U_0)\) is an equilibrium point, i.e., \(F(X_0, U_0) = 0\). Using a Taylor series expansion on \(1\) yields

$$\dot{x} = Ax + Bu + f(x, u)$$  \(2\)

where \(f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n\), and \((A, B)\) is controllable. It should be noted that \(1\) can always be written in the form of \(2\) for any analytic function \(F\). The following assumption is made about \(f\).

Assumption 1: The higher order effects represented by the nonlinearity \(f(x, u)\) in \(2\) lie in the span of \(u\) and \(g(x)\) exists such that \(Bg(x) = f(x, u)\)

$$\dot{x} = Ax + B[u + g(x)].$$  \(3\)

The goal in Problem 1 is the determination of the control input \(u(t)\) in real time, when parameter uncertainties are present in the system dynamics in \(3\), so as to minimize the cost function

$$\min_{u(t) \in U[0, t_0 + T]} \int_{t_0}^{t_0 + T} c(x(t), u(t))dt$$  \(4\)

subject to the dynamics in \(3\), where \(U\) represents the set of all allowable inputs, and \(c\) is a bounded function that is piecewise continuous in \(x\) and continuous in \(u\).

B. Problem 2: Real-Time Control for a Class of Nonlinear Systems

Problem 2 corresponds to a class of control-affine nonlinear systems with parameteric uncertainties, which does not necessarily utilize the presence of an equilibrium point. This class is assumed to be of the form

$$\dot{X}(t) = f(X(t)) + B(X(t))U(t)$$  \(5\)

where \(X(t) \in \mathbb{R}^n\) and \(U(t) \in \mathbb{R}^m\). We note that \(5\) differs from the expansion form of \(2\) and, therefore, allows both \(f(X)\) and \(B(X)\) to be nonlinear. The goal once again is to determine \(U(t)\) in real time, which minimizes a cost function of \(X(t)\) and \(U(t)\), in the same form as \(4\), when parameter uncertainties are present in \(f(X)\) and \(B(X)\).

III. UNDERLYING TOOLS

A. Offline Approach Based on RL

We define a reference system

$$\dot{x}_r = Ax_r + Bu_r + f_r(x_r, u_r)$$  \(6\)

where the subscript \(r\) is used to denote signals and parameters of the reference system. We briefly describe the offline training procedure in an RL: First, it is assumed that the continuous-time dynamics in \(6\) are sampled with sufficient accuracy, resulting in the discrete time dynamics

$$x_{r,k+1} = h(x_{r,k}, u_{r,k}).$$  \(7\)

An appropriate numerical integration scheme ensures that this discrete-time formulation closely approximates the dynamics. The goal of RL is to learn a feedback policy \(u_{r,k} = \pi(x_{r,k})\), with repeated training of \(\pi(\cdot)\), so as to achieve the control objective of \(4\) \([55]\).

In order to facilitate such a training, a simulation environment is assumed to be available so that each timestep, an observation \(x_{r,k}\) is received, a control \(u_{r,k}\) is chosen, and the resulting cost \(c_k = c(x_{r,k}, u_{r,k})\) is computed. Repeating this process, a set of input-state-cost tuples \(D = [(x_{r,1}, u_{r,1}, c_1), \ldots, (x_{r,N}, u_{r,N}, c_N)]\) is formed. These data are used to train and update the policy \(\pi\) \([55]\). Often the policy is parameterized using neural networks so that \(u_k = \pi(x_k)\), where \(\theta\) denotes the weights of the neural network. A learning algorithm, often based on a gradient-descent approach, is used to adjust \(\theta\) so that the expected accumulated cost is minimized \([56]\).

In stochastic settings, the RL goal can be summarized as

$$\min_{\theta} J(\theta) = \mathbb{E}_{\pi(\theta)} \left[ \sum_{k=0}^{T} c_k \right].$$  \(8\)

1) Analytical Guarantees: While majority of the RL literature is focused on extensive simulation studies, a few papers have developed specific RL methods with analytical properties, which can be used for control of nonlinear systems (see, for example, \([14, 19, 20, 22, 21, 23]\)). Watkins and Dayan \([14]\) showed that convergence of Q-learning algorithms can be achieved using action-replay process, Jaakkola et al. \([19]\) developed convergence analysis based on stochastic approximation techniques that apply to both Q-learning and TD(\(\lambda\)) algorithms, and Konda and Tsitsiklis \([20]\) discussed local convergence results for actor–critic algorithms. Convergence analyses in \([14, 19, \) and \(20]\) are established under a tabular setting where state and action spaces are both finite, which may be restrictive for control applications since most physical systems have continuous state space. Results in papers, such as \([21, 22, \) and \(23]\), avoid this restriction. Lalé et al. \([21]\) proposed an RL algorithm that provide stability guarantee for LQR problems; results in \([22]\) provide a proof of convergence of global optimality, and hence stable policy for linear MDPs and cost functions that are a linear combination of a set of known feature functions. In \([23]\), RL is explored for general nonlinear systems, with the requirement that the initial policy is stable and a modification of the cost function that enforces Lyapunov stability-based constraints. Overall, the approach in \([23]\) can be viewed as a policy design with a tradeoff of optimality for closed-loop stability. We propose the use of such RL-based policies in the RL part of our AC-RL controller.

B. Classical AC Methods

We start with an error model description of the form

$$\dot{e}(t) = A_m e(t) + BA \left[ \hat{G}(t) \hat{\Phi}(t) \right].$$  \(9\)
where $c(t) \in \mathbb{R}^n$ is a performance error that is required to be brought to zero, such as tracking error, identification error, and state estimation error. $\dot{\Theta}(t) \in \mathbb{R}^{m \times l}$ is a matrix of parameter errors that quantify the learning error. If the true parameter in the system dynamics is $\Theta^* \in \mathbb{R}^{m \times l}$, and it is estimated as $\hat{\Theta}(t)$ at time $t$, then $\dot{\Theta} = \dot{\Theta} - \Theta^*$. $A_m$ is a Hurwitz matrix, $A_m$ and $B$ are known with $(A_m, B)$ controllable, and $\Lambda \in \mathbb{R}^{m \times m}$ is an unknown parameter matrix that is positive definite. Finally, $\Phi(t) \in \mathbb{R}^l$ is a vector of regressors that correspond to all real-time information measured or computed from the system dynamics and controllers at time $t$. Such an error model is ubiquitous in AC of a large class of nonlinear dynamic systems with parametric uncertainties [1] including (3). The following theorem summarizes the standard global stability result in AC.

**Theorem 1**: Let $\Gamma \in \mathbb{R}^{m \times m}$ and $Q_a$ be symmetric positive-definite matrices with $P$ corresponding to the solution of the Lyapunov equation

$$A_m^T P + PA_m = -Q_a. \quad (10)$$

An adaptive law that adjusts the parameter estimate in real time as

$$\dot{\hat{\Theta}} = -\Gamma B^T P e \Phi^T \quad (11)$$
guarantees that $e(t)$ and $\hat{\Theta}(t)$ are bounded for any initial conditions $e(0)$ and $\hat{\Theta}(0)$. If in addition $\Phi(t)$ is bounded for all $t$, then $\lim_{t \to \infty} e(t) = 0$.

**C. High-Order Tuners**

The core of the adaptive components proposed in this article is based on HTs [32], [33], [34]. The idea behind HT is summarized in the following.

Parameter identification in a linear regression problem of the form $y(t) = \theta^T \phi(t)$, where $\theta \in \mathbb{R}^n$ represents an unknown parameter, and $\phi(t) \in \mathbb{R}^n$ is an underlying regressor that can be measured at each $t$, can be formulated as a minimization problem $L_\theta = \epsilon^2(t)$, where $\epsilon = y - y^*$ and $y^*(t) = \theta^T \phi(t)$. A first-order tuner that can be used to solve the minimization problem is of the form $\hat{\theta} = -\gamma \nabla_\theta L_\theta$. In [41] and [42], an HT of the form

$$\dot{\hat{\theta}} + \beta \hat{\theta} = -\frac{\gamma \beta}{N_\ell} \nabla_\theta L_\theta(\theta) \quad (12)$$

was proposed and shown to correspond to the Lagrangian

$$L(\theta, \dot{\theta}, t) = e^{\beta(\ell - \lambda)} \left( \frac{1}{2} \|\dot{\theta}\|^2 - \frac{\gamma \beta}{N_\ell} L_\theta(\theta) \right). \quad (13)$$

The benefits of the HT in (12) are that 1) it can be guaranteed to be stable even with time-varying regressors for a dynamic error model with a scalar control input [41], and 2) a particular discretization was shown in [42] that lead to an accelerated algorithm, which reaches an $\epsilon$ suboptimal point in $O(1/\sqrt{\epsilon})$ iterations for a linear regression-type convex loss function with constant regressors, as compared with the $O(1/\epsilon)$ guaranteed rate for the standard gradient descent algorithm. We employ elements of HT in [41] in the AC part of our AC-RL controller.

**D. Contraction Theory**

Contraction theory provides tools to analyze the stability of nonlinear systems beyond attraction to an equilibrium point.

Based on results in Riemannian geometry, control contraction metrics [30] have been developed to design a feedback controller that stabilizes a class of control-affine systems to a reference trajectory. A brief review of the salient features of this approach follows.

For an arbitrary pair of points $X_1, X_2 \in \mathbb{R}^n$, let $\zeta(X_1, X_2)$ denote the set of all smooth curves connecting $X_1$ and $X_2$, in which each path $c \in \zeta(X_1, X_2)$ is parameterized by $s \in [0, 1]$, i.e., $c(s) : [0, 1] \to \mathbb{R}^n$, $c(0) = X_1, c(1) = X_2$. Let $S^*_n$ be the set of $n \times n$ positive-definite matrices. A Riemannian metric $M(X) : \mathbb{R}^n \to S^*_n$ is a function that defines a smoothly varying inner product $\langle \cdot, \cdot \rangle$ on the tangent space of $\mathbb{R}^n$. A Riemannian metric $M(X)$ is uniformly bounded if there exists $\alpha_2 \geq \alpha_1 > 0$ such that $\alpha_1 I \leq M(X) \leq \alpha_2 I$. For a smooth path $c(s, t)$ between the pair $(X_1(t), X_2(t))$, which is a time-varying smooth curve parameterized by $s$ at any given $t \geq 0$, its Riemannian length and energy are defined as follows:

$$L(c) := \int_0^1 \|c_s\|e ds, \quad E(c) := \int_0^1 \|c_s\|_e^2 ds$$

where $c_s := \frac{\partial c}{\partial s}$ and $\|c_s\|_e = \sqrt{\epsilon^2 M(c(s, t))}$. A smooth curve is regular if $\frac{\partial c(s, t)}{\partial s} \neq 0$ for all $s \in [0, 1]$.

**E. Persistent Excitation and Parameter Learning**

**Definition 1 (Persistent excitation [1])**: A bounded function $\Phi : [\ell_0, \infty) \to \mathbb{R}^N$ is persistently exciting (PE) if there exists
\[ T > 0 \) and \( \alpha > 0 \) such that
\[ \int_{t}^{t+T} \Phi(\tau)\Phi^T(\tau)d\tau \geq \alpha I \quad \forall t \geq t_0, \quad (15) \]
The definition of PE in (15) is equivalent to the following:
\[ \frac{1}{T} \int_{t}^{t+T} \Phi(\tau)^Tw_1d\tau \geq \epsilon_0 \quad \forall \text{ unit vectors } w \in \mathbb{R}^N \quad \forall t \geq t_0 \]
(16)
for some \( \epsilon_0 > 0 \). In what follows, we will utilize an alternate definition, which is equivalent to (15) and (16) if \( \|\Phi(t)\| \) is bounded for all \( t \) [58].

Definition 2: \( \Phi \) is PE if there exists an \( \epsilon > 0 \) a \( t_2 \) and a subinterval \( [t_2, t_2 + \delta_0] \subset [t, t + T] \) with
\[ \frac{1}{T} \int_{t_2}^{t_2+\delta_0} \Phi(\tau)^Tw_1d\tau \geq \epsilon_0 \quad \forall \text{ unit vectors } w \in \mathbb{R}^N \quad \forall t \geq t_0. \]
(17)
Adaptive control systems enable parameter learning by imposing properties of persistent excitation defined in Section III-E. We briefly summarize the classical result related to this topic, established in [54] and [58]. The starting point is the same error model as in (9), which contains two errors: \( e(t) \in \mathbb{R}^N \) is a performance error that can be measured, and \( \tilde{\theta}(t) = \Theta(t) \in \mathbb{R}^{1 \times l} \) is a parameter learning error. As before, we assume that \( A_m \) is known and Hurwitz, and \( B \) is known. The following theorem summarizes this result.

Theorem 3 (see [54]): The solutions of the error dynamics in (9) together with the adaptive law in (11) lead to \( \lim_{t \to \infty} \tilde{\theta}(t) = 0 \) if the regressor \( \Phi \) satisfies the PE condition in Definition 2.

IV. INTEGRATED AC-RL SOLUTIONS FOR REAL-TIME CONTROL AND LEARNING: PROBLEM 1

We address Problem 1 in this section, and propose an integrated AC-RL controller that combines AC and RL approaches.

A. Restatement of Problem 1

We shall denote (3) when there are no parameteric uncertainties as the reference system, and express it as
\[ \dot{x}_r = Ax_r + B[u_r + g(x_r)]. \]
We assume that \( u_r \) is designed using RL
\[ u_r = \pi(x_r). \]

It should be noted that \( \pi(x_r) \) in (19) does not necessarily cancel \( g(x_r) \), but accommodates it so that the closed-loop system behaves in a satisfactory manner for the requisite control task. This is quantified in Assumption 2, a desirable property of RL controllers.

Assumption 2: RL is used to train a feedback policy \( \pi : x \to u \) such that for a given positive constant \( R_2 \), a positive constant \( R_1 \) exists such that \( \|x_r(0)\| \leq R_1 \) implies \( \|x_r(t)\| \leq R_2 \) \( \forall t \)

Remark 2: Assumption 2 encapsulates the notion that, given a sufficiently long training period, a performant RL algorithm, and appropriate choices of hyperparameters, the learned policy \( \pi \) will be "good" enough to maintain boundedness of the reference system states. As mentioned in Section III-A, for tabular settings where the action and state are finite [14], [19], and for linear dynamics and cost functions that are linear combinations of known feature functions [22], the learned policy is optimal and, therefore, stable, which validates Assumption 2. For nonlinear dynamics, Berkenkamp et al. [23] showed that a stable controller can be learned by optimizing a soft loss function that augments cost functions in (4) with Lyapunov stability-based constraints. Relaxing the soft loss constraints and expanding the scope of RL for general nonlinear systems is one of the objectives in the area of safe RL, a topic of intense research activity. We note that any of the aforementioned frameworks, including [23], can be used to compute a stable RL policy \( u_r \) in (19). In order to avoid the associated optimality gap induced by these soft loss constraints, methods such as proximal policy optimization (PPO), shown to guarantee local optimality in [59], and engineer the underlying cost function through the addition of high-magnitude penalties only on out-of-set excursions.

1) Parametric Uncertainties: We now introduce two parametric uncertainties, one in the form of control effectiveness, i.e., \( u \) gets perturbed as \( A \), and the second as a perturbation in \( g(x) \). Loss of control effectiveness is ubiquitous in practical problems, due to unforeseen anomalies that may occur in real time, such as accidents or aging in system components, especially in actuators. Parametric uncertainties in the nonlinearity \( g(x) \) may be due to modeling errors. The following assumptions are made about these two uncertainties.

Assumption 3: The nonlinearity \( g(x) \) in (3) is parameterized linearly, i.e., \( g(x) = \Theta_{n,r} \Phi_n(x) \), where \( \Theta_{n,r} \in \mathbb{R}^{m \times l} \) and \( \Phi_n(x) \in \mathbb{R}^l \).

Assumption 4: \( \Lambda \) is symmetric and positive definite, with \( \|\Lambda\| \leq 1 \).

Remark 3: Assumption 3 implies that \( g(x) \) in (3) can be approximated accurately using \( \lambda \) basis functions. The use of basis functions to approximate a nonlinearity is a commonplace in AC: see, e.g., [60], [61], and [62]. For simplicity, we assume that the approximation is exact in Assumption 3, which will be relaxed in Section V-A. Assumption 4 introduces a structure in the loss of control effectiveness.

With Assumption 3 in place, a second parameteric uncertainty is introduced where the nonlinear component \( g(x) \) is perturbed as \( \Theta_n \Phi_n(x) \) to \( \Theta_n', \Phi_n(x) \). With these parametric uncertainties, the plant equation in (3) then becomes
\[ \dot{x} = Ax + BA \left[u + \Lambda^{-1}\Theta_n' \Phi_n(x)\right] \quad (20) \]
where \( B \in \mathbb{R}^{n \times m}, A \in \mathbb{R}^{m \times m} \) and \( \Theta_n' \in \mathbb{R}^{m \times l} \). Altogether we note that Problem 1 is restated as the control of (20) where \( A, B, \) and \( \Phi_n(x) \) are known, but \( \Lambda \) and \( \Theta_n' \) are unknown parameters.

B. AC-RL Controller

The goal of the AC-RL controller is to design a control input \( u \) in (20) such that the true system state \( x \) converges to the reference system state \( x_r \) in the presence of the aforementioned perturbations in \( \Lambda \) and \( \Theta_n' \). Subtracting (18) from (20), we obtain the error dynamics
\[ \dot{e} = Ahe + BAu - \Theta' \Phi \]
(21)
where
\[ e := x - x_r, \quad \Theta := \Lambda^{-1} \left[I - \Theta_n'\right], \quad \Phi := \begin{bmatrix} \Phi_n(u_r, x_r, x) \\ \Phi_n(x) \end{bmatrix} \]
(22)
and \( \Theta_{l,r} \) is such that \( A_{HI} := A + B\Theta_{l,r} \) is Hurwitz. In (21), \( \Theta \in \mathbb{R}^{m \times (m+1)} \) corresponds to an unknown parameter matrix and \( \Phi(t) \in \mathbb{R}^{m+l} \) is the regressor vector used for adaptation. The error equation (21) is central to the development of the
AC-RL algorithms derived in the following sections. We now propose the AC-RL controller.

\[ u = \hat{\Theta}(t) \Phi(t) \quad (23) \]

\[ \dot{\hat{\Xi}} = -\gamma B^T P e \Phi^T \quad (24) \]

\[ \dot{\hat{\Theta}} = -\beta (\hat{\Theta} - \Xi) N(t) \quad (25) \]

where

\[ N(t) = 1 + \mu \Phi^T \Phi \quad (26) \]

\[ \mu \geq \frac{2\gamma}{\beta} \| PB \|_F \| \Omega \|_F^2 \quad (27) \]

\( \Omega \) is such that \( \Omega^T \Omega = \Lambda_i \| \cdot \|_F \) in (27) denotes the Frobenius matrix norm, and \( P = P^T \in \mathbb{R}^{n \times n} \) is a positive-definite matrix that solves the equation \( A^T H^2 + P A H = -Q_H \), where \( Q_H \) is a positive-definite matrix and \( Q_H \geq 2I \). The following theorem is the first main result of this article.

**Theorem 4:** Under Assumptions 1–4, the closed-loop adaptive system specified by the plant in (20), the reference system in (18), and the adaptive controller in (23)–(25) leads to bounded solutions for any initial conditions \( x(0) \), with \( \lim_{t \to \infty} e(t) = 0 \).

Using the result in Theorem 4, we quantify the asymptotic property of AC-RL controller in Corollary 1 where \( o(\cdot) \) is defined as in [63]. For this purpose, we define the ideal control input as \( u^* = \Theta \Phi \). From (21), it is easy to see that if \( u = u^* \), then \( x(t) \to x_r(t) \) as \( t \to \infty \).

**Corollary 1:** If \( \hat{\Phi} \) is bounded, the AC-RL controller ensures that (i) \( \lim_{t \to \infty} |u(t) - u^*(t)| = 0 \) and

\[ \int_{t_0}^{t_0+T} \| c(x(t), u(t)) - c(x_r(t), u^*(t)) \| \, dt = o(T) \quad . (28) \]

We make several important observations about the AC-RL controller, given as follows.

1) The parameter update in (24) and (25) is a second-order tuner, and an extension of our earlier results in [41] to the multivariable case. Two different regressors are employed in the AC input, \( \Phi_r \) in (23) and \( \Phi_n \), which are utilized to address the two different sources of parametric uncertainties \( A \) and \( \Theta_n \). The first regressor component, \( \Phi_r \), comes predominantly from the RL-component, whereas the second regressor accommodates the uncertainty \( \Theta_n \) in \( g(x) \) (and employs Assumption 3).

2) The RL-policy is required to satisfy Assumption 2, such as the one in (23).

3) In comparison to a purely AC approach, in the AC-RL controller the RL plays the role of a reference model that is nonlinear with a controller that is nonlinear as well and trained so as to elicit desirable properties from the reference system. It should also be noted that a purely AC approach is agnostic to \( u_r \), relegates the design of \( u_r \) to an outer loop, and essentially focuses only on the inner loop stability and tracking of \( x_r(t) \).

4) If there are no parametric uncertainties, and if the initial conditions of (20) are identical to those of (18), then the choices of \( \Theta(0) = [I, I_{n \times r}]^T \) and \( \Xi(0) = 0 \) ensure that the AC-RL controller \( u(t) \to u^*(t) \to u_r(t) \), thereby accomplishing (4). That is, the AC-RL policy coincides with the near-optimal RL-based policy. When there are parametric uncertainties, the AC-RL policy is necessarily modified; \( u(t) \to u^*(t) \) but \( u^*(t) \neq u_r(t) \). Theorem 4 guarantees that with this modified policy, the closed-loop state \( x \) is globally bounded, sans any online training, and converges to \( x_r \). Corollary 1 precisely quantifies the corresponding cost of the AC-RL controller.

**Remark 4:** Alternate methods rather than RL for determining the feedback policy may be used as well provided they satisfy Assumption 2. Methods based on MPC may be used as well, but may require additional structures on the cost function in the form of terminal costs [64]. The same comment is applicable for other classical nonlinear control methods, but these may involve additional assumptions or restrictions. Furthermore, MPC generally requires online computation, which might be challenging especially when system dynamics is nonlinear, while RL can be trained offline and deployed online with reduced computational burden.

### C. Comparison of AC-RL and RL-Based Controllers

We make a few comments regarding Assumptions 1–4 and the global properties of this AC-RL controller in the following.

1) **Global Properties of the AC-RL Controller versus Local Properties of the RL Controller:** The focus is on nonlinear dynamic systems with parametric uncertainties that satisfy Assumptions 1, 3, and 4 and an RL-component that satisfies Assumption 2. Suppose we start with \( x(t_0) \) with \( |x(t_0)| \leq R_1 \). Suppose that the parametric uncertainties are such that \( x(t) \) departs from \( X \) for some \( t > t_0 \). In such a case, an RL controller alone will not suffice. In contrast, the AC-RL controller will accommodate these departures, modify the policy from \( u_r(t) \) to \( u^*(t) \), and ensure that

\[ |x(t)| \leq (1/\lambda(P_min)) V_0 + R_1 \quad \forall t \geq t_0 \]

where \( V_0 = V((e(t_0), \Xi(t_0) - \Theta(t_0) - \Theta)) \), where \( V \) is defined as in (59). This accommodation was possible because of the change in the control input from \( u_r \) to \( u \) in (23). It should be noted that the AC-RL controller relaxes the restriction of the RL-approach that the state always remains in a compact set, allows parametric uncertainties to be introduced at any \( t_0 \), provided they are in the form specified in Assumptions 1, 3, and 4, and provided the RL approach that satisfies Assumption 2. As this is a nontrivial class of nonlinear problems that can occur in practice, this extension from a local result with RL approaches to a global result with AC-RL approach proposed is an important contribution, and bridges the sim-real gap for this class of nonlinear problems. The benefit of using RL rather than other control approaches is that we can address tasks that correspond to arbitrary cost functions. The cost of using RL is the requirement that Assumption 2 is satisfied.

2) **Assumptions 1–4:** Assumption 2 was needed for the RL part of the controller, whereas the other three assumptions were needed for the AC part of the controller. As the adaptive
controller is designed to function in real time, certain structures needed to be imposed on the type of uncertainties that the AC can deal with, which led to Assumption 1 (that required uncertainties to lie in the span of the control matrix B), Assumption 3 (that required the nonlinearities to be represented using a linear network with a bounded approximation error), and Assumption 4 (which required certain analytic properties in the linear parametric uncertainty $\Lambda$).

3) Optimality of the Control Policy $u^*$: While we have established the asymptotic nature of $u$ in the form of $u^*$, we note that $u^*(t) \neq u_{\text{opt}}(t)$, where

$$u_{\text{opt}}(t) = \arg\min_{u(t) \in \mathcal{U}} \int_{t_0}^{t_0+T} c(x(t), u(t)) dt$$

for the plant (20). The reason for this difference can be argued, thus. The input $u^*$ is designed to make the closed-loop system stable with uncertainties with the reference system as the target response. But such a $u^*$ is optimal with respect to the cost function in (4) subject to the reference system in (18) (i.e., no uncertainties), but not optimal with respect to the perturbed system in (20) (with uncertainties). While choosing the reference system in (18) as a target response provides tractability in the form of global stability, it may not be the right choice for the compromised true system in (20) from the point of view of optimality. A tractable design of a reference system that assures both stability (using AC methods) and optimality (using RL methods) is a topic for future research.

D. Learning in AC-RL Controllers With Persistent Excitation

Section IV-B focused on the control solution, and showed global boundedness with a cost as in (28). In particular, we showed in Corollary 1 that $\Theta(t)\Phi(t) \to 0$ asymptotically, where $\Theta = \Theta - \Theta$. As the vector $\Phi$ can be arbitrary, $\Theta(t)$ can remain orthogonal to $\Phi(t)$, and need not go to zero asymptotically, i.e., there is no guarantee that parameter learning will take place. In this section, we establish learning with the AC-RL algorithm by imposing additional conditions of persistent excitation on $\Phi$. We limit our discussion to the case when $u(t)$ is a scalar.

The starting point is the error equation in (21) and the AC-RL controller in (23)–(27). With a scalar input, $m = 1$, we obtain that $\Lambda \in \mathbb{R}^+$ (using Assumption 4), $\Theta \in \mathbb{R}^{1 \times (l+1)}$, $\Phi(\ell)(t) \in \mathbb{R}$, and $\Phi_{\tau}(t) \in \mathbb{R}^l$ in (22).

Defining $\hat{\theta} = (\hat{\Theta} - \Theta)^T$, $\hat{\dot{\theta}} = (\hat{\Sigma} - \Sigma)^T$, and $B_0 = B\Lambda$, we rewrite (21), (24), and (25) as

$$\dot{e} = AHe + B_0\hat{\theta}^T \Phi$$

and

$$\dot{\hat{\theta}} = -\gamma \dot{\theta} e^T PB$$

$$\dot{\hat{\theta}} = -\beta (\hat{\theta} - \hat{\theta}) N_t$$

where $N_t = 1 + \mu \Phi^T \Phi$ and $\mu \geq 2\gamma\|PB\|^2 / \beta$. As before, the matrix $P$ solves $AP + PA_H = -Q$, and $Q \geq 2I$ is a symmetric, positive-definite matrix. The constants $\gamma$ and $\beta$ are positive. Let $x_1(t) = [e(t)^T, (\hat{\theta} - \hat{\theta})^T]^T$ and $z(t) = [x_1(t)^T, \hat{\theta}(t)^T]^T$.

We note that the stability result related to the AC-RL controller stated in Theorem 4 guarantees that $z(t)$ is uniformly bounded, which follows from a Lyapunov function of the form:

$$V = \frac{\Lambda}{\gamma} \hat{\theta}^T \Phi + \frac{\Lambda}{\gamma} [(\hat{\theta} - \hat{\theta})^T (\hat{\theta} - \hat{\theta})] + e^T Pe$$

whose derivative is bounded by

$$\dot{V} \leq -2\beta\Lambda \|\hat{\theta} - \hat{\theta}\|^2 - \Lambda \|e\|^2.$$  

Noting that our goal is to show that $\lim_{t \to \infty} \hat{\theta}(t) = 0$, it suffices to show that $V(t) \to 0$ as $t \to \infty$. As the time derivative $\dot{V}$ in (33) is only negative semidefinite, additional conditions of persistent excitation are utilized to show parameter learning, and correspond to the second contribution of this article.

The following lemmas are useful in proving the second contribution of this article, which is stated in Theorem 5. We note that Theorem 4 guarantees that $z(t)$ and $\Phi(x, e, u_\tau)$ are bounded, making the properties of persistent excitation applicable for what follows.

Lemma 1: Let $\epsilon_1 > \epsilon_2 > 0$, then there is an $n = n(\epsilon_1, \epsilon_2)$ such that if $z(t) = [x_1(t)^T, \hat{\theta}(t)^T]^T$ is a solution with $\|z(t_1)\| \leq \epsilon_1$ and $S = \{ t \in [t_1, \infty) | \|z(t_1)\| > \epsilon_2 \}$, then $\mu(S) \leq n$, where $\mu$ denotes Lebesgue measure.

Lemma 2: Let $\delta > 0$ and $\epsilon_1 > 0$, then there exist positive numbers $\epsilon$ and $T$ such that if $z(t) = \{ z(t_1) \}$ and $\|z(t_1)\| \leq \epsilon_1$ and $\|\hat{\theta}(t_1)\| > \delta$ for $t \in [t_1, t_1 + T]$, then there exists a $t_2 \in [t_1, t_1 + T]$ such that $\|\hat{\theta}(t_2)\| > \delta$.

Theorem 5: If $\Phi(t)$ satisfies the persistent excitation property in (27), then the origin $(e = 0, \hat{\theta} = 0)$ in (29)–(31) is uniformly asymptotically stable.

The proof of Theorem 5 stems from the three lemmas listed previously, which represent the three main steps. Lemmas 1 and 2 establish that $x_1(t)$ cannot remain small over the entire period of persistent excitation. Lemma 3 then leverages this fact to show that this leads to the parameter error $\hat{\theta}(t)$ to decrease. Together this allows the conclusion of uniform asymptotic stability of all the origin $(29)–(31)$ and, that $\lim_{t \to \infty} \hat{\theta}(t) = 0$. As is apparent from the details of the proof provided in the Appendix, first principles-based arguments had to be employed in order to derive this result. No standard observability properties or time-scale transformations as in [33] have been employed; these are inadequate as the error model structure in (29) includes system dynamics and no filtering techniques are used to convert the error model to a static linear regression model. Extensions to multivariable inputs are a topic for future work. The results of this section also indicate that the RL-based policy $u_\tau$ should be such that various components of $\Phi(t)$ satisfy the PE condition.

V. EXTENSIONS OF THE AC-RL CONTROLLER FOR PROBLEM 1

A. Relaxation of Assumption 3

Assumption 5: The higher order term in (3) is parameterized linearly, i.e., $g(x) = \Theta_n, r(x) = d(t)$ where $\Theta_n, r \in \mathbb{R}^{m \times l}$,
\( \Phi_n(x) \in \mathbb{R}^l \), and \( d(t) \) denotes an approximation error with \( \|d(t)\| \leq d_{\text{max}} \) for all \( t \).

**Remark 5:** While Assumption 5 is a relaxation of Assumption 3, the more general case of interest is when \( d \) to be dependent on the state \( x \), i.e., not known to be bounded a priori. This extension is relegated to future work.

We note that with Assumption 3 replaced by Assumption 5, (20) is modified as
\[
\dot{x} = Ax + BA\left[ u + \Lambda^{-1}\Theta_s\Phi_n(x) + \Lambda^{-1}d \right] \tag{34}
\]
with the resulting error equation modified from (21) as
\[
\dot{e} = AHe + BA[u - \Theta\Phi + \bar{d}] \tag{35}
\]
where \( \bar{d} = \Lambda^{-1}d \) with \( \|\bar{d}\| \leq \bar{d}_{\text{max}} \).

The AC-RL controller in this case remains as in (23) and (25), but the tuner in (24) is modified as
\[
\dot{\hat{z}} = -\gamma B^TPe\Phi^T - f_1(\hat{z}, \hat{\Theta}) \tag{36}
\]
\[
f_1 = \left(2\hat{z} - \hat{\Theta}\right) \tag{37}
\]
with all other quantities defined as in Section IV. The analytical guarantee for this AC-RL controller is summarized in the following theorem.

**Theorem 6:** Under Assumptions 1, 2, 4, and 5, the closed-loop adaptive system specified by the plant in (20), the reference system in (18), and the adaptive controller in (23)–(27), and (24) replaced by (36) and (37), leads to globally bounded solutions, with \( e(t) = O(D) \), where \( D = \max(\bar{d}_{\text{max}}, \|\Theta\|) \).

**B. Magnitude Saturated Adaptive Controller (MSAC)-RL Controller**

As control inputs are often subject to magnitude saturation, we propose another extension, magnitude saturated adaptive controller (MSAC)-RL, which builds on the ideas introduced in [16]. The saturated control input into the true plant is calculated as
\[
u_i(t) = u_{i,\text{max}} \otimes \frac{u_{i,c}(t)}{u_{i,\text{max}}} \tag{38}
\]
where \( u_{i,c}(t) \) denotes the output of the controller and \( u_{i,\text{max}} \) is the allowable magnitude limit on \( u_i \), chosen as \( ||U|| \), where \( U \) is the set of all allowable inputs. This induces a saturation-triggered disturbance \( \Delta u \) vector defined by
\[
\Delta u(t) = u_{i,c}(t) - u(t) \tag{39}
\]
It is easy to see that \( \Delta u(t) = 0 \) when the desired control \( u_{i,c}(t) \) does not saturate. The output of the controller, \( u_i \), is given by
\[
u_i = \hat{\Theta}(t)\Phi(t) \tag{40}
\]

The presence of the disturbance \( \Delta u \) causes the error equation to vary from (21) to
\[
\dot{e} = AHe + BA\left[u_{i,c} - \Delta u - \Theta\Phi\right] \tag{41}
\]
We introduce a new performance error \( e_a \) in order to accommodate the disturbance \( \Delta u \) as follows:
\[
\dot{e}_a = AHe_a + BK_a(\Delta u) \tag{42}
\]
where \( K_a \) is an estimate for \( \Lambda \), which is adjusted according to (44). The error \( e_a \) leads to a new augmented error \( e_a = e - e_a \)
\[
\dot{e}_a = AHe_a + B\hat{\Theta}(t)\Phi(t) - B(\Lambda + K_a)\Delta u \tag{43}
\]
This suggests a different set of adaptive laws
\[
\dot{\hat{z}} = -\gamma B^TPe\Phi^T \tag{44}
\]
\[
\dot{\hat{\Theta}} = -\beta(\hat{\Theta} - \Xi)\mathcal{N}_t \tag{45}
\]
where \( \Xi = [\hat{\Theta}, -K_a] \) and \( \Phi = [\Phi^T, \Delta u^T]^T \), \( \gamma \) and \( \beta \) are positive constants, and \( \mathcal{N}_t \) defined as in (26) with \( \Phi \) replaced by \( \hat{\Theta} \) and \( \mu \) as in (27). The following theorem provides the analytical guarantee for this MSAC-RL controller.

**Theorem 7:** Under Assumptions 1–4, the closed-loop adaptive system specified by the plant in (20), the reference system in (18), the magnitude constraint in (38), and the MSAC-RL controller given by (40), (44), and (45) leads to

i) globally bounded solutions, with \( \lim_{t \to \infty} ||e(t)|| = 0 \) if the target system in (20) is open-loop stable;

ii) bounded solutions for all initial conditions \( x(0), \Xi(0), \hat{\Theta}(0) \), and \( K_a(0) \) in a bounded domain, with \( ||e(t)|| = O(||f^T||\Delta u||d\tau|| \) if the target system in (18) is not open-loop stable.

**Remark 6:** As in Section V-A, if we relax Assumption 3 in the form of Assumption 5, then the asymptotic result in Theorem 7(i) is replaced by convergence to a compact set as in Theorem 6. It should be noted that in both Sections V-A and V-B, the proposed controller still remains global.

**C. Multiple Equilibrium Points**

Suppose the system in (1) has \( p \) equilibrium points \( (X_1, U_1), \ldots, (X_p, U_p) \), so that \( F(X_i, U_i) = 0 \) for \( i = 1, \ldots, p \). Define \( x_i = x - X_i, u_i = u - U_i \). Denoting the state and action spaces as \( \mathcal{X} \) and \( \mathcal{U} \), respectively, we partition the composite domain \( \mathcal{D} = \mathcal{X} \times \mathcal{U} \) into \( p \) disjoint subsets \( S_1, \ldots, S_p \), such that
\[
\mathcal{D} = S_1 \cup S_2 \cup \ldots \cup S_p
\]
where \( S_i \cap S_j = \emptyset, i \neq j \).

One can then express (1) in the presence of parameter uncertainties as
\[
\dot{x} = \begin{cases}
A_1x_1 + B_1[\Lambda u_1 + g(x_1)], & x_1, u_1 \in S_1 \\
A_p x_p + B_p[\Lambda u_p + g(x_p)], & x_p, u_p \in S_p
\end{cases} \tag{46}
\]
If we now consider the effect of parameter uncertainties in \( u_i \) and \( g(x_i) \) as in Section III-A, it is easy to have a switching set of controllers as in (23)–(25) that are invoked when the trajectories enter the set \( S_i \). A corresponding stability result to Theorem 4 can be derived, provided the command signal is such that the dwell time in each set \( S_i \) exceeds a certain threshold, which is not discussed here. We summarize the overall AC-RL controller in Algorithm 1, which specifies the discrete time implementation of the overall AC-RL controller. \( \Delta t \) denotes the integration timestep and the AdaptiveControl function corresponds to the AC input; in the single equilibrium case, this function corresponds to (23)–(27). \( F \) and \( F_r \) represent the velocity vector fields of the target and reference dynamical systems, respectively. That is, \( F \) and \( F_r \) are the right-hand sides of (2) and (6), respectively.

**D. Nonaffine Systems**

We once again start with (1), expand the dynamics around \( (X_0, U_0) \), which together with Assumption 1 leads to a class of
Algorithm 1: Multiple Equilibrium AC-RL.

Require: $F, F_r, \pi, x_r, x$

while running do
  $u_r = \pi(x_r)$
  $x_i \leftarrow x - X_i$
  $u_i \leftarrow u - U_i$
  $e \leftarrow x_i - (x_r - X_i)$
  $\Phi \leftarrow \Pi(x_i, u_i)$
  $u \leftarrow \text{AdaptiveControl}(\Phi, e, \Theta_i, P_i, B_i) + U_i$
  $x_{r, i} \leftarrow x_r + F_r(x_r, u_r) \Delta t$
  $x \leftarrow x + F(x, u) \Delta t$
  $i \leftarrow j, x, u \in S_j$
end while

nonlinear systems

$$\dot{x} = Ax + B[u + h(x, u)].$$  \hspace{1cm} (47)

Noting that $u$ is required to minimize (44), we assume that $u$ can be expressed as an analytic function of the state $x$, which in turn leads to the assumption that

$$h(x, u) = g(x).$$  \hspace{1cm} (48)

The goal is to determine $u$ in real time when parametric uncertainties are present in (47) and (48) so that (4) is accomplished for initial condition $x_0$. As before, the reference system dynamics with no parametric uncertainties is chosen as in (18). Noting that (48) makes the plant in (47) identical to (3), the same AC-RL controller in Section IV-B leads to global boundedness under Assumptions I–IV, with valid extensions as in Sections V-A and V-B. We skip a formal statement of the result due to space limitations.

VI. INTEGRATED AC-RL SOLUTIONS: PROBLEM 2

We now consider Problem 2, which pertains to the class of dynamic system (5), and introduce parametric uncertainties $\theta \in \mathbb{R}^{n_{\theta}}$ and $\nu \in \mathbb{R}^{n_{\nu}}$ in $f(X)$, and $A \in \mathbb{S}^n_{+}$ in $U$, which may be due to loss of effectiveness (LOE), leading to the system

$$\dot{X}(t) = f(X) - \rho(X)\nu + B(X) [AU(t) - \phi(X)\theta]$$  \hspace{1cm} (49)

where $\rho(X) \in \mathbb{R}^{n \times n_{\nu}}$ and $\phi(X) \in \mathbb{R}^{m \times n_{\theta}}$ are known nonlinear dynamics of the parametric uncertainties $\theta$ and $\nu$, respectively. It is clear that $\phi(X)\theta$ denotes a matched uncertainty similar to that introduced by Assumptions 3, whereas $\rho(X)\nu$ denotes an unmatched uncertainty. We make the following assumption regarding the latter, which is similar to an extended matching condition in [5].

Assumption 6: The unmatched uncertainty satisfies $\rho(X)\nu \in \text{span}\{ad_j B\}$, where $ad_j B = [f, B] = \frac{\partial}{\partial x} B - \frac{\partial B}{\partial x} f$ denotes the Lie bracket of $f, B$, $f(X)$ and $B(X)$ are nonlinear functions such that $B$ and $ad_j B$ are linearly independent.

For the system in (49) with Assumption 6, we now propose an AC-RL controller. We first consider a control-affine reference system, similar to (5), of the form

$$\dot{X}_r(t) = f(X_r) + B(X_r)U_r(t).$$  \hspace{1cm} (50)

Given an initial condition $X_0$, $U_r(t)$ minimizes the cost function given in (4) where $x(t)$ and $u(t)$ are replaced by $X(t)$ and $U(t)$, respectively. The reference input $U_r(t)$ can be constructed using the offline RL-based approach in Section III-A. In addition to Assumptions 2 and 4, we introduce the following assumption.

Assumption 7: A uniformly bounded parameter-dependent contraction metric $M(X, \nu)$ can be computed for each parameter estimate $\hat{\nu} \in \mathbb{R}^{n_{\nu}}$.

Under Assumption 7, let $\gamma(t, s)$ be the geodesic that connects $X_r(t)$ to $X(t)$, i.e., $\gamma(t, 0) = X_r(t)$ and $\gamma(t, 1) = X(t)$. The Riemannian energy of the geodesic is defined as $E = \int_0^1 \gamma_t^T M(X, \nu) \gamma_t ds$. We now describe the AC-RL controller in (51)–(53) and the stability result

$$U = \Phi \left( U_c + U_r + \phi(X)\theta \right)$$  \hspace{1cm} (51a)

$$U_r = \sum_{i=1}^{n_r} \nu_i \int_0^1 \left[ \frac{\partial \rho_i^T(\gamma(s))}{\partial X_1} \ldots \frac{\partial \rho_i^T(\gamma(s))}{\partial X_{n-1}} \right] \gamma(s) ds$$  \hspace{1cm} (51b)

$$\dot{\lambda} = \beta_\nu (\nu_a - \hat{\nu}) \mathcal{N}_\nu$$  \hspace{1cm} (51c)

$$\dot{\lambda} = \beta_\theta (\theta_a - \hat{\theta}) \mathcal{N}_\theta$$  \hspace{1cm} (51d)

$$\dot{\hat{\lambda}} = \beta_{\phi} (\Psi_a - \Psi) \mathcal{N}_{\Psi\phi}$$  \hspace{1cm} (51e)

$$\dot{\hat{\lambda}} = \beta_{\psi} \mathcal{N}_{\Psi\psi}$$  \hspace{1cm} (51f)

$$\dot{\Psi} = \beta_{\psi} (\Psi_a - \Psi) \mathcal{N}_{\Psi\phi}$$  \hspace{1cm} (51g)

where

$$\mathcal{N}_\nu = 1 + \omega_\nu^T \omega_\nu, \quad \omega_\nu = \rho(X)^T M \gamma_\nu(1)$$  \hspace{1cm} (52a)

$$\mathcal{N}_\theta = 1 + \omega_\theta^T \omega_\theta, \quad \omega_\theta = \phi(X)^T B(X) M \gamma_\theta(1)$$  \hspace{1cm} (52b)

$$\mathcal{N}_{\Psi\phi} = 1 + \omega_{\Psi\phi}^T \omega_{\Psi\phi}, \quad \omega_{\Psi\phi} = \psi(X)$$  \hspace{1cm} (52c)

$$\rho_i$ is the $\text{th}$ column of $\rho(X, \nu)$, $\Omega_i \in \mathbb{R}^{m \times \nu}$ extracts the $\text{th}$ column vector of $B(x)$ such that $\rho_i(x)\nu_i \in \text{span}\{ad_j B_i\}$, and $U_c$ is a nominal controller that leverages the RL-based input $U_r$ and satisfies

$$\frac{\partial E}{\partial \hat{\nu}} + 2 \gamma_\nu(1)^T M(X, \nu) [f(x) - \rho(X)^T \nu + B(X)U_c]$$

$$2 \gamma_\theta(0)^T M(X, \nu) [f(x) + B(X)U_c] \leq -2\lambda E.$$  \hspace{1cm} (53)

where $\lambda > 0$. The adaptive gain parameters $\beta_\nu$, $\gamma_\nu$, $\beta_\theta$, $\gamma_\theta$, $\beta_{\phi}$, and $\gamma_\phi$ are chosen as positive numbers that satisfy

$$\frac{\beta_\nu}{\gamma_\nu} \geq \frac{3}{\lambda}, \quad \frac{\beta_\theta}{\gamma_\theta} \geq \frac{3}{\lambda}, \quad \frac{\beta_{\phi}}{\gamma_\phi} \geq \frac{3}{\lambda} ||M|| ||B||$$  \hspace{1cm} (54)

Theorem 8: Under Assumptions 2, 4, 6, and 7, the closed-loop system specified by the plant in (49), and the adaptive controller in (51)–(54) has bounded solutions and ensures that $\lim_{t \to \infty} \|X(t) - X_r(t)\| = 0$.

Remark 7: Theorem 8 proposes a solution to Problem 2 and leverages contraction theory rather than Lyapunov theory. The new contributions here, in comparison with [31], are that the controller is based on AC-RL, it utilizes an HT, and incorporates larger uncertainties, such as LOE.
VII. NUMERICAL STUDIES

A. Problem 1

We consider a control task of a quadrotor moving in 3-D to land on a moving platform. The AC design was based on a linearized quadrotor model:

\[ \begin{align*}
\dot{x} &= g\theta \\
\dot{y} &= -g\phi \quad \dot{\phi} = \frac{L}{I_x} \tau_\phi \\
\dot{z} &= f_z - mg \quad \dot{\psi} = \frac{1}{I_z} \tau_\psi
\end{align*} \]

which defines the matrices \( A, B \). \( \Theta_{l,r} \) was chosen via \( \Theta_{l,r} = B^\dagger(A_r - A_H) \) such that all eigenvalues of \( A_H \) were at \(-6\), and we also chose \( Q_H = I \). The generalized forces were generated via the control input as

\[ \begin{bmatrix} f_z \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ L & 0 & -L & 0 \\ 0 & L & 0 & -L \\ \nu & -\nu & \nu & -\nu \end{bmatrix} \Lambda u. \] (56)

In (55), \( g = 9.81, m = 1.2, L = 0.3, I_x = I_y = 0.22, I_z = 0.44 \), and \( \nu = 1.0 \) (in SI units). \( \Lambda \) was chosen to be diagonal with entries decreased from the nominal case of unity to represent LOE of each actuator. A linearized quadrotor model as in [65], however, was used for the training of the RL as well as for the evaluation of the AC-RL controllers.

The cost function in (4) was chosen as

\[ c(\bar{x}, t) = \begin{cases} -1 & \text{box} \\ 1 & -\text{box} \land (\Delta z \leq 0 \lor t \geq T_{\text{max}}) \end{cases} \] (57)

where the Boolean variable \( \text{box} \) is assumed to be true if all of the following simultaneously hold: \( |x| \leq 5 \text{ cm}, |y| \leq 25 \text{ cm}, |z| \leq 10^5, |\theta| \leq 10^\circ, |\phi| \leq 50 \text{ cm/s}, |\omega| \leq 10 \text{ cm/s}, \) and \( T_{\text{max}} = 10 \text{ s} \), where \( |\Delta xy| \leq 25 \text{ cm} \), \( |\phi| \leq 10^\circ \), and \( Q_H = I \). The cost increases if the quadrotor leaves a compact set \( D(\bar{x}) \), if its altitude falls below the platform altitude, or if the termination time exceeds 10 s.

We use the popular PPO method [66] to determine the RL component even though PPO has been shown only to have properties of local optimality [59], as it provides a powerful methodology for training. Any departures of the state from the compact set are accommodated through suitable reward engineering, i.e., through the addition of high-magnitude penalties in the cost function only on out-of-set excursions. The following hyperparameters are used in the training of the RL algorithm, PPO (see [53] and [66] for details), in the outer loop: \( \gamma = 0.99, \Delta t = 0.1, T = 1000 \), and \( N = 64 \). The fourth-order Runge–Kutta method is used to integrate the dynamics, with a stepsize of \( h = 0.001 \). The learning rate used for stochastic gradient descent (SGD) algorithms is 0.001, and both the value and policy networks were chosen to have two hidden layers, each with 128 neurons and rectified linear unit (ReLU) activations. The inner loop controller uses the MSAC algorithm described in Section V-B, with \( \gamma = 50, \beta = 25 \), and \( \mu = 1 \).

We test two types of parametric uncertainties: 1) the mass, length, and inertia properties of the quadrotor varied between \( \pm 25\% \) of their nominal (reference) values (see Table I), and 2) an abrupt LOE in the fourth propeller (see Table II) and, therefore, in \( \Lambda \). Such a LOE may occur if the propeller blades are broken midflight, as demonstrated in [65]. Both types of parametric uncertainties correspond to the target–system structure as in (20), with the symmetric part of \( \Lambda \) being positive definite. We observed that in all cases, the AC-RL ensured boundedness of all signals in the closed-loop system and that the tracking error converged to zero, which validates Theorem 4.

As our goal extends beyond Theorem 4 to minimizing the cost in (57), we evaluate AC-RL using additional performance metrics, success rate (SR), and success time (ST), where SR denotes the percentage of all trials where the quadrotor entered the compact set \( D(\bar{x}) \) within 10 s, and ST is the mean time required to complete this task over all successful tests. The results are reported in Tables I and II, with Table I also including a comparison of the average cost in (57) averaged over 1000 trials. It should be noted that Theorem 4 provides no guarantees for SR to be at 100\% or for ST to be less than a specified amount, as both correspond to the transient performance of a nonlinear controller, which is very difficult to quantify. Nevertheless, the results reported in Table I and II show that AC-RL outperforms its competitors.

We compare four control methods:

1) RL, which just uses the trained \( \pi \) to dictate control;
2) AC-RL, which utilizes both \( \pi \) and the adaptive laws in Section V-B;
3) DR-RL, which utilizes both \( \pi \) and the adaptive laws in Section V-B;
4) ME-RL, which utilizes both \( \pi \) and the adaptive laws in Section V-B.

| Algorithm | SR | ST | Cost (57) |
|-----------|----|----|----------|
| RL        | 48%| 7.5s| 1.2      |
| AC-RL     | 82%| 3.5s| -0.3     |
| DR-RL     | 75%| 7.1s| 0.11     |
| ME-RL     | 88%| 3.4s| -0.41    |

| Algorithm | Result | SR | ST | LOE |
|-----------|--------|----|----|-----|
| AC-RL     |        | 97%| 97%| 0%  |
| ME-RL     |        | 88%| 74%| 10% |
| RL        |        | 73%| 34%| 25% |
| ST        |        | 54%| 14%| 50% |
| ST        |        | 27%| 0% | 75% |
| LOE       |        | 2.6s| 2.8s| 2.6s| 0%  |
| ST        |        | 3.1s| 3.1s| 3.1s| 10% |
| ST        |        | 2.9s| 4.7s| 4.7s| 25% |
| LOE       |        | 3.3s| 5.4s| 5.4s| 50% |
| ST        |        | 3.3s| -- | --  | 75% |

| Algorithm | Result | SR | ST | LOE |
|-----------|--------|----|----|-----|
| AC-RL     |        | 97%| 97%| 0%  |
| ME-RL     |        | 88%| 74%| 10% |
| RL        |        | 73%| 34%| 25% |
| ST        |        | 54%| 14%| 50% |
| ST        |        | 27%| 0% | 75% |
| LOE       |        | 2.6s| 2.8s| 2.6s| 0%  |
| ST        |        | 3.1s| 3.1s| 3.1s| 10% |
| ST        |        | 2.9s| 4.7s| 4.7s| 25% |
| LOE       |        | 3.3s| 5.4s| 5.4s| 50% |
| ST        |        | 3.3s| -- | --  | 75% |

| Algorithm | Result | SR | ST | LOE |
|-----------|--------|----|----|-----|
| AC-RL     |        | 97%| 97%| 0%  |
| ME-RL     |        | 88%| 74%| 10% |
| RL        |        | 73%| 34%| 25% |
| ST        |        | 54%| 14%| 50% |
| ST        |        | 27%| 0% | 75% |
| LOE       |        | 2.6s| 2.8s| 2.6s| 0%  |
| ST        |        | 3.1s| 3.1s| 3.1s| 10% |
| ST        |        | 2.9s| 4.7s| 4.7s| 25% |
| LOE       |        | 3.3s| 5.4s| 5.4s| 50% |
| ST        |        | 3.3s| -- | --  | 75% |
3) a metalearning approach (ME-RL) which employs a deep learning-based policy update in addition to a network that learns all hyperparameters [67];
4) a domain-randomized approach (DR-RL) [68], which uses domain randomization of the plant parameters during policy training.

The observations are given as follows.
1) AC-RL performs better than pure RL or DR-RL (see Table I).
2) ME-RL outperforms AC-RL on all three metrics. This comes with two qualifiers: a) the metalearner in ME-RL is trained using the same distributional shift on which it was tested (the ±25% parameter perturbations), and b) the metalearner is a DNN and, thus, has no guarantees of convergence, unlike AC-RL.
3) The relevance of point 1) becomes apparent from the ME-RL results in Table II, where it can be see that ME-RL greatly underperforms AC-RL. This is because the specific type of uncertainty (asymmetric LOE) studied in Table II was not incorporated into the metalearner’s training regimen. The LOE column represents the degree of propeller thrust lost (with 0% being no loss). For a 75% LOE there is no data on the RL or ME-RL success time because there were no successful tests. See [53] for results with noise.
4) Table I also shows that for a 25% parametric uncertainty, the average cost was −0.3 and 1.2 for AC-RL and RL, respectively. In comparison, the average cost for the case when there were no uncertainties, was observed to be −0.51 for a pure RL approach, which is the best achievable cost, i.e., AC-RL has an optimality gap of 40%.
5) We observed that pure AC led to success only with a smaller set of initial conditions for the state, primarily due to the stringent metric in (57). This led to an optimality gap of 40% under AC-RL, in contrast to 115% under AC alone (see [53] for further details).

B. Problem 2
To demonstrate the effectiveness of our proposed AC-RL controller in (51)–(54), we provide numerical simulation results for a system described in (49) with

\[
X = \begin{bmatrix} X_1 \\ X_2 \end{bmatrix}, \quad f(X) = \begin{bmatrix} -X_1 - X_3^3 + X_2^2 \\ 0 \end{bmatrix}, \quad B(X) = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad \phi(X) = X^T, \quad \rho(X) = \begin{bmatrix} 0 \\ 0 \end{bmatrix}^T, \quad \Lambda = I, \quad \theta = [1 \ 1]^T.
\]

Note that unmatched uncertainty \( \rho(X) \nu \) in (49) is removed for simplicity in the numerical example. As a result, the contraction metric \( M(X) \) does not depend on parameter estimate \( \hat{\nu} \). The initial estimates \( (\hat{\theta}, \hat{\theta}_0) \) and \( (\hat{\Psi}, \hat{\Psi}_0) \) for unknown parameters \( \theta \) and \( \Lambda \), respectively, are 20% off of true values. As shown in Fig. 1, our proposed controller provides bounded closed-loop solution \( X(t) \) that converges to reference state trajectory \( X_r(t) \) generated using RL-based controller

\[
U_r \text{ asymptotically, which validates theoretical results stated in Theorem 8.}
\]

VIII. CONCLUSION
This article proposed solutions for real-time control and learning in dynamic systems with parametric uncertainties using a combination of AC and RL. Two classes of nonlinear dynamic systems are considered, both of which are control-affine. The first class of dynamic systems utilizes equilibrium points and expansion forms around these points and employs a Lyapunov approach. The second class of nonlinear systems uses contraction theory as the underlying framework. For both classes of systems, the AC-RL controller is shown to lead to online policies that guarantee stability, and leverage accelerated convergence properties using an HT. In both cases, when there are no uncertainties, the RL component is assumed to be capable of generating policies that are guaranteed to be stable and near-optimal. In addition, for the first class of systems, the AC-RL controller is shown to lead to parameter learning with persistent excitation. Together, this article takes a first step toward real-time control using machine learning with provable guarantees, by drawing upon key insights, tools, and approaches developed in these two disparate and powerful methodologies of AC and RL. Several more steps need to be carried out in this direction and address the gap between stability and optimality of general nonlinear systems with arbitrary cost functions.

APPENDIX

A. Proof of Theorem 1
A Lyapunov function candidate is chosen as

\[
V(e, \Theta) = e^T P e + \text{Tr} \left( \Theta^T (\Lambda^T S \Theta) \right)
\]

where \( S = \Gamma^{-1} \) in (11) with the symmetric part of \( \Lambda S \) positive definite. Using arguments as in [69], we can show that \( \dot{V} = -e^T Q e \) by using (9) and (11). Thus, \( e(t) \) and the parameter estimates \( \hat{\Theta} \) are uniformly bounded. If, in addition, \( \Phi(t) \) is bounded, we note that \( \dot{e}(t) \) is bounded, and that \( e \in L_2 \). It follows from Barbalat’s lemma that \( \lim_{t \to \infty} \| e(t) \| = 0 \).

B. Proof of Theorem 4
The closed-loop system dynamics with the AC-RL controller is represented by the error equation (21), the AC input (23), and
the adaptive laws in (24)–(27). A Lyapunov function candidate
\[
V = \frac{1}{\gamma} \text{Tr}[\Xi T(\Xi - \Theta)] + \frac{1}{\gamma} \text{Tr}[\Xi (\Xi - \Theta)] + e^T Pe
\]
yields a time derivative
\[
\dot{V} = - \text{Tr}[\Xi T(\Xi - \Theta)] + \frac{1}{\gamma} \text{Tr}[\Xi (\Xi - \Theta)] + e^T Pe
\]
Through algebraic manipulations, it can be shown that
\[
\dot{V} \leq - \frac{2\beta}{\gamma} \text{Tr}[\Xi (\Xi - \Theta)] + 2\|e\|^2
\]
where \(\Xi\) is a positive-definite matrix and \(\|\cdot\|_2\) is the Frobenius matrix norm, and \(\Xi (\Xi - \Theta)\) is the Frobenius matrix norm.

C. Proof of Corollary 1
From Theorem 4, it follows that \(\lim_{t \to \infty} e(t) = 0\). Using (21), it follows that
\[
\lim_{t \to \infty} \int_{t_0}^{t} \exp(AH(t - \tau))BA[u(\tau) - \Theta(\tau)]d\tau = 0.
\]
It also follows that \(\Phi \in L_{\infty}\). In addition, from the proof of Theorem 4, we know that \(\theta, \Xi \in L_{\infty}\). Using (23) and (25) and the supposition that \(\Phi \in L_{\infty}\), we have \(u, \bar{u} \in L_{\infty}\). From here, (62), it can be shown using first principles of real analysis that \(\lim_{t \to \infty} e(t) = 0\), implying conclusion (i). Due to the continuity arguments on \(e(x, u)\), as noted in Section II-A, we have that \(\lim_{t \to \infty} e(x, u) = 0\).

D. Proof of Lemma 1
Note: In the proofs of Lemmas 1–3 and Theorem 5, we assume that \(\|B_0\Phi(t)^T\| \leq c_1\). Such a \(c_1\) exists as \(\Phi\) is bounded following Theorem 4.

Consider the following candidate Lyapunov function:
\[
V = \frac{\Lambda}{\gamma} \|\hat{\theta}\|^2 + \frac{\Lambda}{\gamma} \|\hat{\theta} - \hat{\theta}\|^2 + e^T Pe.
\]
Similar to what has been shown in the proof of Theorem 4, the time derivative of \(V\) may be bounded by
\[
\dot{V} \leq - \frac{2\beta}{\gamma} \|\hat{\theta} - \hat{\theta}\|^2 + \frac{1}{\gamma} \|e\|^2 \leq - c_2 \|x_1\|^2
\]
where \(c_2 = \Lambda \min\{1, \frac{\rho}{\gamma}\}\). Noting that \(P\) is a positive-definite matrix, it follows that for any vector \(v \in \mathbb{R}^n\), \(\alpha, \rho > 0\) exist such that
\[
\alpha e^T v \leq v^T P v \leq \rho e^T v.
\]
We prove Lemma 1 by using contradiction. Assume \(\mu(S) > n\) and denote \(\bar{S} = \{t \in [t_1, \infty)\}\). Integrating \(V\), we have
\[
\int_{t_1}^{\infty} \dot{V}(\tau)d\tau = \int_{S_{t_1}} V(\tau)d\tau + \int_{S_{t_1}} V(\tau)d\tau \leq \int_{S_{t_1}} -c_2 \|x_1(\tau)\|^2 d\tau + \int_{S_{t_1}} \dot{V}(\tau)d\tau \leq -c_2 \|x_1(\tau)\|^2 d\tau - c_2 \|x_1(\tau)\|^2 d\tau.
\]
Choose \(n(\epsilon_1, \epsilon_2) = c_3 \epsilon_1^2 / (c_2 \epsilon_2^2)\). This leads to a contradiction since \(\|x_1(t)\| \leq c_3 \epsilon_1^2\), where \(c_3 = \max\{\frac{\Lambda}{\gamma}, \rho\}\).

E. Proof of Lemma 2
Let \(z(t)\) be a solution with initial condition \(\|z(t_1)\| \leq \epsilon_1\). Suppose that \(\|\hat{\theta}(t)\| \geq \delta\) for all \(t \in [t_1, t_1 + T]\), where \(T = T_0 + \delta_0\).

From the error model in (29), for any \(t \geq t_1\)
\[
e(t + \delta_0) = e(t) + \int_{t}^{t + \delta_0} A_H e(t) + B_0 \Phi(t)^T \hat{\theta}(t)dt
\]
from which we have
\[
\|e(t + \delta_0)\| \geq \|\int_{t}^{t + \delta_0} B_0 \Phi(t)^T \hat{\theta}(t)dt\|
\]
Given \(\epsilon' = \epsilon_0 \delta / (2c_2 \delta_0)\) and \(T = T_0 + \delta_0\), from the adaptive law in (31), there is an \(\epsilon_2 > 0\) such that if \(z(t)\) is a solution to (30) and (31) with \(\|x_1(\tau)\| \leq \epsilon_2\) for all \(\tau \in [t_1, t_1 + T]\), then \(\|\hat{\theta}(\tau) - \hat{\theta}(t_1)\| \leq \epsilon'\). Define \(\epsilon = \min\{\epsilon_2, \frac{\delta_0}{2c_2 \delta_0}, \epsilon_2\}\). Now we show the lemma holds for this choice of \(T\) and \(\epsilon\).

If \(\|x_1(t_2)\| \geq \epsilon\) for some \(t_2 \in [t_1, t_1 + T]\), then we are done. Assume \(\|x_1(t)\| \leq \epsilon\) for all \(t \in [t_1, t_1 + T]\), then
\[
\|e(t) + \int_{t}^{t + \delta_0} A_H e(t)dt\| \leq \epsilon + c_1 \epsilon \delta_0 \leq \epsilon_0 \delta / 4
\]
for all \(t \in [t_1, t_1 + T]\). By hypothesis, there exist a \(t' \in [t_1, t_1 + T]\) such that
\[
\int_{t'}^{t + \delta_0} B_0 \Phi(t)^T wdt \geq \epsilon_0
\]
where \(w = \hat{\theta}(t_1) / \|\hat{\theta}(t_1)\|\) is a unit vector.
We have
\[
\left\| \int_{\tau}^{\tau + \delta_0} B_0 \Phi(\tau)^T \left[ w \right] \beta(\tau) \, d\tau \right\| \\
\leq c_2 \int_{\tau}^{\tau + \delta_0} \left\| \beta(\tau) \right\| \, d\tau \leq c_2 \delta_0 c' = \frac{\epsilon_0 \delta}{2}.
\]
Since \( \left\| e(\tau) \right\| \leq \left\| x_1(\tau) \right\| \leq \epsilon \leq e_2 \) for \( \tau \in [t_1, t_1 + T] \)

\[
\left\| \beta(\tau) \right\| \int_{\tau}^{\tau + \delta_0} B_0 \Phi(\tau)^T w \, d\tau \\
- \left\| \int_{\tau}^{\tau + \delta_0} B_0 \Phi(\tau)^T \beta(\tau) \, d\tau \right\| \leq \frac{e_0 \delta}{2},
\]
which implies
\[
\left\| \int_{\tau}^{\tau + \delta_0} B_0 \Phi(\tau)^T \beta(\tau) \, d\tau \right\| \geq \frac{e_0 \delta}{2} - \frac{e_0 \delta}{4} = \frac{e_0 \delta}{4} > 0.
\]
Thus,
\[
\left\| x_1(\tau + \delta_0) \right\| \geq \left\| x(\tau + \delta_0) \right\| \geq \frac{e_0 \delta}{4} > 0
\]
which is a contradiction.

**F. Proof of Lemma 3**

By Lemma 2, the assumption that \( \left\| z(t_2) \right\| \leq \epsilon_1 \) and \( \left\| \tilde{\beta}(t_2) \right\| \geq \delta \) implies that there exist an \( \epsilon \) such that \( \left\| x_1(t) \right\| \) is periodically both less than \( \epsilon/2 \) and greater than \( \epsilon \). This leads to a contradiction with Lemma 1 if we choose \( \epsilon_1 = \left\| z(t_1) \right\| \) and \( \epsilon_2 = \epsilon/2 \). We, thus, conclude that \( \left\| \beta(t_2) \right\| \leq \delta \).

**G. Proof of Theorem 5**

Consider the candidate
\[
V = \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t) \right\|^2 + 2 \frac{\Lambda}{\gamma} \left\| \tilde{\beta} - \beta(t) \right\|^2 + e^T Pe
\]
with \( \mu \geq 2\gamma \left\| PB \right\|^2 / \beta \) and \( Q_H \geq 2I \), which solves \( A_H \gamma P + PA_H = -Q_H \), the time derivative of (66) may be bounded by
\[
\dot{V} \leq - \frac{2\beta \Lambda}{\gamma} \left\| \beta(t) \right\|^2 - 2 \left\| \beta(t) \right\|^2 \leq 0.
\]
Since \( P \) is positive definite, (63) holds for some \( \alpha, \rho > 0 \).

Now we show that given \( \epsilon_1 > \epsilon_2 > 0 \), there is an \( \eta \) with \( 0 < \eta < 1 \) and \( \Delta T_1 > 0 \) such that if \( \left\| \beta(t) \right\| \) is a solution with
\[
\epsilon_2 \leq \left\| \beta(t) \right\| \leq \epsilon_1, \quad \text{for } t \in [t_1, t_1 + \Delta T_1]
\]
then there is a \( t_2 \in [t_1, t_1 + \Delta T_1] \) such that \( V(t_2) \leq \eta V(t_1) \). Choose \( 0 < \nu < 1, \nu < \sigma < 1, \) and \( \Delta T_2 > 0 \) so that
\[
\rho \sqrt{1 - \sigma} - \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) > 0, \quad \sqrt{(1 - \nu)} - \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) > 0, \quad \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) > 0,
\]
and \( \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) > 0, \) and \( \nu \sigma \sqrt{1 - \sigma} - \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) < 0 \), and \( \nu \sigma \sqrt{1 - \sigma} - \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) < 0 \). From Lemma 3, we can obtain a \( T \) when \( \epsilon = \epsilon_1 \) and \( \delta = \sqrt{c_2} \nu \). Define \( \eta = 1 - \min \{ \Delta T_2 \nu \sigma \sqrt{1 - \sigma} - \Delta T_2 \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) \} \), and \( \Delta T_1 = T + \Delta T_2 \). Next we show that for this \( \eta \) and \( \Delta T_1 \), the results hold.

Let \( t_2' \in [t_1, t_1 + T] \) be such that \( \left\| \tilde{\beta}(t_2') \right\| \leq \delta \sqrt{\gamma} \). If \( V(t_2') \leq \epsilon_2 \), we are done. If \( V(t_2') \geq \epsilon_2 \), then
\[
V(t_2') = \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') \right\|^2 + \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2) \right\|^2 + e(t_2')^T Pe(t_2')
\]
implies
\[
(1 - \nu) V(t_2') \leq V(t_2') - \delta^2
\]
\[
\leq \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2') \right\|^2 + e(t_2')^T Pe(t_2')
\]
\[
\leq \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2') \right\|^2 + e(t_2')^2 \gamma^2. \quad (69)
\]

**Case 1:** \( \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2') \right\|^2 < (1 - \nu) V(t_2') \). From (69)
\[
e(t_2')^2 \geq \frac{1}{\rho}(1 - \sigma) V(t_2')
\]
where \( 0 < \nu \leq \sigma < 1 \). From (29), for any \( t \geq t_2' \)
\[
\left\| e(t_2') - e(t) \right\| \leq \int_{t_2'}^{t} \left\| A_H \epsilon(\tau) + B_0 \Phi(\tau)^T \beta(\tau) \right\| d\tau
\]
\[
\leq \left( \frac{c_1}{\sqrt{\alpha}} + 2c_2 \sqrt{\gamma} \right) (t - t_2') \sqrt{V(t_2')}
\]
which is a contradiction with Lemma 1.

Therefore, \( V(t_2') \leq \eta V(t_2') \) and uniform asymptotic stability holds.

**Case 2:** \( \frac{\Lambda}{\gamma} \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2') \right\|^2 \geq (1 - \nu) V(t_2') \). For any \( t \geq t_2' \), following the process in case 1, we can show that
\[
\left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2') \right\| \leq \left\| \tilde{\beta}(t_2') - \tilde{\beta}(t_2) \right\|
\]
\[
\leq (t - t_2') \left( \beta \sqrt{\gamma} + \frac{c_2 \gamma}{\sqrt{\alpha}} \right) \sqrt{V(t_2')}.
\]
If we let \( t_2 = t_2 + \Delta T_2 \), then
\[
\left\| \hat{\theta}(t) - \hat{\theta}(t) \right\| \geq \left\| \hat{\theta}(t_2') - \hat{\theta}(t_2') \right\| - (t_2 - t_2') \left( \beta \sqrt{\gamma} + \frac{c \gamma \rho}{\sqrt{\alpha}} \right) \sqrt{V(t_2')} \geq \left[ \sqrt{\gamma(1 - \nu)} - \Delta T_2 \left( \beta \sqrt{\gamma} + \frac{c \gamma \rho}{\sqrt{\alpha}} \right) \right] \sqrt{V(t_2')}.
\]

Integrating \( \dot{V} \), we have
\[
V(t_2') - V(t_2) \geq \frac{2 \beta \Delta T_2}{\gamma} \left[ \sqrt{\gamma(1 - \nu)} - \Delta T_2 \left( \beta \sqrt{\gamma} + \frac{c \gamma \rho}{\sqrt{\alpha}} \right) \right]^2 V(t_2').
\]
Therefore, \( V(t_2) \leq \eta V(t_2'), \) which proves the theorem.

**H. Proof of Theorem 6**

When \( d(t) \neq 0 \), with \( f_1 \) as in (37), we obtain additional terms in \( \dot{V} \), in the right-hand side of (60) of the form (for ease of exposition, \( \gamma \) is set to unity)
\[
2e^T PBAd - 2Tr \left( (2\Xi - \hat{\theta})^T (\Lambda + \Lambda^T) (2\Xi - \Theta) \right)
\]
where \( \Xi = \Xi - \Theta \) and \( \Theta = \Theta - \Theta \). Together with these terms, \( \dot{V} \) can be shown to be negative outside an ellipsoid \( \Xi^T \Omega \Xi = D_0 \), where \( \Xi = \{ ||e||, ||\Xi - \Theta||, ||\Xi|| \}^T \), \( \Omega \) is a positive-definite matrix in \( \mathbb{R}^{3 \times 3} \), and \( D_0 \) is a constant that depends on \( d_0 \) and \( ||\Theta|| \) (the constant unknown parameter). We refer the reader to [53] for all technical details of the derivation.

**I. Proof of Theorem 7**

As in Theorem 4, we consider a candidate
\[
V = \frac{1}{\gamma} \text{Tr} \left[ (\Xi - \Theta)^T \Lambda^T (\Xi - \Theta) \right] + \frac{1}{\gamma} \text{Tr} \left[ (\Theta - \Xi)^T \Lambda^T (\Theta - \Xi) \right] + e^T P e_u, \quad (71)
\]
Using (43) and (44), we obtain that \( \dot{V} \leq -e^T P e_u \) since \( Q_H \geq 2I \). This in turn allows us to conclude that \( e_u, \Xi, \Theta \in L_\infty \). For case (i), since all parameters are bounded, together with magnitude saturation, we have that the input \( u \) is bounded. For this case, it implies that the state \( x \) is bounded. Therefore \( e_u \in L_\infty \). Proceeding in a similar fashion to the proof of Theorem 1, Barbalat’s lemma is applied to find that \( \lim_{t \rightarrow \infty} ||e_u(t)|| = 0 \).
It is easy to show then that \( ||e(t)|| = O \left( \int_0^t \| \Delta u(\tau) \| d\tau \right) \).
For case (ii), as the structure of the error model in (43) is identical to that considered in Theorem 1 in [70], the same arguments can be used to establish boundedness of the state.

**J. Proof of Theorem 8**

The first variation of the Riemannian energy [57] is
\[
\dot{E} = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial \hat{\theta}} \dot{\hat{\theta}} + 2(\gamma(s) \dot{\hat{\theta}}(s)) \bigg|_{s=0}^{s=1} - 2 \int_0^1 \left( \frac{D}{Ds} \gamma(s), \dot{\gamma} \right) ds
\]
where \( \frac{D}{Ds} \) is the covariant derivative. Since \( \gamma(s) \) is the geodesic,
\[
\frac{D}{Ds} \gamma(s) = 0.
\]
Under Assumption 6, [31, Lemma 2] implies
\[
\frac{\partial E}{\partial \hat{\theta}} \dot{\hat{\theta}} + 2 \gamma(s) \dot{\hat{\theta}}(s) \bigg|_{s=0}^{s=1} - 2 \int_0^1 \left( \frac{D}{Ds} \gamma(s), \dot{\gamma} \right) ds
\]
Therefore, using (51) and (53), we have
\[
\dot{E} = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial \hat{\theta}} \dot{\hat{\theta}} + 2 \gamma(s) \dot{\hat{\theta}}(s) \bigg|_{s=0}^{s=1} - 2 \int_0^1 \left( \frac{D}{Ds} \gamma(s), \dot{\gamma} \right) ds
\]
Define \( e = \gamma(s) M E \), and use short-hand notation \( \rho = \rho(X, t), \phi = \phi(X, t), \) and \( B = B(X, t) \). Since the Riemannian energy is always nonnegative, i.e., \( E \geq 0 \), we have
\[
\dot{E} = \frac{\partial E}{\partial t} + \frac{\partial E}{\partial \hat{\theta}} \dot{\hat{\theta}} + 2 \gamma(s) \dot{\hat{\theta}}(s) \bigg|_{s=0}^{s=1} - 2 \int_0^1 \left( \frac{D}{Ds} \gamma(s), \dot{\gamma} \right) ds
\]
which can be simplified as $V \leq 2(V_1 + V_2 + V_3)$, where

$$V_1 = -\frac{\lambda}{3} E^2 + 2e^T \rho (\tilde{v} - \nu_a) - \frac{\beta_a}{\gamma_a} (\nu_a - \tilde{v})^T (\nu_a - \tilde{v}) N_{\nu}^a$$

$$V_2 = -\frac{\lambda}{3} E^2 + 2e^T B \Omega (\theta - \theta_a) - \frac{\beta_a}{\gamma_a} (\theta_a - \tilde{\theta})^T (\theta_a - \tilde{\theta}) N_{\theta}^a$$

$$V_3 = -\frac{\beta\psi}{\gamma\psi} \text{Tr} \left[ (\Psi_a - \Psi)^T \Lambda (\Psi_a - \Psi) N_{\Psi} \right].$$

It follows from (52) and (54) that

$$V_1 \leq -\frac{\beta_a}{\gamma_a} \|\nu_a - \tilde{\theta}\|^2 - \lambda_1 (E - 2\|\omega_a\|) \|\nu_a - \tilde{v}\|)^2$$

$$V_2 \leq -\frac{\beta_a}{\gamma_a} \|\theta_a - \tilde{\theta}\|^2 - \lambda_2 (E - 2\|\omega_\theta\|) \|\theta_a - \tilde{\theta}\|)^2 \leq 0.$$

Using similar manipulations as in the proof of Theorem 4

$$V_3 \leq -\frac{\beta\psi}{\gamma\psi} \|\Psi_a - \Psi\| \|\Omega\|_{F}^2 - \lambda_3 (E - 2\|\omega_\psi\|) \|\Psi_a - \Psi\| \|\Omega\|_{F}^2 ||U||^2 \leq 0.$$

Thus, we have $V \leq 2(V_1 + V_2 + V_3) \leq 0$, which implies that $E, \tilde{v}, \tilde{\theta}, \Psi \in L_\infty$, and that $E \in L_2$. Since $\theta$ is a constant vector and $\tilde{\theta}$ is bounded, $\tilde{\theta}$ is bounded. Similarly, $\tilde{v}$ and $(\Lambda \Psi - I)$ are also bounded. By Assumption 7, $M(x, \tilde{v}, \tilde{t})$ is bounded. Since geodesics have constant speed and $E = \int_0^1 \gamma(s)^T M(x, \tilde{v}, \tilde{t}) \gamma(s) ds = (\gamma_s, \gamma_s)$ is bounded, $\gamma_s$ is bounded. Therefore, $\tilde{E} \in L_\infty$. By Barbalat’s Lemma, $\lim_{t \to \infty} \tilde{x}(t) = 0$. An immediate corollary is $V \to 0$ as $t \to \infty$.

REFERENCES:

[1] K. S. Narendra and A. M. Annaswamy, Stable Adaptive Systems. New York, NY, USA: Dover, 2005.
[2] P. A. Ioannou and J. Sun, Robust Adaptive Control. Englewood Cliffs, NJ, USA: Prentice-Hall, 1996.
[3] S. Sastry and M. Bodson, Adaptive Control: Stability, Convergence and Robustness. Englewood Cliffs, NJ, USA: Prentice-Hall, 1989.
[4] J.-J. E. Slotine and W. Li, Applied Nonlinear Control. Englewood Cliffs, NJ, USA: Prentice Hall, 1991.
[5] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, Nonlinear and Adaptive Control Design. Hoboken, NJ, USA: Wiley, 1995.
[6] K. J. Aström and B. Wittenmark, Adaptive Control. Courier Corporation, 2013.
[7] A. Astolfi, D. Karagiannis, and R. Ortega, Nonlinear and Adaptive Control With Applications. Berlin, Germany: Springer, 2007.
[8] M. Guay and T. Zhang, “Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainties,” Automatica, vol. 39, no. 7, pp. 1283–1293, 2003.
[9] N. Hovakimyan and C. Cao, L1 Adaptive Control Theory: Guaranteed Robustness With Fast Adaptation. Philadelphia, PA, USA: SIAM, 2010.
[10] K. Narendra and A. Annaswamy, “Robust adaptive control in the presence of bounded disturbances,” IEEE Trans. Autom. Control, vol. 31, no. 4, pp. 306–315, Apr. 1986.
[11] H. S. Hussein, Y. Yildiz, M. Matsutani, A. M. Annaswamy, and E. Lavretsky, “Computable delay margins for adaptive systems with state variables accessible,” IEEE Trans. Autom. Control, vol. 62, no. 10, pp. 5039–5054, Oct. 2017.
[12] B. Recht, “A tour of reinforcement learning: The view from continuous control,” Annu. Rev. Control Robot. Auton. Syst., vol. 2, pp. 253–279, 2019.
[13] D. P. Bertsekas and J. N. Tsitsiklis, Neuro-Dynamic Programming., Belmont, MA, USA: Athena Scientific, 1996.
ANNASWAMY et al.: INTEGRATION OF ADAPTIVE CONTROL AND RL FOR REAL-TIME CONTROL AND LEARNING 7755

[42] J. E. Gaudio, A. M. Annaswamy, J. M. Moreu, M. A. Bolender, and T. E. Gibson, “Accelerated learning with robustness to adversarial regressors,” in Proc. 3rd Conf. Learn. Dyn. Control, 2020, pp. 636–650.

[43] J. M. Moreu and A. M. Annaswamy, “A stable high-order tuner for general convex functions,” IEEE Contr. Syst. Lett., vol. 6, pp. 566–571, 2022.

[44] A. M. Annaswamy, “Adaptive control and intersections with reinforcement learning,” Annu. Rev. Control Robot. Auton. Syst., 2023.

[45] J. E. Gaudio, T. E. Gibson, A. M. Annaswamy, M. A. Bolender, and E. Lavretsky, “Connections between adaptive control and optimization in machine learning,” in Proc. IEEE 58th Conf. Decis. Control, 2019, pp. 4562–4568.

[46] N. Matni, A. Prouvsiere, A. Rantzer, and S. Tu, “From self-tuning regulators to reinforcement learning and back again,” in Proc. IEEE 58th Conf. Decis. Control, 2019, pp. 3724–3740.

[47] T. Westenbroek, E. Mazumdar, D. Fridovich-Keil, V. Prabhu, C. J. Tomlin, and S. S. Sastry, “Adaptive control for linearizable systems using on-policy reinforcement learning,” in Proc. IEEE 59th Conf. Decis. Control, 2020, pp. 115–120.

[48] R. Sun, M. L. Greene, D. M. Le, Z. I. Bell, G. Chowdhary, and W. E. Dixon, “Lapunov-based real-time and iterative adjustment of deep neural networks,” IEEE Contr. Syst. Lett., vol. 6, pp. 193–198, 2021.

[49] A. Guha and A. Annaswamy, “Online policies for real-time control using MRAC-RL,” in Proc. IEEE 60th Conf. Decis. Control, 2021, pp. 18061–18066.

[50] S. M. Richards, N. Azizan, J.-J. Slotine, and M. Pavone, “Adaptive-control-oriented meta-learning for nonlinear systems,” in Proc. Robot., Sci. Syst. Virtual, 2021.

[51] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu, “Regret bounds for robust adaptive control of the linear quadratic regulator,” in Proc. Adv. Neural Inf. Process. Syst., 2018, pp. 4192–4201.

[52] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi, “Global convergence of policy gradient methods for the linear quadratic regulator,” in Proc. Int. Conf. Mach. Learn., 2018, pp. 1467–1476.

[53] A. M. Annaswamy, A. Guha, Y. Cui, S. Tang, P. A. Fisher, and J. E. Gaudio, “Online algorithms and policies using adaptive and machine learning approaches,” 2023, arXiv:2105.06577.

[54] A. Morgan and K. Narendra, “On the stability of nonautonomous differential equations $\dot{x} = [A + B(t)]x$, with skew symmetric matrix $B(t)$,” SIAM J. Control Optim., vol. 15, no. 1, pp. 163–176, 1977.

[55] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” J. Artif. Intell. Res., vol. 4, pp. 237–285, 1996.

[56] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” in Proc. Int. Conf. Learn. Representations, 2016.

[57] M. P. Do Carmo, Riemannian Geometry. Berlin, Germany: Springer, 2013.

[58] A. P. Morgan and K. S. Narendra, “On the uniform asymptotic stability of certain linear nonautonomous differential equations,” SIAM J. Control Optim., vol. 15, pp. 5–24, 1977.

[59] M. Holzleitner, L. Gruber, J. Arjona-Medina, J. Brandstetter, and S. Hochreiter, “Convergence proof for actor-critic methods applied to PPO and RUDDER,” in Transactions on Large-Scale Data-And Knowledge-Centered Systems XLVIII, Berlin, Germany: Springer, 2021, pp. 105–130.

[60] R. M. Sanner and J.-J. E. Slotine, “Gaussian networks for direct adaptive control,” IEEE Trans. Neural Netw., vol. 3, no. 6, pp. 837–863, Nov. 1992.

[61] H. Palanisundaram-Madapusi, J. Hoag, and D. Bernstein, “Basis-function optimization for subspace-based nonlinear identification of systems with measured-input nonlinearities,” in Proc. Amer. Control Conf., 2004, pp. 4788–4793.

[62] E. Lavretsky and K. A. Wise, Robust and Adaptive Control With Aerospace Applications. Berlin, Germany: Springer, 2013.

[63] G. H. Hardy et al. An Introduction to the Theory of Numbers. London, U.K.: Oxford Univ. Press, 1979.

[64] B. Kouvartaki and M. Cannon, “Model Predictive Control,” Berlin, Germany:Springer, 2016.

[65] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, “Adaptive control of quadrotor UAVs: A design trade study with flight evaluations,” IEEE Trans. Control Syst. Technol., vol. 21, no. 4, pp. 1400–1406, Apr. 2012.

[66] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017, arXiv:1707.06347.

[67] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in Proc. Int. Conf. Mach. Learn., 2017, pp. 1126–1135.

[68] L. P. Kaelbling, M. L. Littman, and A. W. Moore, “Reinforcement learning: A survey,” J. Artif. Intell. Res., vol. 4, pp. 237–285, 1996.

[69] A. Somanath and A. Annaswamy, “Adaptive control of hypersonic vehicles in presence of aerodynamic and center of gravity uncertainties,” in Proc. IEEE 49th Conf. Decis. Control, 2010, pp. 4661–4666.

[70] M. Schwager and A. M. Annaswamy, “Direct adaptive control of multi-input multi-output closed-loop systems subject to magnitude saturation constraints,” in Proc. IEEE 44th Conf. Decis. Control, 2005, pp. 783–788.

Anuradha M. Annaswamy (Life Fellow, IEEE) received the Ph.D. degree in electrical engineering from Yale University, New Haven, CT, USA. She is the Founder and Director of the Active Adaptive Control Laboratory, Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA. She has authored a graduate textbook on adaptive control and coauthored the National Academy of Sciences Committee report on the Future of Electric Power in the United States in 2021. Her research interests include adaptive control theory and its applications to aerospace, automotive, propulsion, and energy systems.

Ms. Annaswamy is the recipient of the Best Paper Awards (Axelby; CSM; Annual Reviews in Control), Distinguished Member, Distinguished Lecturer Awards from the IEEE Control Systems Society (CSS), and Presidential Young Investigator Award from NSF. She is the coeditor of two vision documents on smart grids and the Impact of Control Technology report. She is a Fellow of IFAC. She was the President of CSS in 2020.

Anubhav Guha is currently working toward the Ph.D. degree in mechanical engineering with the Massachusetts Institute of Technology, Cambridge, MA, USA.

His research interests include adaptive control, reinforcement learning, and robotics.

Yingnan Cui is currently working toward the Ph.D. degree with the Department of Mechanical Engineering, Active Adaptive Control Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA.

His research focuses on adaptive control and optimization methods for machine learning.

Sunbochen Tang (Graduate Student Member, IEEE) is currently working toward the Ph.D. degree in aeronautics and astronautics with Active Adaptive Control Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA.

His research interests include adaptive control, optimization methods, and machine learning.

Peter A. Fisher is currently working toward the master’s degree in mechanical engineering with Active Adaptive Control Laboratory, Massachusetts Institute of Technology, Cambridge, MA, USA.

His research interests include adaptive control, optimization methods, and statistical learning.

Joseph E. Gaudio received the Ph.D. degree in mechanical engineering from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2020.

He is currently a Staff Engineer with Aurora Flight Sciences, Manassas, VA, USA, a Boeing Company. His research interests include adaptive control and machine learning.