REINFORCEMENT LEARNING AND OPTIMAL CONTROL METHODS FOR UNCERTAIN NONLINEAR SYSTEMS

By

SHUBHENDU BHASIN

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2011

Dedicated with love to my parents and my brother; and with reverence to my Guru.

ACKNOWLEDGMENTS

I thank my advisor Dr. Warren E. Dixon for his guidance and motivation during my doctoral research. He groomed me during the initial years of my PhD program and made me understand the virtues of rigor in research. In the latter part of my PhD, he gave me enough freedom to develop my own ideas and grow as an independent researcher. His excellent work ethic has been a constant source of inspiration.

I am also thankful to my committee members, Dr. Pramod Khargonekar, Dr. Prabir Barooah, Dr. Mrinal Kumar and Dr. Frank Lewis, for providing insightful suggestions to improve the quality of my research. I specially thank my collaborators, Dr. Frank Lewis and his student, Vamvoudakis Kyriakos, for giving me a new perspective about the field.

I would also like to acknowledge my coworkers at the Nonlinear Controls and Robotics (NCR) Lab who filled my days with lively technical discussions and friendly banter. I will definitely miss the times when the whole lab would go for Tijuana Flats lunch excursions and NCR Happy Hours. I thank the innumerable friends I made in Gainesville for making my stint at the University of Florida, a memorable experience.

From the bottom of my heart, I thank my parents for their love, support and sacrifice. My mother always took a keen interest in my education. I was fortunate to have her with me during the last year of my PhD. Her unconditional love and encouragement saw me through to the end. I cannot thank her enough. Last but not the least, I am grateful to God and Gurus for guiding me and helping me to draw strength and inspiration from within.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AC	Adaptive Critic (or Actor-Critic)
ACI	Actor-Critic-Identifier
ADP	Approximate Dynamic Programming
DNN	Dynamic Neural Network
DHP	Dual Heuristic Programming
DP	Dynamic Programming
GHJB	Generalized Hamilton-Jacobi-Bellman
HDP	Heuristic Dynamic Programming
HJB	Hamilton-Jacobi-Bellman
MDP	Markov Decision Process
NN	Neural Network
PDE	Partial Differential Equation
PE	Persistence of Excitation
PI	Policy Iteration
RISE	Robust Integral of the Sign of the Error
RL	Reinforcement Learning
TD	Temporal Difference
UUB	Uniformly Ultimately Bounded

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

REINFORCEMENT LEARNING AND OPTIMAL CONTROL METHODS FOR UNCERTAIN NONLINEAR SYSTEMS

By

Shubhendu Bhasin

August 2011

Chair: Warren E. Dixon
Major: Mechanical Engineering

Notions of optimal behavior expressed in natural systems led researchers to develop reinforcement learning (RL) as a computational tool in machine learning to learn actions by trial and error interactions yielding either a reward or punishment. RL provides a way for learning agents to optimally interact with uncertain complex environments, and hence, can address problems from a variety of domains, including artificial intelligence, controls, economics, operations research, etc.

The focus of this work is to investigate the use of RL methods in feedback control to improve the closed-loop performance of nonlinear systems. Most RL-based controllers are limited to discrete-time systems, are offline methods, require knowledge of system dynamics and/or lack a rigorous stability analysis. This research investigates new control methods as an approach to address some of the limitations associated with traditional RL-based controllers.

A robust adaptive controller with an adaptive critic or actor-critic (AC) architecture is developed for a class of uncertain nonlinear systems with disturbances. The AC structure is inspired from RL and uses a two pronged neural network (NN) architecture – an action NN, also called the actor, which approximates the plant dynamics and generates appropriate control actions; and a critic NN, which evaluates the performance of the actor, based on some performance index.

In the context of current literature on RL-based control, the contribution of this work is the development of controllers which learn the optimal policy (approximately) for uncertain nonlinear systems. In contrast to model learning strategies for RL-based control of uncertain systems, the requirement of model knowledge is obviated in this work by the development of a robust identification-based state derivative estimator. The robust identifier is designed to yield asymptotically convergent state derivative estimates which are leveraged for model-free formulation of the Bellman error. The identifier is combined with the traditional actor-critic resulting in a novel actor-critic-identifier architecture, which is used to approximate the infinite-horizon optimal control for continuous-time uncertain nonlinear systems. The method is online, partially model-free, and is the first ever indirect adaptive control approach to continuous-time RL.

CHAPTER 1
INTRODUCTION

1.1 Background and Motivation

RL refers to an agent which interacts with its environment and modifies its actions based on stimuli received in response to its actions. Learning happens through trial and error and is based on a cause and effect relationship between the actions and the rewards/punishment. Decisions/actions which lead to a satisfactory outcome are reinforced and are more likely to be taken when the same situation arises again. Although RL originated in psychology to explain human behavior, it has become a useful computational tool for learning by experience in many engineering applications, such as computer game playing, industrial manufacturing, traffic management, robotics and control, etc. From a computational intelligence perspective, an RL agent chooses actions which minimize the cost of its long-term interactions with the environment [1, 2]. A cost function, which captures the performance criteria, is used to critique the actions of the agent as a numerical reward, called the reinforcement signal. Unlike supervised learning where learning is instructional and based on a set of examples of correct input/output behavior, RL is more evaluative and indicates only the measure of goodness of a particular action. Since interaction is done without a teacher, RL is particularly effective in situations where examples of desired behavior are not available but it is possible to evaluate the performance of actions based on some performance criterion. Improving the closed-loop performance of nonlinear systems has been an active research area in the controls community. Strong connections between RL and feedback control [3] have prompted a major effort towards convergence of the two fields – computational intelligence and controls. Several issues still exist that hinder RL methods for control of nonlinear systems, such as stability, convergence, choice of function approximator, etc. This work attempts to highlight and address some of these issues and provide a scaffolding for constructive RL-based methods for optimal control of uncertain nonlinear systems.

1.2 Problem Statement

The analogy between a continuously learning RL-agent in an unknown environment and a continuously adapting and improving controller for an uncertain system define the problem statement of this work. Specifically, the problem addressed in this work is developing RL-based controllers for continuous-time uncertain nonlinear systems. These controllers are inspired by RL and inherit many important features, like learning by interacting with an uncertain environment, reward-based learning, online implementation, and optimality.

1.3 Literature Survey

AC architectures have been proposed as models of RL [2, 4]. Since AC methods are amenable to online implementation, they have become an important subject of research, particularly in the controls community [5–14]. In AC-based RL, an actor network learns to select actions based on evaluative feedback from the critic to maximize future rewards. Due to the success of NNs as universal approximators [15, 16], they have become a natural choice in AC architectures for approximating unknown plant dynamics and cost functions [17, 18]. Typically, the AC architecture consists of two NNs – an action or actor NN and a critic NN. The critic NN approximates the evaluation function, mapping states to an estimated measure of the value function, while the action NN approximates an optimal control law and generates actions or control signals. Following the works of Sutton [1], Barto [19], Watkins [20], and Werbos [21], current research focuses on the relationship between RL and dynamic programming (DP) [22] methods for solving optimal control problems. Due to the *curse of dimensionality* associated with using DP [22], Werbos [5] introduced an alternative Approximate Dynamic Programming (ADP) approach which gives an approximate solution to the DP problem, or the *Hamiltonian-Jacobi-Bellman* (HJB) equation for optimal control. A detailed review of ADP designs can be found in [6]. Various modifications to ADP algorithms have since been proposed [7, 23, 24].

The performance of ADP-based controllers have been successfully demonstrated on various nonlinear plants with unknown dynamics. Venayagamoorthy et al. used ADP for control of turbogenerators, synchronous generators, and power systems [25, 26]. Ferrari and Stengel [27] used a Dual Heuristic Programming (DHP) based ADP approach to control a nonlinear simulation of a jet aircraft in the presence of parameter variations and control failures. Jagannathan et al. [28] used ACs for grasping control of a three-finger-gripper. Some other interesting applications are missile control [29], HVAC control [30], and control of distributed parameter systems [11].

Convergence of ADP algorithms for RL-based control is studied in [7–10, 31, 32]. A policy iteration (PI) algorithm is proposed in [33] using Q-functions for the discrete-time LQR problem and convergence to the state feedback optimal solution is proven. In [34], model-free Q-learning is proposed for linear discrete-time systems with guaranteed convergence to the $\mathcal{H}_2$ and $\mathcal{H}_\infty$ state feedback control solution. Most of the previous work on ADP has focused on either finite state Markovian systems or discrete-time systems [35, 36]. The inherently iterative nature of the ADP algorithm has impeded the development of closed-loop controllers for continuous-time uncertain nonlinear systems. Extensions of ADP-based controllers to continuous-time systems entails challenges in proving stability, convergence, and ensuring the algorithm is online and model-free. Early solutions to the problem consisted of using a discrete-time formulation of time and state, and then applying an RL algorithm on the discretized system. Discretizing the state space for high dimensional systems requires a large memory space and a computationally prohibitive learning process. Convergence of PI for continuous-time LQR was first proved in [37]. Baird [38] proposed *Advantage Updating*, an extension of the Q-learning algorithm which could be implemented in continuous-time and provided faster convergence. Doya [39] used an HJB framework to derive algorithms for value function approximation and policy improvement, based on a continuous-time version of the temporal difference (TD) error. Murray et al. [8] also used the HJB framework to develop a *stepwise stable* iterative ADP

algorithm for continuous-time input-affine systems with an input quadratic performance measure. In Beard et al. [40], Galerkin's spectral method is used to approximate the solution to the generalized HJB (GHJB), using which a stabilizing feedback controller was computed offline. Similar to [40], Abu-Khalaf and Lewis [41] proposed a least-squares successive approximation solution to the GHJB, where an NN is trained offline to learn the GHJB solution. Another continuous-time formulation of adaptive critic is proposed in Hanselman [12].

All of the aforementioned approaches for continuous-time nonlinear systems require complete knowledge of system dynamics. The fact that continuous-time ADP requires knowledge of the system dynamics has hampered the development of continuous-time extensions to ADP-based controllers for nonlinear systems. Recent results by [13, 42] have made new inroads by addressing the problem for partially unknown nonlinear systems. A PI-based hybrid continuous-time/discrete-time sampled data controller is designed in [13, 42], where the feedback control operation of the actor occurs at faster time scale than the learning process of the critic. Vamvoudakis and Lewis [14] extended the idea by designing a model-based online algorithm called *synchronous PI* which involved synchronous continuous-time adaptation of both actor and critic NNs.

1.4 Dissertation Outline

Chapter 1 serves as an introduction. The motivation, problem statement, literature survey and the contributions of the work are provided in this chapter.

Chapter 2 discusses the key elements in the field of RL from a computational intelligence point of view and discusses how these techniques can be applied to solve control problems. Further, the optimal control problem, optimal control methods, and their limitations are discussed. Connections between RL and optimal control are established and implementation issues are highlighted.

Chapter 3 develops a continuous-time adaptive critic controller to yield asymptotic tracking of a class of uncertain nonlinear systems with bounded disturbances. The

proposed AC-based controller consists of two NNs - an action NN, also called the actor, which approximates the plant dynamics and generates appropriate control actions; and a critic NN, which evaluates the performance of the actor based on some performance index. The reinforcement signal from the critic is used to develop a composite weight tuning law for the action NN based on Lyapunov stability analysis. A recently developed robust feedback technique, RISE (Robust Integral of the Sign of the Error), is used in conjunction with the feedforward action NN to yield a semi-global asymptotic result.

Chapter 4 develops a robust identification-based state derivative estimation method for uncertain nonlinear systems. The identifier architecture consists of a recurrent multi-layer dynamic NN which approximates the system dynamics online, and a continuous robust feedback RISE term which accounts for modeling errors and exogenous disturbances. The developed method finds applications in RL-based control methods for uncertain nonlinear systems.

Chapter 5 develops an online adaptive RL-based solution for the infinite-horizon optimal control problem for continuous-time uncertain nonlinear systems. A novel actor-critic-identifier (ACI) is developed to approximate the HJB equation using three NN structures - actor and critic NNs approximate the optimal control and the optimal value function, respectively, and a robust dynamic NN (DNN) identifier asymptotically approximates the uncertain system dynamics. An advantage of the using the ACI architecture is that learning by the actor, critic, and identifier is continuous and simultaneous, without requiring knowledge of system drift dynamics. Convergence of the algorithm is analyzed using Lyapunov-based adaptive control methods.

Chapter 6 concludes the dissertation with a discussion of the key ideas, contributions and limitations of this work. It also points to future research directions and paves a path forward for further developments in the field.

1.5 Contributions

This work focuses on developing RL-based controllers for continuous-time nonlinear systems. The contributions of Chapters 3-*5* are as follows.

Asymptotic tracking by a RL-based adaptive critic controller: AC-based controllers are typically discrete and/or yield a uniformly ultimately bounded stability result due to the presence of disturbances and uncertain approximation errors. A continuous asymptotic AC-based tracking controller is developed for a class of nonlinear systems with bounded disturbances. The approach is different from the optimal control-based ADP approaches proposed in literature [8–10, 13, 14, 32, 42], where the critic usually approximates a long-term cost function and the actor approximates the optimal control. However, the similarity with the ADP-based methods is in the use of the AC architecture, borrowed from RL, where the critic, through a reinforcement signal affects the behavior of the actor leading to an improved performance. The proposed robust adaptive controller consists of a NN feedforward term (actor NN) and a robust feedback term, where the weight update laws of the actor NN are designed as a composite of a tracking error term and a RL term (from the critic), with the objective of minimizing the tracking error [43–45]. The robust term is designed to withstand the external disturbances and modeling errors in the plant. Typically, the presence of bounded disturbances and NN approximation errors lead to a uniformly ultimately bounded (UUB) result. The main contribution of this work is the use of a recently developed continuous feedback technique, RISE [46, 47], in conjunction with the AC architecture to yield asymptotic tracking of an unknown nonlinear system subjected to bounded external disturbances. The use of RISE in conjunction with the action NN makes the design of the critic NN architecture challenging from a stability standpoint. To this end, the critic NN is combined with an additional RISE-like term to yield a reinforcement signal, which is used to update the weights of the action NN. A Lyapunov stability analysis guarantees closed-loop stability

of the system. Experiments are performed to demonstrate the improved performance with the proposed RL-based AC method.

Robust identification-based state derivative estimation for nonlinear systems: A state derivative estimation method is developed which can be used to design complete or partial model-free RL-methods for control of uncertain nonlinear systems. The developed robust identifier provides online estimates of the state derivative of uncertain nonlinear systems in the presence of exogenous disturbances. The result differs from existing pure robust methods in that the proposed method combines an adaptive DNN system identifier with a robust RISE feedback to ensure asymptotic convergence to the state derivative, which is proven using a Lyapunov-based stability analysis. Simulation results in the presence of noise show an improved transient and steady state performance of the developed identifier in comparison to several other derivative estimation methods including: a high gain observer, a 2-sliding mode robust exact differentiator, and numerical differentiation methods, such as backward difference and central difference.

A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems: A novel actor-critic-identifier architecture is developed to learn the approximate solution to the HJB equation for infinite-horizon optimal control of uncertain nonlinear systems. The online method is the first ever indirect adaptive control approach to continuous-time RL. Another contribution of the developed method is that unlike previous results in literature, the learning by the actor, critic and identifier is continuous and simultaneous, and the novel addition of the identifier to the traditional actor-critic architecture eliminates the need to know the system drift dynamics. The stability and convergence of the algorithm is rigorously analyzed. A PE condition is required to ensure exponential convergence to a bounded region in the neighborhood of the optimal control and UUB stability of the closed-loop system.

CHAPTER 2
REINFORCEMENT LEARNING AND OPTIMAL CONTROL

RL refers to the problem of a goal-directed agent interacting with an uncertain environment. The goal of an RL agent is to maximize a long-term scalar reward by sensing the state of the environment and taking actions which affect the state. At each step, an RL system gets evaluative feedback about the performance of its action, allowing it to improve the performance of subsequent actions. Several RL methods have been developed and successfully applied in machine learning to learn optimal policies for finite-state finite-action discrete-time Markov Decision Processes (MDPs), shown in Fig. 2-1. An analogous RL control system is shown in Fig. 2-2, where the controller, based on state feedback and reinforcement feedback about its previous action, calculates the next control which should lead to an improved performance. The reinforcement signal is the output of a performance evaluator function, which is typically a function of the state and the control. An RL system has a similar objective to an optimal controller which aims to optimize a long-term performance criterion while maintaining stability. This chapter discusses the key elements in the field of RL and how they can be applied to solve control problems. Further, the optimal control problem, optimal control methods, and their limitations are discussed. Connections between RL and optimal control are established and implementation issues are highlighted, which motivate the methods developed in this dissertation.

2.1 Reinforcement Learning Methods

RL methods typically estimate the value function, which is a measure of goodness of a given action for a given state. The value function represents the reward/penalty accumulated by the agent in the long run, and for a deterministic MDP, may be defined as an infinite-horizon discounted return as [2]

$$V^u(x_0) \;=\; \sum_{k=0}^{\infty} \gamma^k r_{k+1},$$

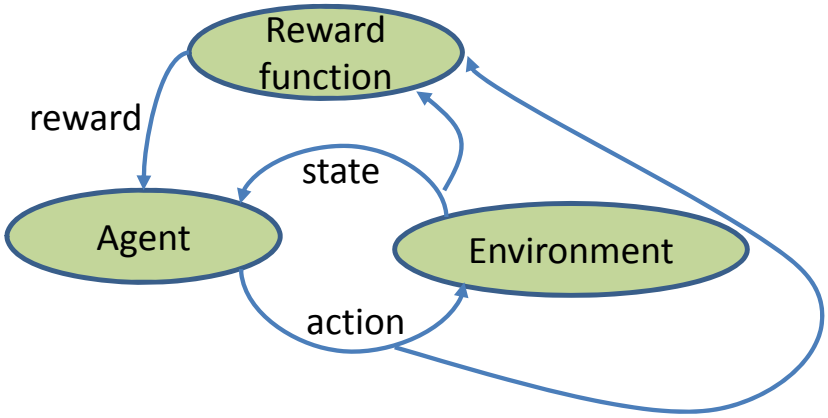

Figure 2-1. Reinforcement Learning for MDP.

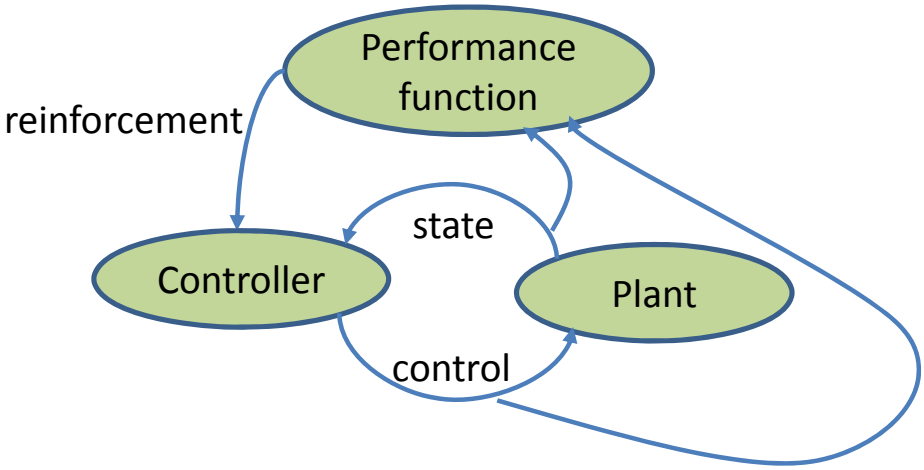

Figure 2-2. Reinforcement Learning control system.

where x_k and u_k are the state and action, respectively, for the discrete-time system $x_{k+1} = f(x_k, u_k)$, $r_{k+1} \triangleq r(x_k, u_k)$ is the reward/penalty at the k^{th} step, and $\gamma \in [0,\ 1)$ is the discount factor used to discount future rewards. The objective of an RL method is to determine a policy which maximizes the value function. Since the value function is unknown, typically the first step is to estimate the value function, which can be expressed using Bellman's equation as [2]

$$V^u(x) \quad = \quad r(x,u) + \gamma V^u(f(x,u)),$$

where the index k is suppressed. The optimal value function is defined as

$$V^*(x) = \min_u V^u(x),$$

which can also be expressed using the Bellman optimality condition as

$$\begin{aligned} V^*(x) &= \min_u [r(x,u) + \gamma V^*(f(x,u))] \\ u^*(x) &= \arg\min_u [r(x,u) + \gamma V^*(f(x,u))]. \end{aligned} \tag{2–1}$$

The above Bellman relations form the basis of all RL methods – policy iteration, value iteration, and Q-learning [2, 20, 35]. RL methods can be categorized as model-based and model-free. Model-based or DP-based RL algorithms utilize the expression in Eq. 2–1 but are offline and require perfect knowledge of the environment, as seen from Eq. 2–1. On the other hand, model-free RL algorithms are based on temporal difference (TD), which refers to the difference between temporally successive estimates of the same quantity. In contrast to DP-based RL methods, TD-based RL methods are online and do not use an explicit model of the system, rather they use data (set of samples, trajectories etc.) obtained from the process, i.e., they learn by interacting with the environment. Some of the popular RL methods are subsequently discussed.

2.1.1 Policy Iteration

Policy Iteration (PI) algorithms [22, 48] successively alternate between policy evaluation and policy improvement. The algorithm starts with an initial admissible policy, estimates the value function (policy evaluation), and then improves the policy using a greedy search on the estimated value function (policy improvement). The policy evaluation step in DP-based PI is performed using the following recurrence relations until convergence to the value function

$$V^u(x) \leftarrow r(x,u) + \gamma V^u(f(x,u)), \tag{2–2}$$

where the symbol '$\leftarrow$' denotes the value on the right being assigned to the quantity on the left. After the convergence of policy evaluation, policy improvement is performed using

$$\bar{u}(x) \quad = \quad \arg\min_{a} \left[r(x,a) + \gamma V^{u}(f(x,a)) \right] \tag{2–3}$$

It can be seen from Eqs. 2–2 and 2–3 that the DP-based PI algorithm requires knowledge of the system model $f(x,u)$. Using the model-free $TD(0)$ algorithm [1], which learns from interacting with the environment, this limitation is overcome. Using the $TD(0)$ algorithm, the value function is estimated using the following update

$$V^{u}(x) \quad \leftarrow \quad V^{u}(x) + \alpha \left[r(x,u) + \gamma V^{u}(\bar{x}) - V^{u}(x) \right], \tag{2–4}$$

where $\alpha \in (0,\ 1]$ is the learning rate, and $\bar{x}$ denotes the next state observed after performing action u at x. In contrast to DP-based policy evaluation, the value function estimation in Eq. 2–4 does not require an explicit model of the system. The PI algorithm converges to the optimal policy [48]. Online PI algorithms do not wait for the convergence of the policy evaluation step to implement policy improvement; however, their convergence can only be guaranteed only under very restrictive conditions, such as generation of infinitely long trajectories for each iteration [49].

2.1.2 Value Iteration

Value Iteration (VI) algorithms directly estimate the optimal value function, which is then used to compute the optimal policy. It combines the truncated policy evaluation and policy improvement steps in one step using the following recurrence relations from DP [2]

$$V(x) \quad \leftarrow \quad \min_{a} \left[r(x,a) + \gamma V(f(x,a)) \right]$$

VI converges to the optimal $V^{*}(x)$, and is said to be less computationally intensive than PI, although PI typically converges in fewer iterations [35].

2.1.3 Q-Learning

Q-Learning algorithms use Q-factors $Q(x,u)$, which are state-action pairs instead of the state value function $V(x)$. The Q-iteration algorithm uses TD learning to find the optimal Q-factor $Q^*(x,u)$ as

$$Q(x,u) \leftarrow Q(x,u) + \alpha \left[r(x,u) + \gamma \min_a Q(\bar{x},a) - Q(x,u) \right].$$

The Q-learning algorithm [20] is one of the major breakthroughs in reinforcement learning, since it involves learning the optimal action-value function independent of the policy being followed (also called off-policy)[1] , which greatly simplifies the convergence analysis of the algorithm. Adequate exploration is, however, needed for the convergence to Q^*. The optimal policy can be directly found from performing a greedy search on Q^* as

$$u^*(x) = \arg\min_a Q^*(x,a).$$

2.2 Aspects of Reinforcement Learning Methods

This section discusses aspects and issues in implementation of the RL methods on high dimensional and large-scale practical systems.

2.2.1 Curse of Dimensionality and Function Approximation

RL methods where value function estimates are represented as a table require, at every iteration, storage and updating of all the table entries corresponding to the entire state space. In fact, the computation and storage requirements increase exponentially with the size of the state space, also called the *curse of dimensionality*. The problem is compounded when considering continuous spaces which contain infinitely many states and actions. One solution approach is to represent value functions using function approximators, which are based on supervised learning, and generalize based on limited

[1] An on-policy variant of Q-learning, SARSA [50], is based on policy iteration.

information about the state space [2]. A convenient way to represent value functions is by using linearly parameterized approximators of the form $\theta^T \phi(x)$, where θ is the unknown parameter vector, and ϕ is a user-defined basis function. Selecting the right basis function which represents all the independent features of the value function is crucial in solving the RL problem. Some prior knowledge regarding the process is typically included in the basis function. The parameter vector is estimated using optimization algorithms, e.g., gradient descent, least squares etc. Multi-layer neural networks may also be used as nonlinearly parameterized approximators; however, weight convergence is harder to prove as compared to linearly parameterized network structures.

2.2.2 Actor-Critic Architecture

Actor-critic methods, introduced by Barto [19], implement the policy iteration algorithm online, where the critic is typically a neural network which implements policy evaluation and approximates the value function, whereas the actor is another neural network which approximates the control. The critic evaluates the performance of the actor using a scalar reward from the environment and generates a TD error. The actor-critic neural networks, shown in Fig. 2-3 are updated using gradient update laws based on the TD error.

2.2.3 Exploitation Vs Exploration

The trade-off between exploitation and exploration has been a topic of much research in the RL community [51]. For an agent in an unknown environment, exploration is required to try out different actions and learn based on trial and error, whereas past experience may also be exploited to select the best actions and minimize the cost of learning. For sample or trajectory based RL methods (e.g., Monte Carlo) in large dimensional spaces, selecting best actions (e.g., greedy policy) based on current estimates is not sufficient because better alternative actions may potentially never be explored. Sufficient exploration is essential to learn the global optimal solution. However, too much exploration can also be costly in terms of performance and stability when the method is

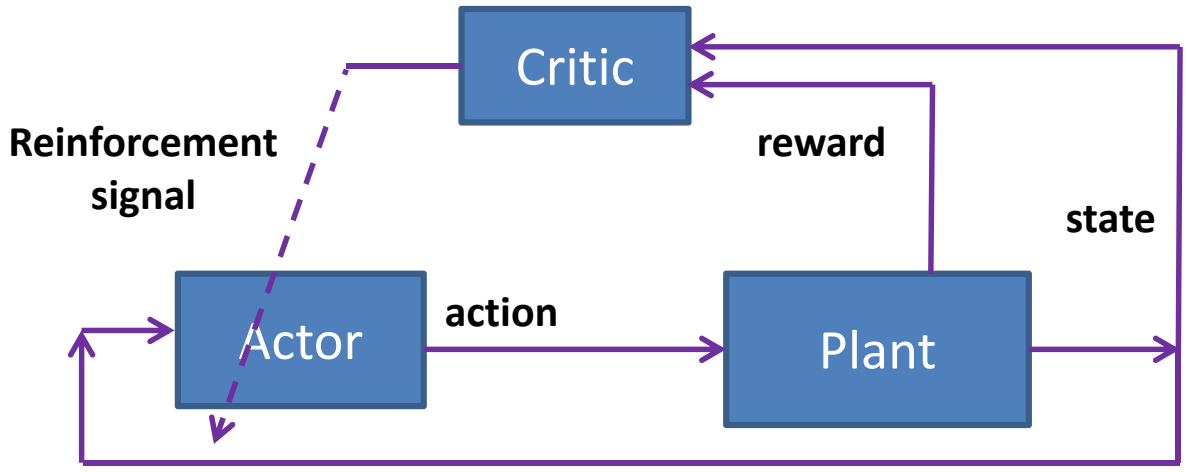

Figure 2-3. Actor-critic architecture for online policy iteration.

implemented online. One approach is to use a ε-greedy policy, where the exploration is the highest when the agent starts learning, but gradually decays as experience is gained and exploitation is preferred to reach the optimal solution.

2.3 Infinite Horizon Optimal Control Problem

RL has close connections with optimal control. In this section, the undiscounted infinite horizon optimal control problem is formulated for continuous-time nonlinear systems. Consider a continuous-time nonlinear system

$$\dot{x} \quad = \quad F(x,u), \tag{2–5}$$

where $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$, $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control input, $F : \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}^n$ is Lipschitz continuous on $\mathcal{X} \times \mathcal{U}$ containing the origin, such that the solution $x(t)$ of the system in Eq. 2–5 is unique for any finite initial condition x_0 and control $u \in \mathcal{U}$. It is also assumed that $F(0,0) = 0$. Further, the system is stabilizable, i.e. there exists a continuous feedback control law $u(x(t))$ such that the closed-loop system is asymptotically stable.

The infinite-horizon scalar cost function for the system Eq. 2–5 can be defined as

$$J(x(t),\ \underset{t\leq\tau<\infty}{u(\tau)}\ ) \quad = \quad \int_t^{\infty} r(x(s),u(s))\,ds, \tag{2–6}$$

where t is the initial time, $r(x,u) \in \mathbb{R}$ is the immediate or local cost for the state and control, defined as

$$r(x,u) \quad = \quad Q(x) + u^T R u, \tag{2–7}$$

where $Q(x) \in \mathbb{R}$ is continuously differentiable and positive definite, and $R \in \mathbb{R}^{m\times m}$ is a positive-definite symmetric matrix. The optimal control problem is to find an admissible control $u^* \in \Psi(\mathcal{X})$, such that the cost in Eq. 2–6 associated with the system Eq. 2–5 is minimized [52]. An admissible control input $u(t)$ can be defined as a continuous feedback control law $u(x(t)) \in \Psi(\mathcal{X})$, where $\Psi(\cdot)$ denotes the set of admissible controls, which asymptotically stabilizes the system Eq. 2–5 on $\mathcal{X}$, $u(0) = 0$, and $J(\cdot)$ in Eq. 2–6 is finite.

The optimal value function can be defined as

$$V^*(x(t)) \quad = \quad \min_{\substack{u(\tau)\in\Psi(\mathcal{X})\\ t\leq\tau<\infty}} \int_t^{\infty} r(x(s),u(x(s)))\,ds. \tag{2–8}$$

Assuming the value function is continuously differentiable, Bellman's principle of optimality can be used to derive the following optimality condition [52]

$$0 \quad = \quad \min_{u(t)\in\Psi(\mathcal{X})} \left[r(x,u) + \frac{\partial V^*(x)}{\partial x} F(x,u) \right], \tag{2–9}$$

which is a nonlinear partial differential equation (PDE), also called the HJB equation. Based on the assumption that $V^*(x)$ is continuously differentiable, the HJB in Eq. 2–9 provides a means to obtain the optimal control $u^*(x)$ in feedback form. Using the convex local cost in Eqs. 2–7 and 2–9, a closed-form expression for the optimal control can be derived as

$$u^*(x) \quad = \quad -\frac{1}{2} R^{-1} \frac{\partial F(x,u)}{\partial u}^T \frac{\partial V^*(x)}{\partial x}^T. \tag{2–10}$$

For the control-affine dynamics of the form

$$\dot{x} \;=\; f(x) + g(x)u, \tag{2–11}$$

where $f(x) \in \mathbb{R}^n$ and $g(x) \in \mathbb{R}^{n\times m}$, the expression in Eq. 2–10 can be written in terms of the system state as

$$u^*(x) \;=\; -\frac{1}{2}R^{-1}g^T(x)\frac{\partial V^*(x)}{\partial x}^T. \tag{2–12}$$

In general, the solutions to the optimal control problem may not be smooth [53]. Existence of a unique non-smooth solution (called the viscosity solution) is studied in [53], [54].

The HJB in Eq. 2–9 can be rewritten in terms of the optimal value function by substituting for the local cost in Eq. 2–7, the system in Eq. 2–11 and the optimal control in Eq. 2–12, as

$$\begin{aligned} 0 \;&=\; Q(x) + \frac{\partial V^*(x)}{\partial x}f(x) - \frac{1}{4}\frac{\partial V^*(x)}{\partial x}g(x)R^{-1}g^T(x)\frac{\partial V^*(x)}{\partial x}^T, \\ 0 \;&=\; V^*(0). \end{aligned} \tag{2–13}$$

Although in closed-form, the optimal policy in Eq. 2–12 requires knowledge of the optimal value function $V^*(x)$, the solution of the HJB equation in Eq. 2–13. The HJB equation is problematic to solve in general and may not have an analytical solution.

2.4 Optimal Control Methods

Since the solution of the HJB is prohibitively difficult and sometimes even impossible, several alternative methods are investigated in literature. The *calculus of variations* approach generates a set of a first-order necessary optimality conditions, called the Euler-Lagrange equations, resulting in a two-point (or multi-point) boundary value problem, which is typically solved numerically using indirect methods, such as shooting, multiple shooting etc [52]. Another numerical approach is to use direct methods where the state and/or control are approximated using function approximators or discretized using

collocation and the optimal control problem is transcribed to a nonlinear programming problem, which can solved using methods such as direct shooting, direct collocation, pseudo-spectral methods etc. [55, 56]. Although these numerical approaches are effective and practical, they are open-loop, offline, require exact model knowledge and are dependent on initial conditions. Another approach based on feedback linearization involves robustly canceling the system nonlinearities, thereby reducing the system to a linear system, and solving the associated Algebraic Riccati Equation (ARE)/Differential Riccati Equation (DRE) for optimal control [57, 58]. A drawback of feedback linearization is that it solves a transformed optimal control problem with respect to a part of the control while the other part is used to cancel the nonlinear terms. Moreover, linearization cancels all nonlinearities, some of which may be useful for the system. Inverse optimal controllers circumvent the task of solving the HJB by proving optimality of a control law for a meaningful cost function [59–61]. The fact that the cost function cannot be chosen a priori by the user limits the applicability of the method.

Given the limitations of methods that seek an exact optimal solution, the focus of some literature has shifted towards developing methods which yield a sub-optimal or an approximately optimal solution. Model-predictive control (MPC) or receding horizon control (RHC) [62, 63] is an example of an online model-based approximate optimal control method which solve the optimal control problem over a finite time horizon at every state transition leading to a state feedback optimal control solution. These methods have been successfully applied in process control where the model is exactly known and the dynamics are slowly varying [64, 65]. An offline successive approximation method, proposed in [66], improves the performance of an initial stabilizing control by approximating the solution to the generalized HJB (GHJB) equation and then using the Bellman's optimality principle to compute an improved control law. This process is repeated and proven to converge to the optimal policy. The GHJB, unlike the HJB, is a linear PDE which is more tractable to solve, e.g., using methods like the Galerkin

projection [40]. The successive approximation method is similar to the policy iteration algorithm in RL; however, the method is offline and requires complete model knowledge. To alleviate the curse of dimensionality associated with dynamic programming, a family of methods, called AC designs (also called ADP), were developed in [6, 17, 35, 36] to solve the optimal control problem using RL and neural network backpropagation algorithms. The methods are, however, applicable only for discrete-time systems and lack a rigorous Lyapunov stability analysis.

2.5 Adaptive Optimal Control and Reinforcement Learning

Most optimal control approaches discussed in Section 2.4 are offline and require complete model knowledge. Even for linear systems, where the LQR gives the closed-form analytical solution to the optimal control problem, the ARE is solved offline and requires exact knowledge of the system dynamics. Adaptive control provides an inroad to design controllers which can adapt online to the uncertainties in system dynamics, based on minimization of the output error (e.g., using gradient or least squares methods). However, classical adaptive control methods do not maximize a long-term performance function, and hence are not optimal. *Adaptive optimal control* refers to methods which learn the optimal solution online for uncertain systems. RL methods described in Section 2.1 have been successfully used in MDPs to learn optimal polices in uncertain environments, e.g., TD-based Q-learning is an online model-free RL method for learning optimal policies. In [3], Sutton et al. argue that RL is a direct adaptive optimal control technique. Owing to the discrete nature of RL algorithms, many methods have been proposed for adaptive optimal control of discrete-time systems [6, 7, 10, 33, 67–70]. Unfortunately, an RL formulation for continuous-time systems is not as straightforward as in the discrete-time case, because while the TD error in the latter is model-free, it is not the case with the former, where the TD error formulation inherently requires complete knowledge of the system dynamics (see Eq. 2–9). RL methods based on the model-based TD error for continuous-time systems are proposed in [8, 14, 39, 41]. A partial model-free solution

is proposed in [13] using an actor-critic architecture, however, the resulting controller is hybrid with a continuous-time actor and a discrete-time critic. Other issues concerning RL-based controllers are: closed-loop stability, convergence to the optimal control, function approximation, and tradeoff between exploitation and exploration. Few results have rigorously addressed these issues which are critical for successful implementation of RL methods for feedback control. The work in this dissertation is motivated by the need to provide a theoretical foundation for RL-based control methods and explore their potential as adaptive optimal control methods.

CHAPTER 3
ASYMPTOTIC TRACKING BY A REINFORCEMENT LEARNING-BASED ADAPTIVE CRITIC CONTROLLER

AC based controllers are typically discrete and/or yield a uniformly ultimately bounded stability result due to the presence of disturbances and unknown approximation errors. The objective in this chapter is to design a continuous-time AC controller which yields asymptotic tracking of a class of uncertain nonlinear systems with bounded disturbances. The proposed AC-based controller architecture consists of two NNs — an action NN, also called the actor, which approximates the plant dynamics and generates appropriate control actions; and a critic NN, which evaluates the performance of the actor based on some performance index. The reinforcement signal from the critic is used to develop a composite weight tuning law for the action NN based on Lyapunov stability analysis. A recently developed robust feedback technique, RISE, is used in conjunction with the feedforward action neural network to yield a semi-global asymptotic result.

3.1 Dynamic Model and Properties

The mn-th order MIMO Brunovsky form[1] can be written as [43]

$$\begin{aligned} \dot{x}_1 &= x_2 \\ &\vdots \\ \dot{x}_{n-1} &= x_n \\ \dot{x}_n &= g(x) + u + d \\ y &= x_1, \end{aligned} \tag{3–1}$$

[1] The Brunovsky form can be used to model many physical systems, e.g., Euler-Lagrange systems.

where $x(t) \triangleq [x_1^T \; x_2^T \ldots x_n^T]^T \in \mathbb{R}^{mn}$ are the measurable system states, $u(t) \in \mathbb{R}^m$, $y \in \mathbb{R}^m$ are the control input and system output, respectively; $g(x) \in \mathbb{R}^m$ is an unknown smooth function, locally Lipschitz in x; and $d(t) \in \mathbb{R}^m$ is an external bounded disturbance.

Assumption 3.1. *The function $g(x)$ is second order differentiable, i.e., $g(\cdot), \dot{g}(\cdot), \ddot{g}(\cdot) \in \mathcal{L}_\infty$ if $x^{(i)}(t) \in \mathcal{L}_\infty$, $i = 0, 1, 2$ where $(\cdot)^{(i)}(t)$ denotes the i^{th} derivative with respect to time.*

Assumption 3.2. *The desired trajectory $y_d(t) \in \mathbb{R}^m$ is designed such that $y_d^{(i)}(t) \in \mathcal{L}_\infty$, $i = 0, 1, ..., n+1$.*

Assumption 3.3. *The disturbance term and its first and second time derivatives are bounded i.e. $d(t), \dot{d}(t), \ddot{d}(t) \in \mathcal{L}_\infty$.*

3.2 Control Objective

The control objective is to design a continuous RL-based NN controller such that the output $y(t)$ tracks a desired trajectory $y_d(t)$. To quantify the control objective, the tracking error $e_1(t) \in \mathbb{R}^m$ is defined as

$$e_1 \triangleq y - y_d. \tag{3–2}$$

The following filtered tracking errors are defined to facilitate the subsequent stability analysis

$$\begin{aligned} e_2 &\triangleq \dot{e}_1 + \alpha_1 e_1 \\ e_i &\triangleq \dot{e}_{i-1} + \alpha_{i-1} e_{i-1} + e_{i-2}, \qquad i = 3, ..., n \qquad (3–3) \\ r &\triangleq \dot{e}_n + \alpha_n e_n, \qquad (3–4) \end{aligned}$$

where $\alpha_1, ..., \alpha_n \in \mathbb{R}$ are positive constant control gains. Note that the signals $e_1(t), ..., e_n(t) \in \mathbb{R}^m$ are measurable whereas the filtered tracking error $r(t) \in \mathbb{R}^m$ in Eq. 3–4 is not measurable since it depends on $\dot{x}_n(t)$. The filtered tracking errors in Eq. 3–3 can be expressed in terms of the tracking error $e_1(t)$ as

$$e_i = \sum_{j=0}^{i-1} a_{ij} e_1^{(j)}, \qquad i = 2, ..., n \tag{3–5}$$

where $a_{ij} \in \mathbb{R}$ are positive constants obtained from substituting Eq. 3–5 in Eq. 3–3 and comparing coefficients [47]. It can be easily shown that

$$a_{ij} = 1, \qquad j = i - 1. \tag{3–6}$$

3.3 Action NN-Based Control

Using Eqs. 3–2-3–6, the open loop error system can be written as

$$r = y^{(n)} - y_d^{(n)} + f, \tag{3–7}$$

where $f(e_1, \dot{e}_1, ..., e_1^{(n-1)}) \in \mathbb{R}^m$ is a function of known and measurable terms, defined as

$$f = \sum_{j=0}^{n-2} a_{nj}(e_1^{(j+1)} + \alpha_n e_1^{(j)}) + \alpha_n e_1^{(n-1)}.$$

Substituting the dynamics from Eq. 3–1 into Eq. 3–7 yields

$$r = g(x) + d - y_d^{(n)} + f + u. \tag{3–8}$$

Adding and subtracting $g(x_d) : \mathbb{R}^{mn} \to \mathbb{R}^m$, where $g(x_d)$ is a smooth unknown function of the desired trajectory $x_d(t) \triangleq [y_d^T \;\; \dot{y}_d^T \ldots (y_d^{(n-1)})^T]^T \in \mathbb{R}^{mn}$, the expression in Eq. 3–8 can be written as

$$r = g(x_d) + S + d + Y + u, \tag{3–9}$$

where $Y(e_1, \dot{e}_1, ..., e_1^{(n-1)}, y_d^{(n)}) \in \mathbb{R}^m$ contains known and measurable terms and is defined as

$$Y \triangleq -y_d^{(n)} + f, \tag{3–10}$$

and the auxiliary function $S(x, x_d) \in \mathbb{R}^m$ is defined as

$$S \triangleq g(x) - g(x_d).$$

The unknown nonlinear term $g(x_d)$ can be represented by a multi-layer NN as

$$g(x_d) = W_a^T \sigma(V_a^T x_a) + \varepsilon\left(x_a\right), \tag{3–11}$$

where $x_a(t) \in \mathbb{R}^{mn+1} \triangleq [1 \;\; x_d^T]^T$ is the input to the NN, $W_a \in \mathbb{R}^{(N_a+1)\times m}$ and $V_a \in \mathbb{R}^{(mn+1)\times N_a}$ are the constant bounded ideal weights for the output and hidden layers respectively with N_a being the number of neurons in the hidden layer, $\sigma(\cdot) \in \mathbb{R}^{N_a+1}$ is the bounded activation function, and $\varepsilon(x_a) \in \mathbb{R}^m$ is the function reconstruction error.

Remark 3.1. *The NN used in Eq. 3–11 is referred to as the action NN or the associative search element (ASE) [19], and it is used to approximate the system dynamics and generate appropriate control signals.*

Based on the assumption that the desired trajectory is bounded, the following inequalities hold

$$\|\varepsilon_a(x_a)\| \leq \varepsilon_{a1}, \quad \|\dot{\varepsilon}_a(x_a, \dot{x}_a)\| \leq \varepsilon_{a_2}, \quad \|\ddot{\varepsilon}_a(x_a, \dot{x}_a, \ddot{x}_a)\| \leq \varepsilon_{a_3}, \tag{3–12}$$

where $\varepsilon_{a_1}, \varepsilon_{a_2}, \varepsilon_{a_3} \in \mathbb{R}$ are known positive constants. Also, the ideal weights are assumed to exist and be bounded by known positive constants [18], such that

$$\|V_a\| \leq \bar{V}_a, \quad \|W_a\| \leq \bar{W}_a. \tag{3–13}$$

Substituting Eq. 3–11 in Eq. 3–9, the open loop error system can now be written as

$$r = W_a^T \sigma(V_a^T x_a) + \varepsilon(x_a) + S + d + Y + u. \tag{3–14}$$

The NN approximation for $g(x_d)$ can be represented as

$$\hat{g}(x_d) = \hat{W}_a^T \sigma(\hat{V}_a^T x_a),$$

where $\hat{W}_a(t) \in \mathbb{R}^{(N_a+1)\times m}$ and $\hat{V}_a(t) \in \mathbb{R}^{(mn+1)\times N_a}$ are the subsequently designed estimates of the ideal weights. The control input $u(t)$ in Eq. 3–14 can now be designed as

$$u \triangleq -Y - \hat{g}(x_d) - \mu_a, \tag{3–15}$$

where $\mu_a(t) \in \mathbb{R}^m$ denotes the RISE feedback term defined as [46, 47]

$$\mu_a \triangleq (k_a + 1)e_n(t) - (k_a + 1)e_n(0) + v, \tag{3–16}$$

where $v(t) \in \mathbb{R}^m$ is the generalized solution to

$$\dot{v} = (k_a + 1)\alpha_n e_n + \beta_1 sgn(e_n), \quad v(0) = 0 \tag{3–17}$$

where $k_a, \beta_1 \in \mathbb{R}$ are constant positive control gains, and $sgn(\cdot)$ denotes a vector signum function.

Remark 3.2. *Typically, the presence of the function reconstruction error and disturbance terms in Eq. 3–14 would lead to a UUB stability result. The RISE term used in Eq. 3–15 robustly accounts for these terms guaranteeing asymptotic tracking with a continuous controller [71] (i.e., compared with similar results that can be obtained by discontinuous sliding mode control). The derivative of the RISE structure includes a $sgn(\cdot)$ term in Eq. 3–17 which allows it to implicitly learn and cancel terms in the stability analysis that are C^2 with bounded time derivatives.*

Substituting the control input from Eq. 3–15 in Eq. 3–14 yields

$$r = W_a^T \sigma(V_a^T x_a) - \hat{W}_a^T \sigma(\hat{V}_a^T x_a) + S + d + \varepsilon_a - \mu_a. \tag{3–18}$$

To facilitate the subsequent stability analysis, the time derivative of Eq. 3–18 is expressed as

$$\begin{aligned} \dot{r} \;=\; & \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T \dot{x}_a + \tilde{W}_a^T \sigma'(\hat{V}_a^T x_a)\hat{V}_a^T \dot{x}_a + W_a^T \sigma'(V_a^T x_a)V_a^T \dot{x}_a - W_a^T \sigma'(\hat{V}_a^T x_a)\hat{V}_a^T \dot{x}_a \\ & - \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T \dot{x}_a - \dot{\hat{W}}_a^T \sigma(\hat{V}_a^T x_a) - \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\dot{\hat{V}}_a^T x_a + \dot{S} + \dot{d} + \dot{\varepsilon}_a - \dot{\mu}_a, \end{aligned} \tag{3–19}$$

where $\sigma'(\hat{V}_a^T x_a) \equiv d\sigma\left(V_a^T x_a\right)/d\left(V_a^T x_a\right)|_{V_a^T x_a = \hat{V}_a^T x_a}$, and $\tilde{W}_a(t) \in \mathbb{R}^{(N_a+1)\times m}$ and $\tilde{V}_a(t) \in \mathbb{R}^{(mn+1)\times N_a}$ are the mismatch between the ideal and the estimated weights, and are defined as

$$\tilde{V}_a \triangleq V_a - \hat{V}_a, \qquad \tilde{W}_a \triangleq W_a - \hat{W}_a.$$

The weight update laws for the action NN are designed based on the subsequent stability analysis as

$$\begin{aligned}\dot{\hat{W}}_a &\triangleq proj(\Gamma_{aw}\alpha_n\sigma'(\hat{V}_a^T x_a)\hat{V}_a^T\dot{x}_a e_n^T + \Gamma_{aw}\sigma(\hat{V}_a^T x_a)R\hat{W}_c^T\sigma'(\hat{V}_c^T e_n)\hat{V}_c^T)\\ \dot{\hat{V}}_a &= proj(\Gamma_{av}\alpha_n\dot{x}_a e_n^T\hat{W}_a^T\sigma'(\hat{V}_a^T x_a) + \Gamma_{av}x_a R\hat{W}_c^T\sigma'(\hat{V}_c^T e_n)\hat{V}_c^T\hat{W}_a^T\sigma'(\hat{V}_a^T x_a)),\end{aligned} \tag{3–20}$$

where $\Gamma_{aw} \in \mathbb{R}^{(N_a+1)\times(N_a+1)}, \Gamma_{av} \in \mathbb{R}^{(mn+1)\times(mn+1)}$ are constant, positive definite, symmetric gain matrices, $R(t) \in \mathbb{R}$ is the subsequently designed reinforcement signal, $proj(\cdot)$ is a smooth projection operator utilized to guarantee that the weight estimates $\hat{W}_a(t)$ and $\hat{V}_a(t)$ remain bounded [72], [73], and $\hat{V}_c(t) \in \mathbb{R}^{m\times N_c}$ and $\hat{W}_c(t) \in \mathbb{R}^{(N_c+1)\times 1}$ are the subsequently introduced weight estimates for the critic NN. The NN weight update law in Eq. 3–20 is composite in the sense that it consists of two terms, one of which is affine in the tracking error $e_n(t)$ and the other in the reinforcement signal $R(t)$.

The update law in Eq. 3–20 can be decomposed into two terms

$$\dot{\hat{W}}_a^T = \chi_{e_n}^W + \chi_R^W \qquad \dot{\hat{V}}_a^T = \chi_{e_n}^V + \chi_R^V. \tag{3–21}$$

Using Assumption 3.2, Eq. 3–13 and the use of projection algorithm in Eq. 3–20, the following bounds can be established

$$\begin{aligned}\left\|\chi_{e_n}^W\right\| &\leq \gamma_1\left\|e_n\right\| \qquad \left\|\chi_R^W\right\| \leq \gamma_2\left|R\right|,\\ \left\|\chi_{e_n}^V\right\| &\leq \gamma_3\left\|e_n\right\| \qquad \left\|\chi_R^V\right\| \leq \gamma_4\left|R\right|,\end{aligned} \tag{3–22}$$

where $\gamma_1, \gamma_2, \gamma_3, \gamma_4 \in \mathbb{R}$ are known positive constants. Substituting Eqs. 3–16, 3–20, and 3–21 in Eq. 3–19, and grouping terms, the following expression is obtained

$$\dot{r} = \tilde{N} + N_R + N - e_n - (k_a+1)r - \beta_1 sgn(e_n), \tag{3–23}$$

where the unknown auxiliary terms $\tilde{N}(t) \in \mathbb{R}^m$ and $N_R(t) \in \mathbb{R}^m$ are defined as

$$\tilde{N} \triangleq \dot{S} + e_n - \chi_{e_n}^W\sigma(\hat{V}_a^T x_a) - \hat{W}_a^T\sigma'(\hat{V}_a^T x_a)\chi_{e_n}^V x_a, \tag{3–24}$$

$$N_R \triangleq -\chi_R^W \sigma(\hat{V}_a^T x_a) - \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\chi_R^V x_a. \tag{3–25}$$

The auxiliary term $N(t) \in \mathbb{R}^m$ is segregated into two terms as

$$N = N_d + N_B, \tag{3–26}$$

where $N_d(t) \in \mathbb{R}^m$ is defined as

$$N_d \triangleq W_a^T \sigma'(V_a^T x_a) V_a^T \dot{x}_a + \dot{d} + \dot{\varepsilon}_a, \tag{3–27}$$

and $N_B(t) \in \mathbb{R}^m$ is further segregated into two terms as

$$N_B = N_{B1} + N_{B2} \tag{3–28}$$

where $N_{B1}(t)$, $N_{B2}(t) \in \mathbb{R}^m$ are defined as

$$N_{B1} \triangleq -W_a^T \sigma'(\hat{V}_a^T x_a)\hat{V}_a^T \dot{x}_a - \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T \dot{x}_a,$$

$$N_{B2} \triangleq \tilde{W}_a^T \sigma'(\hat{V}_a^T x_a)\hat{V}_a^T \dot{x}_a + \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T \dot{x}_a. \tag{3–29}$$

Using the Mean Value Theorem, the following upper bound can be developed [47], [71]

$$\left\| \tilde{N}(t) \right\| \leq \rho_1 \left(\|z\| \right) \|z\|, \tag{3–30}$$

where $z(t) \in \mathbb{R}^{(n+1)m}$ is defined as

$$z \triangleq [e_1^T \quad e_2^T \ldots e_n^T \quad r^T]^T, \tag{3–31}$$

and the bounding function $\rho_1(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function. Using Assumptions 3.2 and 3.3, Eqs. 3–12, 3–13, and 3–20, the following bounds can be developed for Eqs. 3–25-3–29

$$\begin{aligned} \|N_d\| &\leq \zeta_1, \qquad \|N_{B1}\| \leq \zeta_2, \qquad \|N_{B2}\| \leq \zeta_3 \\ \|N\| &\leq \zeta_1 + \zeta_2 + \zeta_3, \qquad \|N_R\| \leq \zeta_4 |R|. \end{aligned} \tag{3–32}$$

The bounds for the time derivative of Eqs. 3–27 and 3–28 can be developed using Assumptions 3.2 and 3.3, Eqs. 3–12 and 3–20

$$\left\|\dot{N}_d\right\| \leq \zeta_5, \qquad \left\|\dot{N}_B\right\| \leq \zeta_6 + \zeta_7 \left\|e_n\right\| + \zeta_8 \left|R\right|, \tag{3–33}$$

where $\zeta_i \in \mathbb{R}$, $(i = 1, 2, ..., 8)$ are computable positive constants.

Remark 3.3. *The segregation of the auxiliary terms in Eqs. 3–21 and 3–23 follows a typical RISE strategy [71] which is motivated by the desire to separate terms that can be upper bounded by state-dependent terms and terms that can be upper bounded by constants. Specifically, $\tilde{N}(t)$ contains terms upper bounded by tracking error state-dependent terms, $N(t)$ has terms bounded by a constant, and is further segregated into $N_d(t)$ and $N_B(t)$ whose derivatives are bounded by a constant and linear combination of tracking error states, respectively. Similarly, $N_R(t)$ contains reinforcement signal dependent terms. The terms in Eq. 3–28 are further segregated because $N_{B1}(t)$ will be rejected by the RISE feedback, whereas $N_{B2}(t)$ will be partially rejected by the RISE feedback and partially canceled by the NN weight update law.*

3.4 Critic NN Architecture

In RL literature [2], the critic generates a scalar evaluation signal which is then used to tune the action NN. The critic itself consists of a NN which approximates an evaluation function based on some performance measure. The proposed AC architecture is shown in Fig. 3-1. The filtered tracking error $e_n(t)$ can be considered as an instantaneous utility function of the plant performance [43, 44].

The reinforcement signal $R(t) \in \mathbb{R}$ is defined as [43]

$$R \triangleq \hat{W}_c^T \sigma(\hat{V}_c^T e_n) + \psi, \tag{3–34}$$

where $\hat{V}_c \in \mathbb{R}^{m \times N_c}$, $\hat{W}_c \in \mathbb{R}^{(N_c+1)\times 1}$, $\sigma(\cdot) \in \mathbb{R}^{N_c+1}$ is the nonlinear activation function, N_c are the number of hidden layer neurons of the critic NN, and the performance measure $e_n(t)$ defined in Eq. 3–3 is the input to the critic NN, and $\psi \in \mathbb{R}$ is an auxilliary term

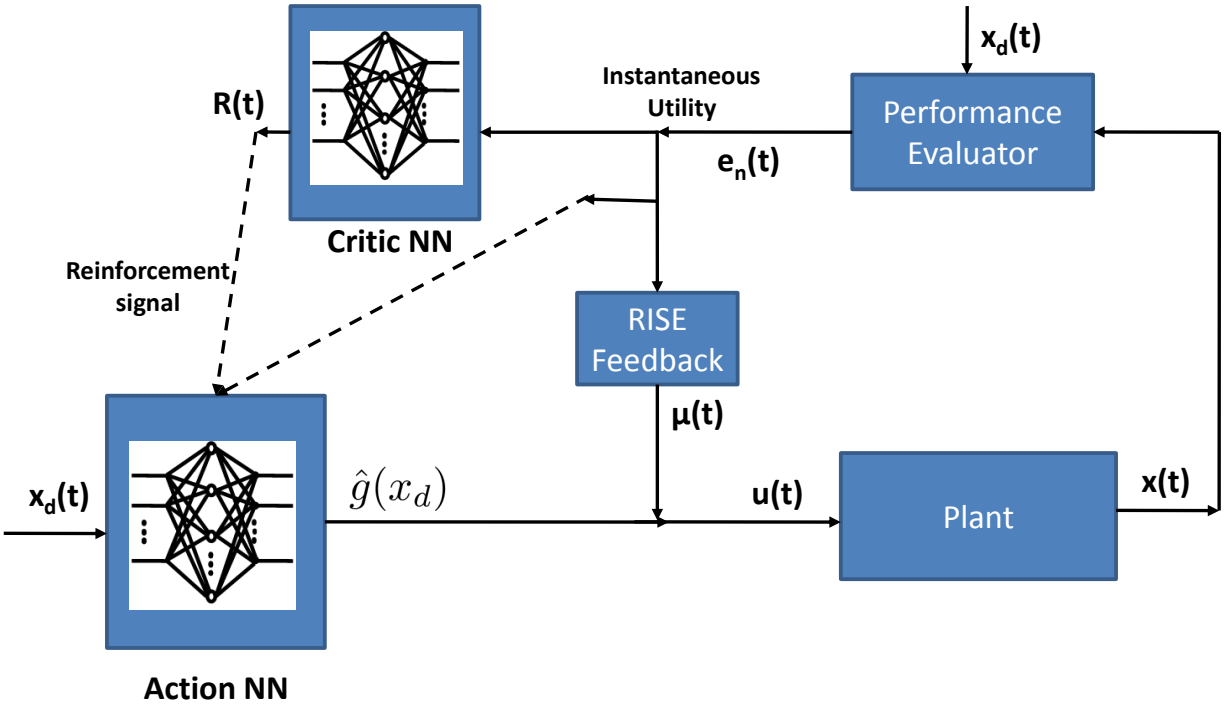

Figure 3-1. Architecture of the RISE-based AC controller.

generated as

$$\dot{\psi} = \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T(\mu_a + \alpha_n e_n) - k_c R - \beta_2 sgn(R), \tag{3–35}$$

where $k_c, \beta_2 \in \mathbb{R}$ are constant positive control gains. The weight update law for the critic NN is generated based on the subsequent stability analysis as

$$\begin{aligned} \dot{\hat{W}}_c &= proj(-\Gamma_{cw}\sigma(\hat{V}_c^T e_n)R - \Gamma_{cw}\hat{W}_c) \\ \dot{\hat{V}}_c &= proj(-\Gamma_{cv} e_n \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)R - \Gamma_{cv}\hat{V}_c), \end{aligned} \tag{3–36}$$

where $\Gamma_{cw}, \Gamma_{cv} \in \mathbb{R}$ are constant positive control gains.

Remark 3.4. *The structure of the reinforcement signal $R(t)$ in Eq. 3–34 is motivated by literature such as [43–45], where the reinforcement signal is typically the output of a critic NN which tunes the actor based on a performance measure. The performance measure considered in this work is the tracking error $e_n(t)$, and the critic weight update laws are*

designed using a gradient algorithm to minimize the tracking error, as seen from the subsequent stability analysis. The auxiliary term $\psi(t)$ in 3–34 is a RISE-like robustifying term which is added to account for certain disturbance terms which appear in the error system of the reinforcement learning signal. Specifically, the inclusion of $\psi(t)$ is used to implicitly learn and compensate for disturbances and function reconstruction errors in the reinforcement signal dynamics, yielding an asymptotic tracking result.

To aid the subsequent stability analysis, the time derivative of the reinforcement signal in Eq. 3–34 is obtained as

$$\dot{R} = \dot{\hat{W}}_c^T \sigma(\hat{V}_c^T e_n) + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\dot{\hat{V}}_c^T e_n + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \dot{e}_n + \dot{\psi}. \tag{3–37}$$

Using Eqs. 3–18, 3–35, 3–36, and the Taylor series expansion [18]

$$\sigma(V_a^T x_a) = \sigma\left(\hat{V}_a^T x_a\right) + \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T x_a + O\left(\tilde{V}_a^T x_a\right)^2,$$

where $O(\cdot)^2$ represents higher order terms, the expression in Eq. 3–37 can be written as

$$\begin{aligned} \dot{R} = {} & \dot{\hat{W}}_c^T \sigma(\hat{V}_c^T e_n) + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\dot{\hat{V}}_c^T e_n + N_{dc} + N_s + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \tilde{W}_a^T \sigma\left(\hat{V}_a^T x_a\right) \\ & + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T x_a - k_c R - \beta_2 sgn(R), \end{aligned} \tag{3–38}$$

where the auxiliary terms $N_{dc}(t) \in \mathbb{R}$ and $N_s(t) \in \mathbb{R}$ are unknown functions defined as

$$\begin{aligned} N_{dc} &\triangleq \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \tilde{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T x_a + W_a^T O\left(\tilde{V}_a^T x_a\right)^2 + d + \varepsilon_a \\ N_s &\triangleq \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T S. \end{aligned} \tag{3–39}$$

Using Assumptions 3.2 and 3.3, Eqs. 3–12, 3–36, and the Mean Value Theorem, the following bounds can be developed for Eq. 3–39

$$\|N_{dc}\| \leq \zeta_9, \quad \|N_s\| \leq \rho_2(\|z\|)\|z\|, \tag{3–40}$$

where $\zeta_9 \in \mathbb{R}$ is a computable positive constant, and $\rho_2(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function.

3.5 Stability Analysis

Theorem 3.1. *The RISE-based AC controller given in Eqs. 3–15 and 3–34 along with the weight update laws for the action and critic NN given in Eqs. 3–20 and 3–36, respectively, ensure that all system signals are bounded under closed-loop operation and that the tracking error is regulated in the sense that*

$$\|e_1(t)\| \to 0 \qquad as \qquad t \to \infty$$

provided the control gains k_a and k_c are selected sufficiently large based on the initial conditions of the states, α_{n-1}, α_n, β_2, and k_c, are chosen according to the following sufficient conditions

$$\alpha_{n-1} > \frac{1}{2}, \qquad \alpha_n > \beta_3 + \frac{1}{2}, \qquad \beta_2 > \zeta_9, \qquad k_c > \beta_4, \tag{3–41}$$

and $\beta_1, \beta_3, \beta_4 \in \mathbb{R}$, introduced in Eq. 3–46, are chosen to satisfy the following sufficient conditions [2]

$$\beta_1 \;>\; \max\left(\zeta_1 + \zeta_2 + \zeta_3, \zeta_1 + \zeta_2 + \frac{\zeta_5}{\alpha_n} + \frac{\zeta_6}{\alpha_n}\right), \quad \beta_3 > \zeta_7 + \frac{\zeta_8}{2}, \quad \beta_4 > \frac{\zeta_8}{2}. \tag{3–42}$$

Proof. Let $\mathcal{D} \subset \mathbb{R}^{(n+1)m+3}$ be a domain containing $y(t) = 0$, where $y(t) \in \mathbb{R}^{(n+1)m+3}$ is defined as

$$y \triangleq [z^T \;\; R \;\; \sqrt{P} \;\; \sqrt{Q}]^T, \tag{3–43}$$

where the auxiliary function $Q(t) \in \mathbb{R}$ is defined as

$$Q \;\triangleq\; \frac{1}{2}tr(\tilde{W}_a^T \Gamma_{aw}^{-1} \tilde{W}_a) + \frac{1}{2}tr(\tilde{V}_a^T \Gamma_{av}^{-1} \tilde{V}_a) + \frac{1}{2}tr(\hat{W}_c^T \hat{W}_c) + \frac{1}{2}tr(\hat{V}_c^T \hat{V}_c), \tag{3–44}$$

[2] The derivation of the sufficient conditions in Eq. 3–42 is provided in the Appendix A.1.

where $tr(\cdot)$ is the trace of a matrix. The auxiliary function $P(z,R,t) \in \mathbb{R}$ in Eq. 3–43 is the generalized solution to the differential equation

$$\dot{P} = -L, \quad P(0) = \beta_1 \sum_{i=1}^{m} |e_{ni}(0)| - e_n(0)^T N(0), \tag{3–45}$$

where the subscript $i = 1, 2, .., m$ denotes the ith element of the vector, and the auxiliary function $L(z,R,t) \in \mathbb{R}$ is defined as

$$L \triangleq r^T(N_d + N_{B1} - \beta_1 sgn(e_n)) + \dot{e}_n^T N_{B2} - \beta_3 \|e_n\|^2 - \beta_4 |R|^2, \tag{3–46}$$

where $\beta_1, \beta_3, \beta_4 \in \mathbb{R}$ are chosen according to the sufficient conditions in Eq. 3–42. Provided the sufficient conditions introduced in Eq. 3–42 are satisfied, then $P(z,R,t) \geq 0$. From Eqs. 3–23, 3–32, 3–38 and 3–40, some disturbance terms in the closed-loop error systems are bounded by a constant. Typically, such terms (e.g., NN reconstruction error) lead to a UUB stability result. The definition of $P(z,R,t)$ is motivated by the RISE control structure to compensate for such disturbances so that an asymptotic tracking result is obtained.

Let $V(y) : \mathcal{D} \times [0,\infty) \to \mathbb{R}$ be a Lipschitz continuous regular positive definite function defined as

$$V \triangleq \frac{1}{2} z^T z + \frac{1}{2} R^2 + P + Q \tag{3–47}$$

which satisfies the following inequalities:

$$U_1(y) \leq V(y) \leq U_2(y) \tag{3–48}$$

where $U_1(y), U_2(y) \in \mathbb{R}$ are continuous positive definite functions. From Eqs. 3–3, 3–4, 3–23, 3–38, 3–44, and 3–45, the differential equations of the closed-loop system are continuous except in the set $\{(y,t) | e_n = 0 \ or \ R = 0\}$. Using Filippov's differential inclusion [74–77], the existence and uniqueness of solutions can be established for $\dot{y} = f(y,t)$ (a.e.), where $f(y,t) \in \mathbb{R}^{(n+1)m+3}$ denotes the right-hand side of the the closed-loop error signals. Under Filippov's framework, a generalized Lyapunov stability theory can be

used (see [77–80] and Appendix A.2 for further details) to establish strong stability of the closed-loop system. The generalized time derivative of Eq. 3–47 exists almost everywhere (a.e.), and $\dot{V}(y) \in^{a.e.} \dot{\tilde{V}}(y)$ where

$$\dot{\tilde{V}} = \bigcap_{\xi \in \partial V(y)} \xi^T K \left[\begin{array}{ccccc} \dot{z}^T & \dot{R} & \frac{1}{2}P^{-\frac{1}{2}}\dot{P} & \frac{1}{2}Q^{-\frac{1}{2}}\dot{Q} & 1 \end{array} \right]^T,$$

where ∂V is the generalized gradient of V [78], and $K[\cdot]$ is defined as [79, 80]

$$K[f](y,t) \triangleq \bigcap_{\delta>0} \bigcap_{\mu N=0} \overline{co} f(B(y,\delta) - N, t), \tag{3–49}$$

where $\bigcap_{\mu N=0}$ denotes the intersection of all sets N of Lebesgue measure zero, $\overline{co}$ denotes convex closure, and $B(y,\delta)$ represents a ball of radius δ around y. Since $V(y)$ is a Lipschitz continuous regular function,

$$\begin{aligned} \dot{\tilde{V}} &= \nabla V^T K \left[\begin{array}{ccccc} \dot{z}^T & \dot{R} & \frac{1}{2}P^{-\frac{1}{2}}\dot{P} & \frac{1}{2}Q^{-\frac{1}{2}}\dot{Q} & 1 \end{array} \right]^T \\ &= \left[\begin{array}{ccccc} z^T & R & 2P^{\frac{1}{2}} & 2Q^{\frac{1}{2}} & 0 \end{array} \right] K \left[\begin{array}{ccccc} \dot{z}^T & \dot{R} & \frac{1}{2}P^{-\frac{1}{2}}\dot{P} & \frac{1}{2}Q^{-\frac{1}{2}}\dot{Q} & 1 \end{array} \right]^T \end{aligned}$$

Using the calculus for $K[\cdot]$ from [80] and substituting the dynamics from Eqs. 3–23, 3–38, 3–44, and 3–45, and splitting k_c as $k_c = k_{c1} + k_{c2}$, yields

$$\begin{aligned} \dot{\tilde{V}} \subset\ & r^T(\tilde{N} + N_R + N - e_n - (k_a+1)r - \beta_1 K[sgn(e_n)]) + \sum_{i=1}^{n} e_i^T \dot{e}_i \\ & + R(\dot{\hat{W}}_c^T \sigma(\hat{V}_c^T e_n) + N_{dc} + N_s) + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\dot{\hat{V}}_c^T e_n R - \beta_2 R K[sgn(R)] \\ & + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \tilde{W}_a^T \sigma\left(\hat{V}_a^T x_a\right) R - k_c R^2 + \hat{W}_c^T \sigma'(\hat{V}_c^T e_n)\hat{V}_c^T \hat{W}_a^T \sigma'(\hat{V}_a^T x_a)\tilde{V}_a^T x_a R \\ & - r^T(N_d + N_{B1} - \beta_1 K[sgn(e_n)]) - \dot{e}_n(t)^T N_{B2} + \beta_3 \|e_n\|^2 + \beta_4 |R|^2 \\ & - \frac{1}{2} tr(\tilde{W}_a^T \Gamma_{aw}^{-1} \dot{\hat{W}}_a) - \frac{1}{2} tr(\tilde{V}_a^T \Gamma_{av}^{-1} \dot{\hat{V}}_a) - \frac{1}{2} tr(\hat{W}_c^T \dot{\hat{W}}_c) - \frac{1}{2} tr(\hat{V}_c^T \dot{\hat{V}}_c). \end{aligned}$$

$$\begin{aligned} = & -\sum_{i=1}^{n} \alpha_i \|e_i\|^2 + e_{n-1}^T e_n - \|r\|^2 - (k_{c1} + k_{c2})|R|^2 + r^T(\tilde{N} + N_R - k_a r) \\ & + R(N_{dc} + N_s - k_c R) - \beta_2 |R| - \Gamma_{cw} |R|^2 \left\| \sigma(\hat{V}_c^T e_n) \right\|^2 - \Gamma_{cw} \left\| \hat{W}_c \right\|^2 \end{aligned}$$

$$+2\Gamma_{cw}\left|R\right|\left\|\hat{W}_c\sigma(\hat{V}_c^T e_n)\right\| + \beta_3\left\|e_n\right\|^2 + \beta_4\left|R\right|^2 - \Gamma_{cv}\left\|\hat{W}_c^T\sigma'(\hat{V}_c^T e_n)\right\|^2\left\|e_n\right\|^2\left|R\right|^2$$
$$-\Gamma_{cv}(\left\|\hat{V}_c\right\|^2 - 2\left\|\hat{W}_c^T\sigma'(\hat{V}_c^T e_n)\right\|\left\|e_n\right\|\left\|\hat{V}_c\right\|\left|R\right|), \tag{3–50}$$

where the NN weight update laws from Eqs. 3–20, 3–36, and the fact that $(r^T - r^T)_i SGN(e_{ni}) = 0$ is used (the subscript i denotes the i^{th} element), where $K[sgn(e_n)] = SGN(e_n)$ [80], such that $SGN(e_{ni}) = 1$ if $e_{ni} > 0$, $[-1, 1]$ if $e_{ni} = 0$, and -1 if $e_{ni} < 0$. Upper bounding the expression in Eq. 3–50 using Eqs. 3–30, 3–32, and 3–40, yields

$$\begin{aligned}\dot{\tilde{V}} \leq & -\sum_{i=1}^{n-2}\alpha_i\left\|e_i\right\|^2 - \left(\alpha_{n-1} - \frac{1}{2}\right)\left\|e_{n-1}\right\|^2 - \left\|r\right\|^2 - \left(\alpha_n - \beta_3 - \frac{1}{2}\right)\left\|e_n\right\|^2 \\ & -(k_{c1} - \beta_4)\left|R\right|^2 + (\zeta_9 - \beta_2)\left|R\right| - [k_a\left\|r\right\|^2 - \rho_1\left(\left\|z\right\|\right)\left\|z\right\|\left\|r\right\|] \\ & -[k_{c2}\left|R\right|^2 - (\rho_2(\left\|z\right\|) + \zeta_4)\left|R\right|\left\|z\right\|]. \end{aligned} \tag{3–51}$$

Provided the gains are selected according to Eq. 3–41, the expression in Eq. 3–51 can be further upper bounded by completing the squares as

$$\begin{aligned}\dot{\tilde{V}} &\leq -\lambda\left\|z\right\|^2 + \frac{\rho^2\left(\left\|z\right\|\right)\left\|z\right\|^2}{4k} - (k_{c1} - \beta_4)\left|R\right|^2 \\ &\leq -U(y) \quad \forall y \in \mathcal{D}, \end{aligned} \tag{3–52}$$

where $k \triangleq \min(k_a, k_{c2})$ and $\lambda \in \mathbb{R}$ is a positive constant defined as

$$\lambda = \min\left\{\alpha_1, \alpha_2, ..., \alpha_{n-2}, \left(\alpha_{n-1} - \frac{1}{2}\right), \left(\alpha_n - \beta_3 - \frac{1}{2}\right), 1\right\}.$$

In Eq. 3–52, $\rho(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function defined as

$$\rho^2\left(\left\|z\right\|\right) = \rho_1^2\left(\left\|z\right\|\right) + \left(\rho_2\left(\left\|z\right\|\right) + \zeta_4\right)^2,$$

and $U(y) \triangleq c\left\|\begin{bmatrix} z^T & R \end{bmatrix}^T\right\|^2$, for some positive constant c, is a continuous, positive semi-definite function defined on the domain

$$\mathcal{D} \triangleq \left\{y(t) \in \mathbb{R}^{(n+1)m+3} \mid \left\|y\right\| \leq \rho^{-1}\left(2\sqrt{\lambda k}\right)\right\}.$$

The size of the domain $\mathcal{D}$ can be increased by increasing k. The result in Eq. 3–52 indicates that $\dot{V}(y) \leq -U(y) \; \forall \; \dot{V}(y) \in \dot{\tilde{V}}(y) \; \forall y \in \mathcal{D}$. The inequalities in Eqs. 3–48 and 3–52 can be used to show that $V(y) \in \mathcal{L}_\infty$ in $\mathcal{D}$; hence, $e_1(t)$, $e_2(t)$, $...e_n(t)$, $r(t)$ and $R(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. Standard linear analysis methods can be used along with Eqs. 3–1-3–5 to prove that $\dot{e}_1(t)$, $\dot{e}_2(t)$, $...\dot{e}_n(t)$, $x^{(i)}(t) \in \mathcal{L}_\infty$ $(i = 0, 1, 2)$ in $\mathcal{D}$. Further, Assumptions 3.1 and 3.3 can be used to conclude that $u(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. From these results, Eqs. 3–12, 3–13, 3–19, 3–20, and 3–34-3–37 can be used to conclude that $\dot{r}(t), \psi(t), \dot{R}(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. Hence, $U(y)$ is uniformly continuous in $\mathcal{D}$. Let $\mathcal{S} \subset \mathcal{D}$ denote a set defined as follows:

$$\mathcal{S} \triangleq \left\{ y(t) \subset \mathcal{D} \mid U_2(y(t)) < \lambda_1 \left(\rho^{-1} \left(2\sqrt{\lambda k} \right) \right)^2 \right\}. \tag{3–53}$$

The region of attraction in Eq. 3–53 can be made arbitrarily large to include any initial conditions by increasing the control gain k (i.e. a semi-global type of stability result), and hence

$$\|e_1(t)\|, |R| \to 0 \qquad \text{as} \qquad t \to \infty \qquad \forall y(0) \in \mathcal{S}.$$

□

3.6 Experimental Results

To test the performance of the proposed AC-based approach, the controller in Eqs. 3–15, 3–20, 3–34-3–36 was implemented on a two-link robot manipulator, where two aluminum links are mounted on a 240 Nm (first link) and a 20 Nm (second link) switched reluctance motor. The motor resolvers provide rotor position measurements with a resolution of 614400 pulses/revolution, and a standard backwards difference algorithm is used to numerically determine angular velocity from the encoder readings (Fig. 3-2). The two-link revolute robot is modeled as an Euler-Lagrange system with the following dynamics

$$M(q)\ddot{q} + V_m(q, \dot{q})\dot{q} + F(\dot{q}) + \tau_d = \tau, \tag{3–54}$$

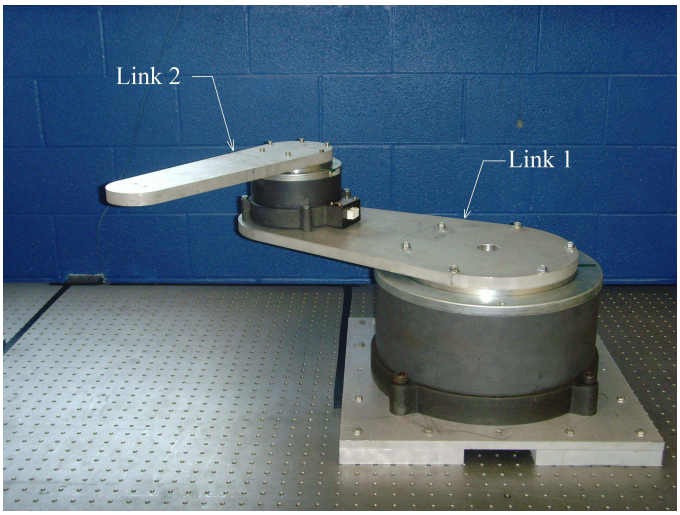

Figure 3-2. Two-link experiment testbed.

where $M(q) \in \mathbb{R}^{2\times 2}$ denotes the inertia matrix, $V_m(q,\dot{q}) \in \mathbb{R}^{2\times 2}$ denotes the centripetal-Coriolis matrix, $F(\dot{q}) \in \mathbb{R}^2$ denotes friction, $\tau_d(t) \in \mathbb{R}^2$ denotes an unknown external disturbance, $\tau(t) \in \mathbb{R}^2$ represents the control torque, and $q(t), \dot{q}(t), \ddot{q}(t) \in \mathbb{R}^2$ denote the link position, velocity and acceleration. The dynamics in (3–54) can be transformed into the Brunovsky form as

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= g(x) + u + d, \end{aligned} \tag{3–55}$$

where $x_1 \triangleq q$, $x_2 \triangleq \dot{q}$, $x = [x_1 \;\; x_2]^T$, $g(x) \triangleq -M^{-1}(q)[V_m(q,\dot{q})\dot{q} + F(\dot{q})]$, $u \triangleq M^{-1}(q)\tau(t)$, and $d \triangleq M^{-1}(q)\tau(t)$. The control objective is to track a desired link trajectory, selected as (in degrees):

$$q_d(t) = 60\sin(2.5t)(1 - e^{-0.01t^3}).$$

Two controllers are implemented on the system, both having the same expression for the control $u(t)$ as in Eq. 3–15; however, they differ in the NN weight update laws. The first

[1] For this experiment, the inertia matrix is assumed to be known as it is required for calculation of joint torques $\tau(t)$, which are determined using the expression $\tau = M(q)u$.

controller (denoted by NN+RISE) employs a standard NN gradient-based weight update law which is affine in the tracking error, given as

$$\dot{\hat{W}}_a \triangleq \Gamma_{aw}\left[proj(\alpha_n \sigma'(\hat{V}_a^T x_a)\hat{V}_a^T \dot{x}_a e_n^T)\right]$$
$$\dot{\hat{V}}_a = \Gamma_{av}\left[proj(\alpha_n \dot{x}_a e_n^T \hat{W}_a^T \sigma'(\hat{V}_a^T x_a))\right].$$

The proposed AC-based controller (denoted by AC+RISE) uses a composite weight update law, consisting of a gradient-based term and a reinforcement-based term, as in Eq. 3–20, where the reinforcement term is generated from the critic architecture in Eq. 3–34. For the NN+RISE controller, the initial weights of the NN, $\hat{W}_a(0)$ is chosen to be zero, whereas $\hat{V}_a(0)$ is randomly initialized in $[-1, 1]$, such that it forms a basis [81]. The input to the action NN is chosen as $x_a = [1 \quad q_d^T \quad \dot{q}_d^T]$, and the number of hidden layer neurons are chosen by trial and error as $N_a = 10$. All other states are initialized to zero. A sigmoid activation function is chosen for the NN and the adaptation gains are selected as $\Gamma_{aw} = I_{11} \qquad \Gamma_{av} = 0.1 I_{11}$, with feedback gains selected as $\alpha_1 = diag(10, 15)$, $\alpha_2 = diag(20, 15)$, $k_a = (20, 15)$ and $\beta_1 = diag(2, 1)$. For the AC+RISE controller, the critic is added to the NN+RISE by including an additional RL term in the weight update law of the action NN. The actor NN and the RISE term in AC+RISE use the same gains as NN+RISE. The number of hidden layer neurons for the critic are selected by trial and error as $N_c = 3$. The initial critic NN weights $\hat{W}_c(0)$ and $\hat{V}_c(0)$ are randomly chosen in $[-1, 1]$. The control gains for the critic are selected as $k_c = 5 \quad \beta_2 = 0.1 \quad \Gamma_{cw} = 0.4 \quad \Gamma_{cv} = 1$. Experiments for both controllers were repeated 10 consecutive times with the same gains to check the repeatability and accuracy of results. For each run, the RMS values of the tracking error $e_1(t)$ and torques $\tau(t)$ are calculated. A one-tail unpaired t-test is performed with a significance level of $\alpha = 0.05$. A summary of comparative results with the two controllers are tabulated in Tables 3-1 and 3-2.

Tables 3-1 and 3-2 indicate that the AC+RISE controller has statistically smaller mean RMS errors for Link 1 ($P = 0.003$) and Link 2 ($P = 0.046$) as compared to

Table 3-1. Summarized experimental results and P values of one tailed unpaired t-test for Link 1.

Experiment	RMS error [Link1]		Torque [Link1] (Nm)	
	NN+RISE	AC+RISE	NN+RISE	AC+RISE
Maximum	0.143°	0.123°	15.937	16.013
Minimum	0.101°	0.098°	15.451	15.470
Mean	0.125°	0.108°	15.687	15.764
Std. dev.	0.014°	0.009°	0.152	0.148
P(T<=t)		0.003*		0.134

∗ denotes statistically significant value.

Table 3-2. Summarized experimental results and P values of one tailed unpaired t-test for Link 2.

Experiment	RMS error [Link2]		Torque [Link2] (Nm)	
	NN+RISE	AC+RISE	NN+RISE	AC+RISE
Maximum	0.161°	0.138°	1.856	1.858
Minimum	0.112°	0.107°	1.717	1.670
Mean	0.137°	0.127°	1.783	1.753
Std. dev.	0.015°	0.010°	0.045	0.054
P(T<=t)		0.046*		0.098

∗ denotes statistically significant value.

the NN+RISE controller. The AC+RISE controller, while having a reduced error, uses approximately the same amount of control torque (statistically insignificant difference) as NN+RISE. The results indicate that the mean RMS the position tracking errors for Link 1 and Link 2 are approximately 14% and 7% smaller for the proposed AC+RISE controller. The plots for tracking error and control torques are shown for a typical experiment in Figs. 3-3 and 3-4.

3.7 Comparison with Related Work

A continuous asymptotic AC-based tracking controller is developed for a class of nonlinear systems with bounded disturbances. The approach is different from the optimal control-based ADP approaches proposed in literature [8–10, 13, 14, 32, 42], where the critic usually approximates a long-term cost function and the actor approximates the optimal control. However, the similarity with the ADP-based methods is in the use of the AC architecture, borrowed from RL, where the critic, through a reinforcement signal affects the behavior of the actor leading to an improved performance. The

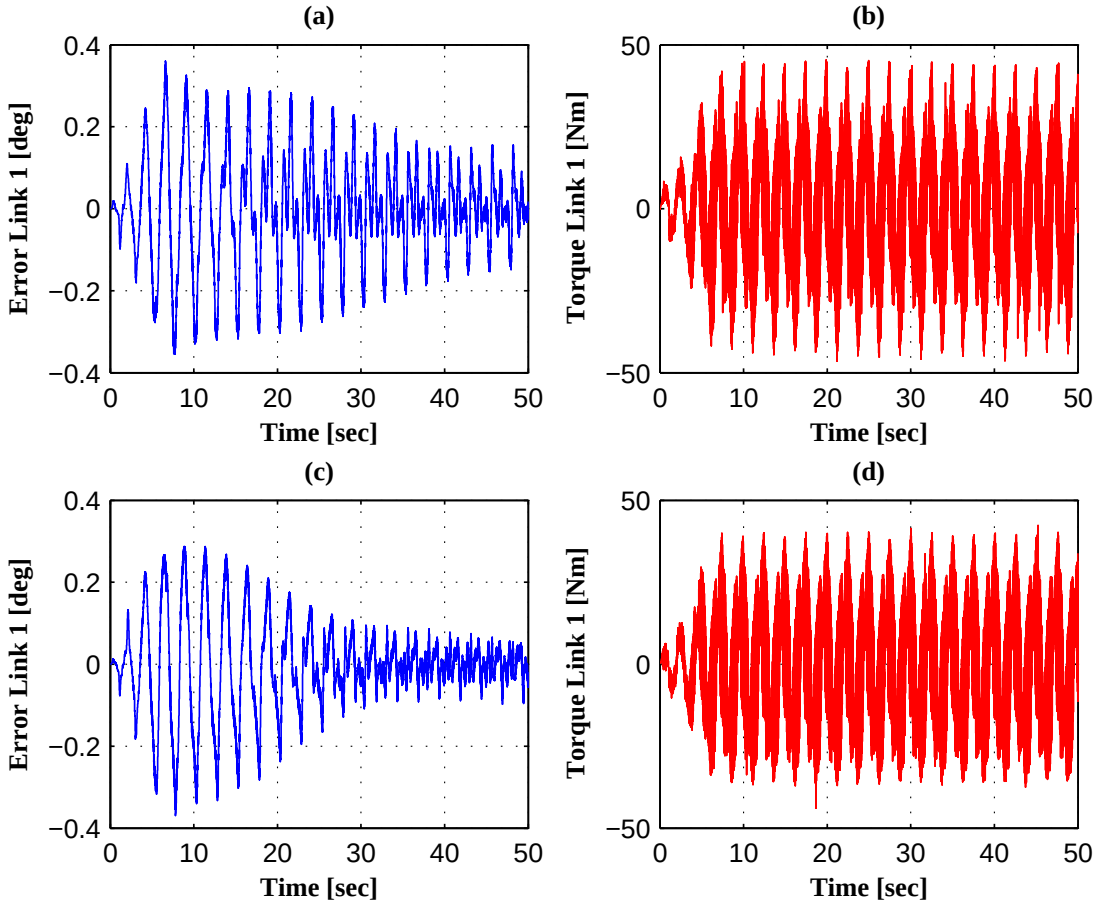

Figure 3-3. Comparison of tracking errors and torques between NN+RISE and AC+RISE for link 1 (a) Tracking error with NN+RISE, (b) Control Torque with NN+RISE, (c) Tracking error with AC+RISE, (d) Control Torque with AC+RISE.

proposed adaptive robust controller consists of a NN feedforward term (actor NN) and a robust feedback term, where the weight update laws of the actor NN are designed as a composite of a tracking error term and a RL term (from the critic), with the objective of minimizing the tracking error [43–45]. The robust term is designed to withstand the external disturbances and modeling errors in the plant. Typically, the presence of bounded disturbances and NN approximation errors lead to a UUB result. The main contribution of this work is the use of a recently developed continuous feedback technique, RISE [46, 47], in conjunction with the AC architecture to yield asymptotic tracking of

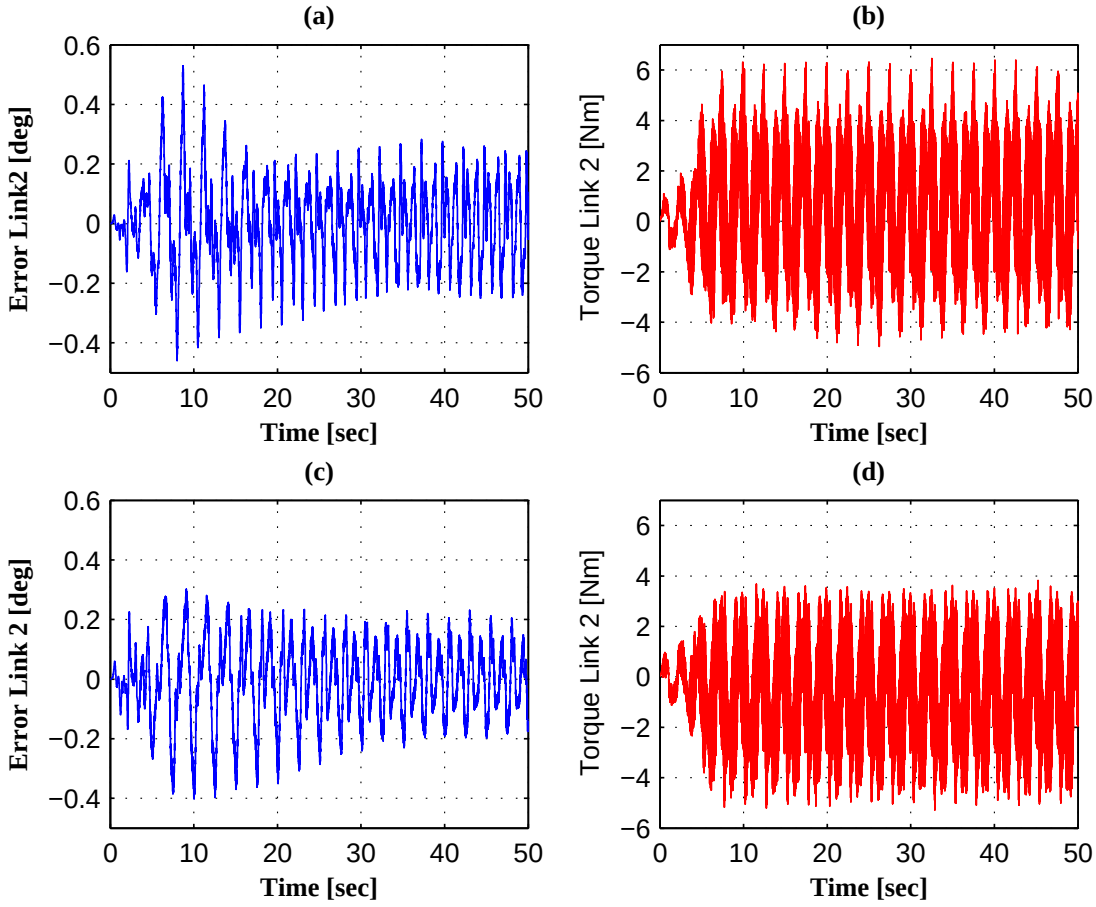

Figure 3-4. Comparison of tracking errors and torques between NN+RISE and AC+RISE for link 2 (a) Tracking error with NN+RISE, (b) Control Torque with NN+RISE, (c) Tracking error with AC+RISE, (d) Control Torque with AC+RISE.

an unknown nonlinear system subjected to bounded external disturbances. The use of RISE in conjunction with the action NN makes the design of the critic NN architecture challenging from a stability standpoint. To this end, the critic NN is combined with an additional RISE-like term to yield a reinforcement signal, which is used to update the weights of the action NN. A smooth projection algorithm is used to bound the NN weight estimates and a Lyapunov stability analysis guarantees closed-loop stability of the system.

3.8 Summary

An AC-based controller is developed for a class of uncertain nonlinear systems with additive bounded disturbances. The main contribution of this work is the combination of the continuous RISE feedback with the AC architecture to guarantee asymptotic tracking for the nonlinear system. The feedforward action NN approximates the nonlinear system dynamics and the robust feedback (RISE) rejects the NN functional reconstruction error and disturbances. In addition, the action NN is trained online using a combination of tracking error and a reinforcement signal, generated by the critic. Experimental results and t-test analysis demonstrate faster convergence of the tracking error when a RL term is included in the NN weight update laws.

CHAPTER 4
ROBUST IDENTIFICATION-BASED STATE DERIVATIVE ESTIMATION FOR NONLINEAR SYSTEMS

The requirement of complete model knowledge has impeded the development of RL-based optimal control solutions for continuous-time uncertain nonlinear systems, which motivates the development of state derivative estimator in this chapter. Besides providing a model-free value function approximation in RL-based control, estimation of the state derivative is useful for many other applications including: disturbance and parameter estimation [82], fault detection in dynamical systems [83], digital differentiation in signal processing, acceleration feedback in robot contact transition control [84], DC motor control [85] and active vibration control [86]. The problem of computing the state derivative becomes trivial if the state is fully measurable and the system dynamics are exactly known. The presence of uncertainties (parametric and non-parametric) and exogenous disturbances, however, make the problem challenging and motivate the state derivative estimation method for uncertain nonlinear systems developed in this work.

4.1 Robust Identification-Based State Derivative Estimation

Consider a control-affine uncertain nonlinear system

$$\dot{x} = f(x) + \sum_{i=1}^{m} g_i(x)u_i + d, \tag{4–1}$$

where $x(t) \in \mathbb{R}^n$ is the measurable system state, $f(x) \in \mathbb{R}^n$ and $g_i(x) \in \mathbb{R}^n$, $i = 1, ..., m$ are unknown functions, $u_i(t) \in \mathbb{R}$, $i = 1, ..., m$ is the control input, and $d(t) \in \mathbb{R}^n$ is an exogenous disturbance. The objective is to design an estimator for the state derivative $\dot{x}(t)$ using a robust identification-based approach that adaptively identifies the uncertain dynamics.

Assumption 4.1. *The functions $f(x)$ and $g_i(x)$, $i = 1, ..., m$ are second-order differentiable.*

Assumption 4.2. *The system in Eq. 4–1 is bounded input bounded state (BIBS) stable, i.e.,* $u_i(t), x(t) \in \mathcal{L}_\infty$, $i = 1, ..., m$. *Also,* $u_i(t)$ *is second order differentiable, and* $\dot{u}_i(t), \ddot{u}_i(t) \in \mathcal{L}_\infty$ $i = 1, ..., m$.

Assumption 4.3. *The disturbance* $d(t)$ *is second order differentiable, and* $d(t), \dot{d}(t), \ddot{d}(t) \in \mathcal{L}_\infty$.

Assumption 4.4. *Given a continuous function* $F : \mathbb{S} \to \mathbb{R}^n$, *where* $\mathbb{S}$ *is a compact simply connected set, there exists ideal weights* θ, *such that the output of the NN, denoted by* $\hat{F}(\cdot, \theta)$, *approximates* $F(\cdot)$ *to an arbitrary accuracy [15].*

Remark 4.1. *Assumptions 4.1-4.3 indicate that the technique developed in this work is only applicable for sufficiently smooth systems (i.e. at least second-order differentiable) that are BIBS stable. The requirement that the disturbance is* C^2 *can be restrictive. For example, random noise does not satisfy this assumption; however, simulations with added noise show robustness to these disturbances as well. Assumption 4.4 states the universal approximation property of the NNs which is proved for sigmoidal activation functions in [15]. Since* $x(t)$ *is assumed to be bounded (Assumption 4.2), the functions* $f(x)$ *and* $g(x)$ *can be defined on a compact set; hence, the NN universal approximation property (Assumption 4.4) holds.*

Using Assumption 4.4, the dynamic system in Eq. 4–1 can be represented by replacing the unknown functions with multi-layer NNs, as

$$\dot{x} = W_f^T \sigma(V_f^T x) + \varepsilon_f(x) + \sum_{i=1}^{m} \left[W_{gi}^T \sigma(V_{gi}^T x) + \varepsilon_{gi}(x) \right] u_i + d, \tag{4–2}$$

where $W_f \in \mathbb{R}^{L_f+1 \times n}$, $V_f \in \mathbb{R}^{n \times L_f}$, $W_{gi} \in \mathbb{R}^{L_{gi}+1 \times n}$, $V_{gi} \in \mathbb{R}^{n \times L_{gi}}$, $i = 1, ..., m$ are the unknown ideal NN weights, $\sigma_f \triangleq \sigma(V_f^T x) \in \mathbb{R}^{L_f+1}$ and $\sigma_{gi} \triangleq \sigma(V_{gi}^T x) \in \mathbb{R}^{L_{gi}+1}$ are the NN activation functions, and $\varepsilon_f(x) \in \mathbb{R}^n$ and $\varepsilon_{gi}(x) \in \mathbb{R}^n$ are the function reconstruction errors.

Assumption 4.5. *The ideal weights are bounded by known positive constants [18], i.e.* $\|W_f\|_F \leq \bar{W}_f$, $\|V_f\|_F \leq \bar{V}_f$, $\|W_{gi}\|_F \leq \bar{W}_g$ *and* $\|V_{gi}\|_F \leq \bar{V}_g$, $\forall i$.

Assumption 4.6. *The activation functions $\sigma_f(\cdot)$ and $\sigma_{gi}(\cdot)$, and their derivatives with respect to their arguments, $\sigma'_f(\cdot)$, $\sigma'_{gi}(\cdot)$, $\sigma''_f(\cdot)$, $\sigma''_{gi}(\cdot)$, are bounded with known bounds (e.g., sigmoidal and hyperbolic tangent activation functions).*

Assumption 4.7. *The function reconstruction errors $\varepsilon_f(\cdot)$ and $\varepsilon_{gi}(\cdot)$, and their derivatives with respect to their arguments, $\varepsilon'_f(\cdot)$, $\varepsilon'_{gi}(\cdot)$, $\varepsilon''_f(\cdot)$, $\varepsilon''_{gi}(\cdot)$, are bounded with known bounds [18].*

The following multi-layer dynamic neural network (MLDNN) identifier is proposed to identify the system in Eq. 4–2 and estimate the state derivative

$$\dot{\hat{x}} = \hat{W}_f^T \hat{\sigma}_f + \sum_{i=1}^{m} \hat{W}_{gi}^T \hat{\sigma}_{gi} u_i + \mu, \tag{4–3}$$

where $\hat{x}(t) \in \mathbb{R}^n$ is the identifier state, $\hat{W}_f(t) \in \mathbb{R}^{L_f+1\times n}$, $\hat{V}_f(t) \in \mathbb{R}^{n\times L_f}$, $\hat{W}_{gi}(t) \in \mathbb{R}^{L_{gi}+1\times n}$, $\hat{V}_{gi}(t) \in \mathbb{R}^{n\times L_{gi}}$, $i = 1, ..., m$ are the weight estimates, $\hat{\sigma}_f \triangleq \sigma(\hat{V}_f^T \hat{x}) \in \mathbb{R}^{L_f+1}$, $\hat{\sigma}_{gi} \triangleq \sigma(\hat{V}_{gi}^T \hat{x}) \in \mathbb{R}^{L_{gi}+1}$, $i = 1, ..., m$, and $\mu(t) \in \mathbb{R}^n$ denotes the RISE feedback term defined as [47, 71]

$$\mu \triangleq k\tilde{x}(t) - k\tilde{x}(0) + v,$$

where $\tilde{x}(t) \triangleq x(t) - \hat{x}(t) \in \mathbb{R}^n$ is the identification error, and $v(t) \in \mathbb{R}^n$ is the generalized solution (in Filippov's sense [74]) to

$$\dot{v} = (k\alpha + \gamma)\tilde{x} + \beta_1 sgn(\tilde{x}); \quad v(0) = 0,$$

where k, α, γ, $\beta_1 \in \mathbb{R}$ are positive constant control gains, and $sgn(\cdot)$ denotes a vector signum function.

Remark 4.2. *The DNN-based system identifiers in literature, [87–91], typically do not include a feedback term based on the identification error, except in results such as [92–94], where a high gain proportional feedback term is used to guarantee bounded stability. The novel use of RISE feedback term, $\mu(t)$ in Eq. 4–3, ensures asymptotic regulation of the identification error in the presence of disturbance and NN function approximation errors.*

The identification error dynamics can be written as

$$\dot{\tilde{x}} = W_f^T \sigma_f - \hat{W}_f^T \hat{\sigma}_f + \sum_{i=1}^{m} \left[(W_{gi}^T \sigma_{gi} - \hat{W}_{gi}^T \hat{\sigma}_{gi}) + \varepsilon_{gi}(x) \right] u_i + \varepsilon_f(x) + d - \mu. \quad (4\text{–}4)$$

A filtered identification error is defined as

$$r \triangleq \dot{\tilde{x}} + \alpha \tilde{x}. \quad (4\text{–}5)$$

Taking the time derivative of Eq. 4–5 and using Eq. 4–4 yields

$$\begin{aligned} \dot{r} = {} & W_f^T \sigma_f' V_f^T \dot{x} - \dot{\hat{W}}_f^T \hat{\sigma}_f - \hat{W}_f^T \hat{\sigma}_f' \dot{\hat{V}}_f^T \hat{x} - \hat{W}_f^T \hat{\sigma}_f' \hat{V}_f^T \dot{\hat{x}} + \sum_{i=1}^{m} (W_{gi}^T \sigma_{gi} - \hat{W}_{gi}^T \hat{\sigma}_{gi}) \dot{u}_i \\ & + \sum_{i=1}^{m} \left[W_{gi}^T \sigma_{gi}' V_{gi}^T \dot{x} u_i - \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi} u_i - \hat{W}_{gi}^T \hat{\sigma}_{gi}' \dot{\hat{V}}_{gi}^T \hat{x} u_i - \hat{W}_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{\hat{x}} u_i \right] \\ & + \sum_{i=1}^{m} [\dot{\varepsilon}_{gi}(x) u_i + \varepsilon_{gi}(x) \dot{u}_i] + \dot{\varepsilon}_f(x) + \dot{d} - kr - \gamma \tilde{x} - \beta_1 sgn(\tilde{x}) + \alpha \dot{\tilde{x}}. \end{aligned} \quad (4\text{–}6)$$

The weight update laws for the DNN in Eq. 4–3 are developed based on the subsequent stability analysis as

$$\begin{aligned} \dot{\hat{W}}_f &= proj(\Gamma_{wf} \hat{\sigma}_f' \hat{V}_f^T \dot{\hat{x}} \tilde{x}^T), \qquad \dot{\hat{V}}_f = proj(\Gamma_{vf} \dot{\hat{x}} \tilde{x}^T \hat{W}_f^T \hat{\sigma}_f'), \\ \dot{\hat{W}}_{gi} &= proj(\Gamma_{wgi} \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{\hat{x}} u_i \tilde{x}^T), \quad \dot{\hat{V}}_{gi} = proj(\Gamma_{vgi} \dot{\hat{x}} u_i \tilde{x}^T \hat{W}_{gi}^T \hat{\sigma}_{gi}') \quad i = 1...m, \end{aligned} \quad (4\text{–}7)$$

where $proj(\cdot)$ is a smooth projection operator, and $\Gamma_{wf} \in \mathbb{R}^{L_f+1 \times L_f+1}$, $\Gamma_{vf} \in \mathbb{R}^{n \times n}$, $\Gamma_{wgi} \in \mathbb{R}^{L_{gi}+1 \times L_{gi}+1}$, $\Gamma_{vgi} \in \mathbb{R}^{n \times n}$ are constant positive diagonal adaptation gain matrices. The space of DNN weight estimates is projected onto a compact convex set, constructed using known upper bounds of the ideal weights (Assumption 4.5). This ensures that the weight estimates are always bounded, which is exploited in the subsequent stability analysis. Any of the several smooth projection algorithms may be used ([72, 73]). Adding and subtracting $\frac{1}{2} W_f^T \hat{\sigma}_f' \hat{V}_f^T \dot{\hat{x}} + \frac{1}{2} \hat{W}_f^T \hat{\sigma}_f' V_f^T \dot{\hat{x}} + \sum_{i=1}^{m} \left[\frac{1}{2} W_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{\hat{x}} u_i + \frac{1}{2} \hat{W}_{gi}^T \hat{\sigma}_{gi}' V_{gi}^T \dot{\hat{x}} u_i \right]$, and grouping similar terms, the expression in Eq. 4–6 can be rewritten as

$$\dot{r} = \tilde{N} + N_{B1} + \hat{N}_{B2} - kr - \gamma \tilde{x} - \beta_1 sgn(\tilde{x}), \quad (4\text{–}8)$$

where the auxiliary signals, $\tilde{N}(x, \tilde{x}, r, \hat{W}_f, \hat{V}_f, \hat{W}_{gi}, \hat{V}_{gi}, t)$, $N_{B1}(x, \hat{x}, \hat{W}_f, \hat{V}_f, \hat{W}_{gi}, \hat{V}_{gi}, t)$, and $\hat{N}_{B2}(\hat{x}, \dot{\hat{x}}, \hat{W}_f, \hat{V}_f, \hat{W}_{gi}, \hat{V}_{gi}, t) \in \mathbb{R}^n$ in Eq. 4–8 are defined as

$$\begin{aligned}\tilde{N} \triangleq\ & \alpha\dot{\tilde{x}} - \dot{\hat{W}}_f^T\hat{\sigma}_f - \hat{W}_f^T\hat{\sigma}_f'\dot{\hat{V}}_f^T\hat{x} + \frac{1}{2}W_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\tilde{x}} + \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'V_f^T\dot{\tilde{x}} \\ & - \sum_{i=1}^{m}\left[\dot{\hat{W}}_{gi}^T\hat{\sigma}_{gi}u_i + \hat{W}_{gi}^T\hat{\sigma}_{gi}'\dot{\hat{V}}_{gi}^T\hat{x}u_i - \frac{1}{2}\hat{W}_{gi}^T\hat{\sigma}_{gi}'V_{gi}^T\dot{\tilde{x}}u_i - \frac{1}{2}W_{gi}^T\hat{\sigma}_{gi}'\hat{V}_{gi}^T\dot{\tilde{x}}u_i\right],\end{aligned} \tag{4–9}$$

$$\begin{aligned}N_{B1} \triangleq\ & \sum_{i=1}^{m}\left[W_{gi}^T\sigma_{gi}\dot{u}_i + W_{gi}^T\sigma_{gi}'V_{gi}^T\dot{x}u_i + \dot{\varepsilon}_{gi}(x)u_i + \varepsilon_{gi}(x)\dot{u}_i\right] + W_f^T\sigma_f'V_f^T\dot{x} \\ & + \dot{\varepsilon}_f(x) + \dot{d} - \sum_{i=1}^{m}\left[\frac{1}{2}\hat{W}_{gi}^T\hat{\sigma}_{gi}'V_{gi}^T\dot{x}u_i + \frac{1}{2}W_{gi}^T\hat{\sigma}_{gi}'\hat{V}_{gi}^T\dot{x}u_i + \hat{W}_{gi}^T\hat{\sigma}_{gi}\dot{u}_i\right] \\ & - \frac{1}{2}W_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{x} - \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'V_f^T\dot{x},\end{aligned} \tag{4–10}$$

$$\hat{N}_{B2} \triangleq \sum_{i=1}^{m}\left[\frac{1}{2}\tilde{W}_{gi}^T\hat{\sigma}_{gi}'\hat{V}_{gi}^T\dot{\hat{x}}u_i + \frac{1}{2}\hat{W}_{gi}^T\hat{\sigma}_{gi}'\tilde{V}_{gi}^T\dot{\hat{x}}u_i\right] + \frac{1}{2}\tilde{W}_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\hat{x}} + \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'\tilde{V}_f^T\dot{\hat{x}}. \tag{4–11}$$

To facilitate the subsequent stability analysis, an auxiliary term $N_{B2}(\hat{x}, \dot{x}, \hat{W}_f, \hat{V}_f, \hat{W}_{gi}, \hat{V}_{gi}, t) \in \mathbb{R}^n$ is defined by replacing $\dot{\hat{x}}(t)$ in $\hat{N}_{B2}(\cdot)$ by $\dot{x}(t)$, and $\tilde{N}_{B2}(\hat{x}, \dot{\tilde{x}}, \hat{W}_f, \hat{V}_f, \hat{W}_{gi}, \hat{V}_{gi}, t) \triangleq \hat{N}_{B2}(\cdot) - N_{B2}(\cdot)$. The terms $N_{B1}(\cdot)$ and $N_{B2}(\cdot)$ are grouped as $N_B \triangleq N_{B1} + N_{B2}$. Using Assumptions 4.2, 4.5-4.7, Eq. 4–5 and Eq. 4–7, the following bound can be obtained for Eq. 4–9

$$\left\|\tilde{N}\right\| \leq \rho_1(\|z\|)\|z\|, \tag{4–12}$$

where $z \triangleq \left[\tilde{x}^T\ r^T\right]^T \in \mathbb{R}^{2n}$, and $\rho_1(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function. The following bounds can be developed based on Eq. 4–2, Assumptions 4.2-4.3, 4.5-4.7, Eq. 4–7, Eq. 4–10 and Eq. 4–11

$$\|N_{B1}\| \leq \zeta_1, \quad \|N_{B2}\| \leq \zeta_2, \quad \left\|\dot{N}_B\right\| \leq \zeta_3 + \zeta_4\rho_2(\|z\|)\|z\|, \tag{4–13}$$

$$\left\|\dot{\tilde{x}}^T\tilde{N}_{B2}\right\| \leq \zeta_5\|\tilde{x}\|^2 + \zeta_6\|r\|^2, \tag{4–14}$$

where $\zeta_i \in \mathbb{R}$, $i = 1, ..., 6$ are computable positive constants, and $\rho_2(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function.

To facilitate the subsequent stability analysis, let $\mathcal{D} \subset \mathbb{R}^{2n+2}$ be a domain containing $y(t) = 0$, where $y(t) \in \mathbb{R}^{2n+2}$ is defined as

$$y \triangleq \left[\tilde{x}^T \quad r^T \quad \sqrt{P} \quad \sqrt{Q}\right]^T, \tag{4–15}$$

where the auxiliary function $P(z,t) \in \mathbb{R}$ is the generalized solution (in Filippov's sense) to the differential equation

$$\dot{P} = -L, \qquad P(0) = \beta_1 \sum_{i=1}^{n} |\tilde{x}_i(0)| - \tilde{x}^T(0) N_B(0), \tag{4–16}$$

where the auxiliary function $L(z,t) \in \mathbb{R}$ is defined as

$$L \triangleq r^T(N_{B1} - \beta_1 sgn(\tilde{x})) + \dot{\tilde{x}}^T N_{B2} - \beta_2 \rho_2(\|z\|) \|z\| \|\tilde{x}\|, \tag{4–17}$$

where $\beta_1, \beta_2 \in \mathbb{R}$ are selected according to the following sufficient conditions[1] :

$$\beta_1 > \max(\zeta_1 + \zeta_2, \ \zeta_1 + \frac{\zeta_3}{\alpha}), \qquad \beta_2 > \zeta_4, \tag{4–18}$$

to ensure that $P(t) \geq 0$. The auxiliary function $Q(\tilde{W}_f, \tilde{V}_f, \tilde{W}_{gi}, \tilde{V}_{gi}) \in \mathbb{R}$ in Eq. 4–15 is defined as

$$Q \triangleq \frac{1}{4}\alpha \left[tr(\tilde{W}_f^T \Gamma_{wf}^{-1} \tilde{W}_f) + tr(\tilde{V}_f^T \Gamma_{vf}^{-1} \tilde{V}_f) + \sum_{i=1}^{m} (tr(\tilde{W}_{gi}^T \Gamma_{wgi}^{-1} \tilde{W}_{gi}) + tr(\tilde{V}_{gi}^T \Gamma_{vgi}^{-1} \tilde{V}_{gi})) \right], \tag{4–19}$$

where $tr(\cdot)$ denotes the trace of a matrix.

[1] The derivation of the sufficient conditions in Eq. 4–18 is provided in the Appendix.

Theorem 4.1. *The identifier developed in Eq. 4–3 along with its weight update laws in Eq. 4–7 ensures asymptotic convergence, in the sense that*

$$\lim_{t\to\infty} \|\tilde{x}(t)\| = 0 \qquad \textit{and} \qquad \lim_{t\to\infty} \left\|\dot{\tilde{x}}(t)\right\| = 0$$

provided the control gains k and γ are selected sufficiently large based on the initial conditions of the states[2] *, and satisfy the following sufficient conditions*

$$\gamma > \frac{\zeta_5}{\alpha}, \qquad k > \zeta_6, \tag{4–20}$$

where ζ_5 and ζ_6 are introduced in Eq. 4–14, and β_1 and β_2 are selected according to the sufficient conditions in Eq. 4–18.

Proof. Let $V : \mathcal{D} \to \mathbb{R}$ be a Lipschitz continuous regular positive definite function defined as

$$V \triangleq \frac{1}{2} r^T r + \frac{1}{2}\gamma \tilde{x}^T \tilde{x} + P + Q, \tag{4–21}$$

which satisfies the following inequalities:

$$U_1(y) \leq V(y) \leq U_2(y), \tag{4–22}$$

where $U_1(y), U_2(y) \in \mathbb{R}$ are continuous positive definite functions defined as $U_1 \triangleq \frac{1}{2} min(1, \gamma) \|y\|^2$ and $U_2 \triangleq max(1, \gamma) \|y\|^2$, respectively.

Let $\dot{y} = F(y, t)$ represent the closed-loop differential equations in Eqs. 4–4, 4–7, 4–8, and 4–16, where $F(\cdot) \in \mathbb{R}^{2n+2}$ denotes the right-hand side of the the closed-loop error signals. Since $F(y, t)$ is discontinuous in the set $\{(y, t)|\tilde{x} = 0\}$, the existence and stability of solutions cannot be studied in the classical sense. Using the differential inclusion $\dot{y} \in F(y, t)$, where y is absolutely continuous and $F(\cdot)$ is Lebesgue measurable and locally bounded, existence and uniqueness of solutions can be established in the

[2] See subsequent stability analysis.

Filippov's sense (see [74, 76] and Appendix A.2 for further details). Stability of solutions based on differential inclusion is studied using non-smooth Lyapunov functions, using the development in [79, 80]. The generalized time derivative of Eq. 4–21 exists almost everywhere (a.e.), and $\dot{V}(y) \in^{a.e.} \dot{\tilde{V}}(y)$ where

$$\dot{\tilde{V}} = \bigcap_{\xi \in \partial V(y)} \xi^T K \left[\dot{r}^T \;\; \dot{\tilde{x}}^T \;\; \frac{1}{2} P^{-\frac{1}{2}} \dot{P} \;\; \frac{1}{2} Q^{-\frac{1}{2}} \dot{Q} \;\; 1 \right]^T, \tag{4–23}$$

where ∂V is the generalized gradient of V [78], and $K[\cdot]$ is defined in 3–49. Since $V(y)$ is a Lipschitz continuous regular function, Eq. 4–23 can be simplified as [79]

$$\begin{aligned} \dot{\tilde{V}} &= \nabla V^T K \left[\dot{r}^T \;\; \dot{\tilde{x}}^T \;\; \frac{1}{2} P^{-\frac{1}{2}} \dot{P} \;\; \frac{1}{2} Q^{-\frac{1}{2}} \dot{Q} \;\; 1 \right]^T \\ &= \left[r^T \;\; \gamma \tilde{x}^T \;\; 2P^{\frac{1}{2}} \;\; 2Q^{\frac{1}{2}} \;\; 0 \right] K \left[\dot{r}^T \;\; \dot{\tilde{x}}^T \;\; \frac{1}{2} P^{-\frac{1}{2}} \dot{P} \;\; \frac{1}{2} Q^{-\frac{1}{2}} \dot{Q} \;\; 1 \right]^T. \end{aligned}$$

Using the calculus for $K[\cdot]$ from [80] (Theorem 1, Properties 2,5,7), and substituting the dynamics from Eq. 4–8 and Eq. 4–16, yields

$$\begin{aligned} \dot{\tilde{V}} \subset\; & r^T(\tilde{N} + N_{B1} + \hat{N}_{B2} - kr - \beta_1 K[sgn(\tilde{x})] - \gamma\tilde{x}) + \gamma\tilde{x}^T(r - \alpha\tilde{x}) \\ & - r^T(N_{B1} - \beta_1 K[sgn(\tilde{x})]) - \dot{\tilde{x}}^T N_{B2} + \beta_2 \rho_2(\|z\|)\, \|z\|\, \|\tilde{x}\| \\ & - \frac{1}{2}\alpha \left[tr(\tilde{W}_f^T \Gamma_{wf}^{-1} \dot{\hat{W}}_f) + tr(\tilde{V}_f^T \Gamma_{vf}^{-1} \dot{\hat{V}}_f) \right] \\ & - \frac{1}{2}\alpha \sum_{i=1}^{m} \left[tr(\tilde{W}_{gi}^T \Gamma_{wgi}^{-1} \dot{\hat{W}}_{gi}) + tr(\tilde{V}_{gi}^T \Gamma_{vgi}^{-1} \dot{\hat{V}}_{gi}) \right]. \\ = \; & -\alpha\gamma\tilde{x}^T\tilde{x} - kr^T r + r^T\tilde{N} + \frac{1}{2}\alpha\tilde{x}^T\tilde{W}_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\tilde{x}} + \frac{1}{2}\alpha\tilde{x}^T\hat{W}_f^T\hat{\sigma}_f'\tilde{V}_f^T\dot{\tilde{x}} \\ & + \sum_{i=1}^{m} \frac{1}{2}\alpha\tilde{x}^T\tilde{W}_{gi}^T\hat{\sigma}_{gi}'\hat{V}_{gi}^T\dot{\tilde{x}}u_i + \frac{1}{2}\alpha\tilde{x}^T\hat{W}_{gi}^T\hat{\sigma}_{gi}'\tilde{V}_{gi}^T\dot{\tilde{x}}u_i + \dot{\tilde{x}}^T(\hat{N}_{B2} - N_{B2}) \\ & + \beta_2\rho_2(\|z\|)\, \|z\|\, \|\tilde{x}\| - \frac{1}{2}\alpha tr(\tilde{W}_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\tilde{x}}\tilde{x}^T) - \frac{1}{2}\alpha tr(\tilde{V}_f^T\dot{\tilde{x}}\tilde{x}^T\hat{W}_f^T\hat{\sigma}_f') \\ & - \frac{1}{2}\alpha \sum_{i=1}^{m} \left[tr(\tilde{W}_{gi}^T\hat{\sigma}_{gi}'\hat{V}_{gi}^T\dot{\tilde{x}}u_i\tilde{x}^T) + tr(\tilde{V}_{gi}^T\dot{\tilde{x}}u_i\tilde{x}^T\hat{W}_{gi}^T\hat{\sigma}_{gi}') \right], \end{aligned} \tag{4–24}$$

where Eq. 4–7, $K[sgn(\tilde{x})] = SGN(\tilde{x})$ [80], and the fact that $(r^T - r^T)_i SGN(\tilde{x}_i) = 0$, is used (the subscript i denotes the i^{th} element), such that $SGN(\tilde{x}_i) = 1$ if $\tilde{x}_i > 0$, $[-1, 1]$

if $\tilde{x}_i = 0$, and -1 if $\tilde{x}_i < 0$. Canceling common terms, substituting for $k \triangleq k_1 + k_2$ and $\gamma \triangleq \gamma_1 + \gamma_2$, using Eqs. 4–12, 4–14, and completing the squares, the expression in Eq. 4–24 can be upper bounded as

$$\dot{\tilde{V}} \leq -(\alpha\gamma_1 - \zeta_5)\|\tilde{x}\|^2 - (k_1 - \zeta_6)\|r\|^2 + \frac{\rho_1(\|z\|)^2}{4k_2}\|z\|^2 + \frac{\beta_2^2\rho_2(\|z\|)^2}{4\alpha\gamma_2}\|z\|^2. \quad (4\text{–}25)$$

Provided the sufficient conditions in Eq. 4–20 are satisfied, the expression in Eq. 4–25 can be rewritten as

$$\begin{aligned} \dot{\tilde{V}} &\leq -\lambda\|z\|^2 + \frac{\rho(\|z\|)^2}{4\eta}\|z\|^2 \\ &\leq -U(y) \quad \forall y \in \mathcal{D} \end{aligned} \quad (4\text{–}26)$$

where $\lambda \triangleq min\{\alpha\gamma_1 - \zeta_5,\ k_1 - \zeta_6\}$, $\eta \triangleq min\{k_2,\ \frac{\alpha\gamma_2}{\beta_2^2}\}$, $\rho(\|z\|)^2 \triangleq \rho_1(\|z\|)^2 + \rho_2(\|z\|)^2$ is a positive, globally invertible, non-decreasing function, and $U(y) = c\|z\|^2$, for some positive constant c, is a continuous, positive semi-definite function defined on the domain $\mathcal{D} \triangleq \left\{y(t) \in \mathbb{R}^{2n+2} \mid \|y\| \leq \rho^{-1}\left(2\sqrt{\lambda\eta}\right)\right\}$. The size of the domain $\mathcal{D}$ can be increased by increasing the gains k and γ. The result in Eq. 4–26 indicates that $\dot{V}(y) \leq -U(y)$ $\forall\ \dot{V}(y) \in^{a.e.} \dot{\tilde{V}}(y)\ \forall y \in \mathcal{D}$. The inequalities in Eq. 4–22 and Eq. 4–26 can be used to show that $V(y) \in \mathcal{L}_\infty$ in $\mathcal{D}$; hence, $\tilde{x}(t), r(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. Using Eq. 4–5, standard linear analysis can be used to show that $\dot{\tilde{x}}(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. Since $\dot{x}(t) \in \mathcal{L}_\infty$ from Eq. 4–1 and Assumption 4.2-4.3, $\dot{\hat{x}}(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. From the use of projection in Eq. 4–7, $\hat{W}_f(t), \hat{W}_{gi}(t) \in \mathcal{L}_\infty$, $i = 1...m$. Using the above bounding arguments, it can be shown from Eq. 4–8 that $\dot{r}(t) \in \mathcal{L}_\infty$ in $\mathcal{D}$. Since $\tilde{x}(t)$,$r(t) \in \mathcal{L}_\infty$, the definition of $U(y)$ can be used to show that it is uniformly continuous in $\mathcal{D}$. Let $\mathcal{S} \subset \mathcal{D}$ denote a set defined as $\mathcal{S} \triangleq \left\{y(t) \subset \mathcal{D} \mid U_2(y(t)) < \frac{1}{2}\left(\rho^{-1}\left(2\sqrt{\lambda\eta}\right)\right)^2\right\}$, where the region of attraction can be made arbitrarily large to include any initial conditions by increasing the control gain η (i.e. a semi-global type of stability result), and hence $c\|z\|^2 \to 0$ as $t \to \infty$ $\forall y(0) \in \mathcal{S}$. Using the definition of $z(t)$, it can be shown that $\|\tilde{x}(t)\|, \left\|\dot{\tilde{x}}(t)\right\|, \|r\| \to 0$ as $t \to \infty$ $\forall y(0) \in \mathcal{S}$. □

4.2 Comparison with Related Work

The most common approach to estimate derivatives is by using numerical differentiation methods. The Euler backward difference approach is one of the simplest and the most common numerical methods to differentiate a signal; however, this ad hoc approach yields erroneous results in the presence of sensor noise. The central difference algorithm performs better than backward difference; however, the central difference algorithm is non-causal since it requires future state values to estimate the current derivative. Noise attenuation in numerical differentiators may be achieved by using a low-pass filter, at the cost of introducing a phase delay in the system. A more analytically rigorous approach is to cast the problem of state derivative estimation as an observer design problem by augmenting the state with its derivative, where the state is fully measurable and the state derivative is not, thereby, reducing the problem to designing an observer for the unmeasurable state derivative. Previous approaches to solve the problem use pure robust feedback methods requiring infinite gain or infinite frequency [95–97]. A high gain observer is presented in [96] to estimate the output derivatives, and asymptotic convergence to the derivative is achieved as the gain tends to infinity, which is problematic in general and especially in the presence of noise. In [97], a robust exact differentiator using a 2-sliding mode algorithm is developed which assumes a known upper bound for a Lipschitz constant of the derivative.

All the above mentioned methods are robust non model-based approaches. In contrast to purely robust feedback methods, an identification-based robust adaptive approach is considered in this work. The proposed identifier consists of a dynamic neural network (DNN) [87, 88, 90, 98] and a RISE (Robust Integral of the Sign of the Error) term [47, 71], where the DNN adaptively identifies the unknown system dynamics online, while RISE, a continuous robust feedback term, is used to guarantee asymptotic convergence to the state derivative in the presence of uncertainties and exogenous disturbances. The DNN with its recurrent feedback connections has been shown to learn dynamics of high dimensional uncertain nonlinear systems with arbitrary accuracy [98, 99], motivating their use in the

proposed identifier. Unlike most previous results on DNN-based system identification [88–91, 94], which only guarantee bounded stability of the identification error system in the presence of DNN approximation errors and exogenous disturbances, the addition of RISE to the DNN identifier guarantees asymptotic identification.

The RISE structure combines the features of the high gain observer and higher order sliding mode methods, in the sense that it consists of high gain proportional and integral state feedback terms (similar to a high gain observer), and the integral of a signum term, allowing it to implicitly learn and cancel the effects of DNN approximation errors and exogenous disturbances in the Lyapunov stability analysis, guaranteeing asymptotic convergence.

4.3 Experiment and Simulation Results

Experiments and simulations on a two-link robot manipulator (Fig. 3-2) are performed to compare the proposed method with several other derivative estimation methods. The following robot dynamics are considered:

$$M(q)\ddot{q} + V_m(q,\dot{q})\dot{q} + F_d\dot{q} + F_s(\dot{q}) = u(t), \tag{4–27}$$

where $q(t) = [q_1 \;\; q_2]^T$ and $\dot{q}(t) = [\dot{q}_1 \;\; \dot{q}_2]^T$ are the angular positions (rad) and angular velocities (rad/sec) of the two links, respectively, $M(q)$ is the inertia matrix, and $V_m(q,\dot{q})$ is the centripetal-Coriolis matrix, defined as

$$M \triangleq \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \quad V_m \triangleq \begin{bmatrix} -p_3s_2\dot{q}_2 & -p_3s_2(\dot{q}_1 + \dot{q}_2) \\ p_3s_2\dot{q}_1 & 0 \end{bmatrix},$$

where $p_1 = 3.473\ kg \cdot m^2$, $p_2 = 0.196\ kg \cdot m^2$, $p_3 = 0.242\ kg \cdot m^2$, $c_2 = cos(q_2)$, $s_2 = sin(q_2)$, $F_d = diag\{5.3,\ 1.1\}\ Nm \cdot sec$ and $F_s(\dot{q}) = diag\{8.45tanh(\dot{q}_1),\ 2.35tanh(\dot{q}_2)\}\ Nm$ are the models for dynamic and static friction, respectively. The robot model in Eq. 4–27 can be expressed as $\dot{x} = f(x) + g(x)u + d$, where the state $x(t) \in \mathbb{R}^4$ is defined as $x(t) \triangleq [q_1 \;\; q_2 \;\; \dot{q}_1 \;\; \dot{q}_2]^T$, $d(t) \triangleq 0.1sin(10t)[1\ 1\ 1\ 1]^T$ is an exogenous disturbance, and

$f(x) \in \mathbb{R}^4$ and $g(x) \in \mathbb{R}^{4\times 2}$ are defined as $f(x) \triangleq \begin{bmatrix} \dot{q}^T & \{M^{-1}(-V_m - F_d)\dot{q} - F_s\}^T \end{bmatrix}^T$ and $g(x) = [0_{2\times 2} \quad M^{-1}]$, respectively. The control input is designed as a PD controller to track the desired trajectory $q_d(t) = [0.5sin(2t) \quad 0.5cos(2t)]^T$, as $u(t) = -2[q_1(t) - 0.5sin(2t) \quad q_2(t) - 0.5cos(2t)]^T - [\dot{q}_1(t) - cos(2t) \quad \dot{q}_2(t) + sin(2t)]^T$. The objective is to design a state derivative estimator $\dot{\hat{x}}(t)$ to asymptotically converge to $\dot{x}(t)$. The performance of the developed RISE-based DNN identifier in Eqs. 4–3 and 4–7 is compared with the 2-sliding mode robust exact differentiator [97]

$$\dot{\hat{x}} = z_s + \lambda_s \sqrt{|\tilde{x}|} sgn(\tilde{x}), \qquad \dot{z_s} = \alpha_s sgn(\tilde{x}), \tag{4–28}$$

and the high gain observer [96]

$$\dot{\hat{x}} = z_h + \frac{\alpha_{h1}}{\varepsilon_{h1}}(\tilde{x}), \qquad \dot{z_h} = \frac{\alpha_{h2}}{\varepsilon_{h2}}(\tilde{x}). \tag{4–29}$$

The motor encoders in Fig. 3-2 provide position measurements for the two links ($x_1(t)$ and $x_2(t)$) with a resolution of 614400 pulses/revolution, and a standard backwards difference algorithm is used to numerically determine angular velocities ($x_3(t)$ and $x_4(t)$) from the encoder readings. The experimental results for the state derivative estimates with the 2-sliding mode, the high gain observer, the proposed method, and the backward difference algorithm, are shown in Fig. 4-1. Because of unavailability of velocity and acceleration sensors to verify the state derivative estimates, no performance comparisons could be made. However, a few observations can be made from Fig. 4-1 – the steady-state estimates of the state derivative for the 2-sliding mode, the high gain observer, and the proposed method look similar; however, the transient response of the 2-sliding mode differs from that of the high gain observer and the proposed method. On the other hand, the state derivative estimate with backward difference is very noisy and does not resemble the response of any of the other methods. The experimental results demonstrate that the performance of the proposed identifier-based state derivative estimator is comparable to existing methods in literature, and that the estimates from backward difference are

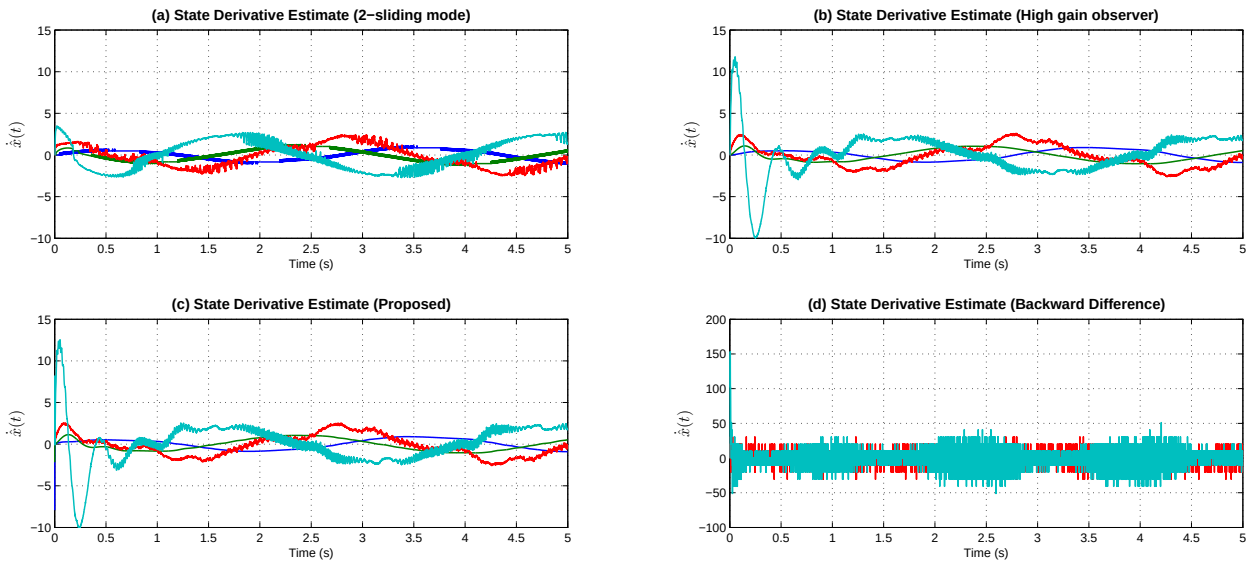

Figure 4-1. Comparison of the state derivative estimate $\dot{\hat{x}}(t)$ for (a) 2-sliding mode, (b) high gain observer, (c) proposed method, and (d) backward difference on a two-link experiment testbed.

prone to error in presence of sensor noise. Simulations are performed to compare, qualitatively and quantitatively, the performance of the different estimators. The gains for the identifier in Eqs. 4–3 and 4–7 are selected as $k = 20$, $\alpha = 5$, $\gamma = 200$, $\beta_1 = 1.25$, and the DNN adaptation gains are selected as $\Gamma_{wf} = 0.1\mathbb{I}_{11\times 11}$, $\Gamma_{vf} = \mathbb{I}_{4\times 4}$, $\Gamma_{wg1} = 0.7\mathbb{I}_{4\times 4}$, $\Gamma_{wg2} = 0.4\mathbb{I}_{4\times 4}$, $\Gamma_{vg1} = \Gamma_{vg2} = \mathbb{I}_{4\times 4}$, where $\mathbb{I}_{n\times n}$ denotes an identity matrix of appropriate dimensions. The neural networks for $f(x)$ and $g(x)$ are designed to have 10 and 3 hidden layer neurons, respectively, and the DNN weights are initialized as uniformly distributed random numbers in the interval $[-1, 1]$. The gains for the 2-sliding mode differentiator in Eq. 4–28 are selected as $\lambda_s = 4.1$, $\alpha_s = 4$, while the gains for the high gain observer in Eq. 4–29 are selected as $\alpha_{h1} = 0.2$, $\varepsilon_{h1} = 0.01$, $\alpha_{h2} = 0.3$, $\varepsilon_{h2} = 0.001$. To ensure a fair comparison, the gains of all the three estimators were tuned for best performance (least RMS error) for the same settling time of approximately 0.4 seconds for the state derivative estimation errors. A white Gaussian noise was added to the state measurements, maintaining a signal to noise ratio of 60 dB. The initial conditions of the system and the estimators are chosen as $x(t) = \hat{x}(t) = [1\ 1\ 1\ 1]^T$.

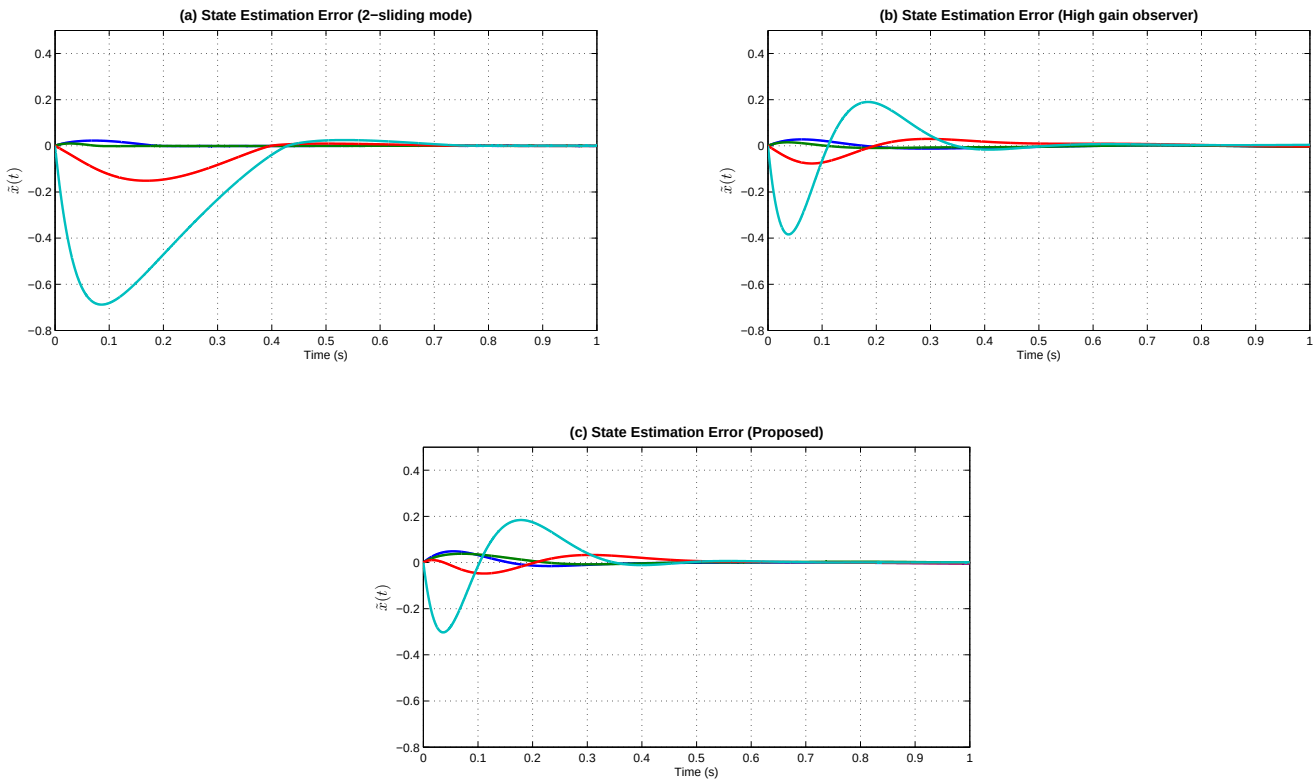

Figure 4-2. Comparison of the state estimation errors $\tilde{x}(t)$ for (a) 2-sliding mode, (b) high gain observer, and (c) proposed methods, in the presence of sensor noise (SNR 60 dB).

Table 4-1. Comparison of transient $(t = 0 - 5$ sec.) and steady-state $(t = 5 - 10$ sec.) state derivative estimation errors $\dot{\tilde{x}}(t)$ for different derivative estimation methods in presence of noise (60 dB).

	Backward difference	Central difference	2-sliding mode	High gain observer	Proposed
Transient RMS error	14.4443	7.6307	2.3480	2.1326	1.7808
Steady state RMS error	14.1461	7.0583	0.1095	0.0414	0.0297

Figs. 4-2-4-4 show the simulation results for state estimation and state derivative estimation errors for the 2-sliding mode robust exact differentiator in [97], the high gain observer in [96], and the developed RISE-based DNN estimator. While the maximum overshoot in estimating the state derivative (Fig. 4-3) using 2-sliding mode is smaller,

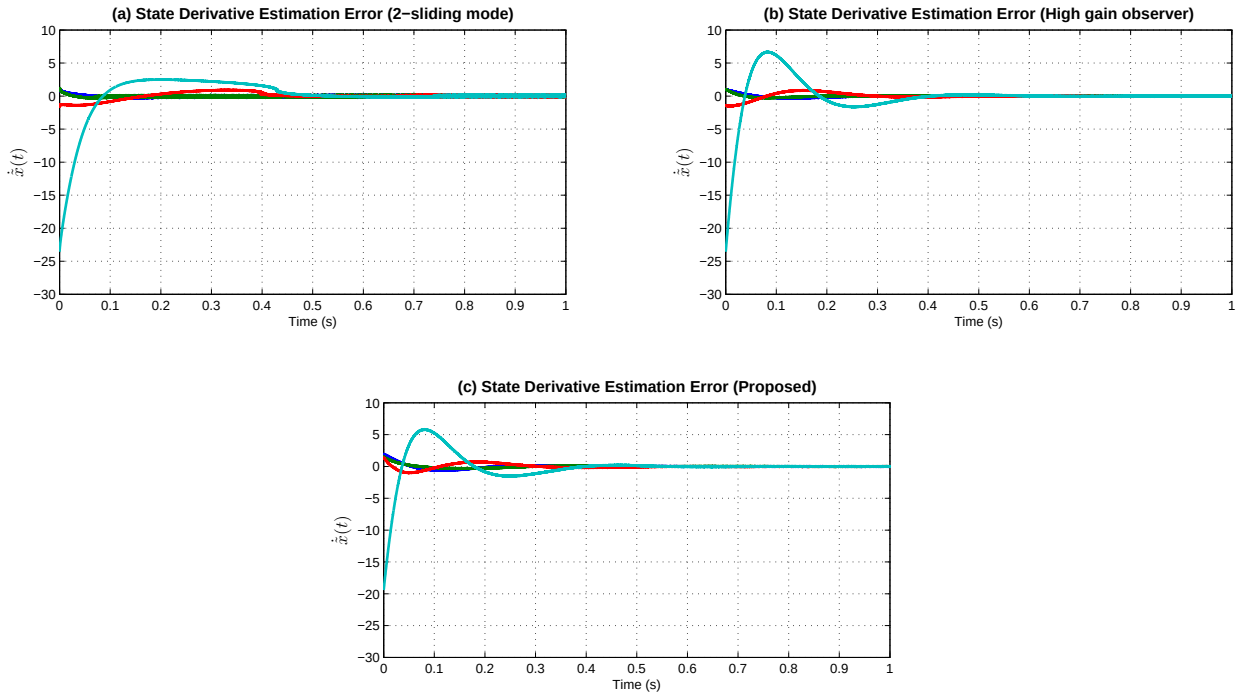

Figure 4-3. Comparison of the state derivative estimation errors $\dot{\tilde{x}}(t)$ for (a) 2-sliding mode, (b) high gain observer, and (c) proposed methods, in the presence of sensor noise (SNR 60 dB).

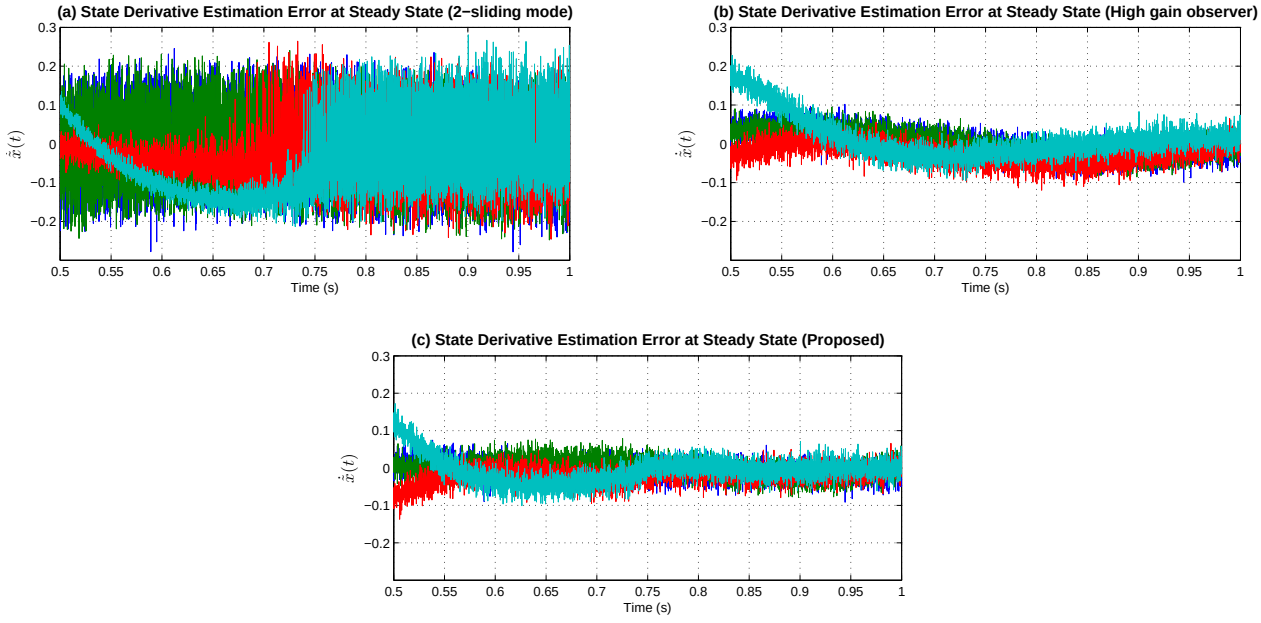

Figure 4-4. Comparison of the state derivative estimation errors $\dot{\tilde{x}}(t)$ at steady state, for (a) 2-sliding mode, (b) high gain observer, and (c) proposed methods, in the presence of sensor noise (SNR 60 dB).

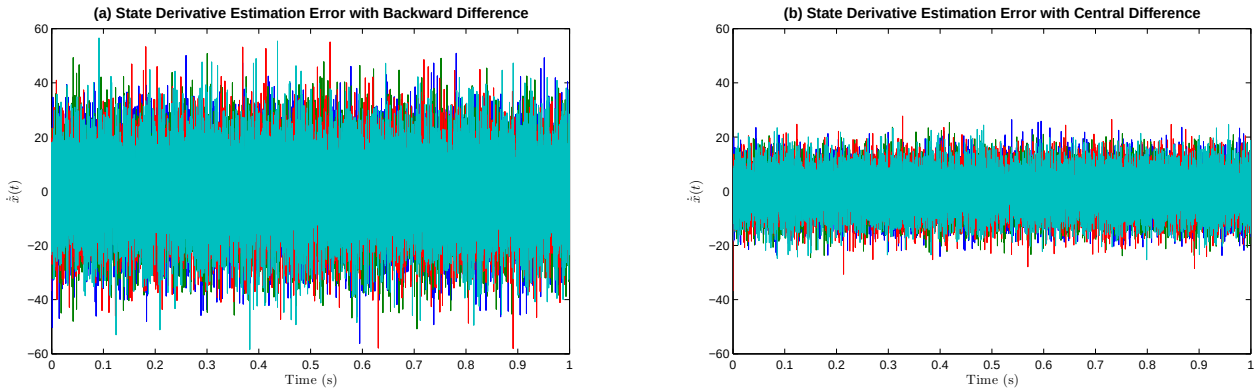

Figure 4-5. State derivative estimation errors $\dot{\tilde{x}}(t)$ for numerical differentiation methods (a) backward difference and (b) central difference with step-size of 10^{-4}, in the presence of sensor noise (SNR 60 dB).

the steady state errors are comparatively larger than both the high gain observer and the proposed method. Table 4-1 gives a comparison of the transient and steady state RMS state derivative estimation errors for different estimation methods. Results of standard numerical differentiation algorithms - backward difference and central difference (with a step-size of 10^{-4}) are also included; as seen from Table 4-1 and Fig. 4-5, they perform significantly worse than the other methods, in presence of noise. Although, simulation results for the high gain observer and the developed method are comparable, as seen from Figs. 4-2-4-4 and Table 4-1, differences exist in the structure of the estimators and proof of convergence of the estimates. The developed identifier includes the RISE structure, which combines the features of the high gain observer with the integral of a signum term, allowing it to implicitly learn and cancel terms in the stability analysis; thus, guaranteeing asymptotic convergence. While singular perturbation methods can be used to prove asymptotic convergence of the high gain observer to the derivative of the output signal ($\dot{x}(t)$ in this case) as the gains tend to infinity [100], Lyapunov-based stability methods are used to prove asymptotic convergence of the proposed identifier (as $t \to \infty$) with finite gains. Further, while both high gain observer and 2-sliding mode robust exact differentiator are purely robust feedback methods, the developed method,

in addition to using a robust RISE feedback term, uses a DNN to adaptively identify the system dynamics.

4.4 Summary

A robust identifier is developed for online estimation of the state derivative of uncertain nonlinear systems in the presence of exogenous disturbances. The result differs from existing pure robust methods in that the proposed method combines a DNN system identifier with a robust RISE feedback to ensure asymptotic convergence to the state derivative, which is proven using a Lyapunov-based stability analysis. Simulation results in the presence of noise show an improved transient and steady state performance of the developed identifier in comparison to several other derivative estimation methods.

CHAPTER 5
AN ACTOR-CRITIC-IDENTIFIER ARCHITECTURE FOR APPROXIMATE OPTIMAL CONTROL OF UNCERTAIN NONLINEAR SYSTEMS

RL uses evaluative feedback from the environment to take appropriate actions [101]. One of the most widely used architectures to implement RL algorithms is the AC architecture, where an actor performs certain actions by interacting with its environment, the critic evaluates the actions and gives feedback to the actor, leading to improvement in performance of subsequent actions [4, 19, 101]. AC algorithms are pervasive in machine learning and are used to learn the optimal policy online for finite-space discrete-time Markov decision problems [6, 17, 19, 101, 102]. The objective of this chapter is to append an identifier structure to the standard AC architecture, called actor-critic-identifier, which solves the continuous-time optimal control problem for nonlinear systems without requiring complete knowledge of system dynamics.

5.1 Actor-Critic-Identifier Architecture for HJB Approximation

Consider a continuous-time nonlinear system

$$\dot{x} = F(x,u),$$

where $x(t) \in \mathcal{X} \subseteq \mathbb{R}^n$, $u(t) \in \mathcal{U} \subseteq \mathbb{R}^m$ is the control input, $F : \mathcal{X} \times \mathcal{U} \to \mathbb{R}^n$ is Lipschitz continuous on $\mathcal{X} \times \mathcal{U}$ containing the origin, such that the solution $x(t)$ of the system is unique for any finite initial condition x_0 and control $u \in \mathcal{U}$. The optimal value function can be defined as

$$V^*(x(t)) = \min_{\substack{u(\tau)\in\Psi(\mathcal{X}) \\ t\leq\tau<\infty}} \int_t^\infty r(x(s),u(x(s)))\,ds, \tag{5–1}$$

where $\Psi(\mathcal{X})$ is a set of admissible policies, and $r(x,u) \in \mathbb{R}$ is the immediate or local cost, defined as

$$r(x,u) = Q(x) + u^T R u, \tag{5–2}$$

where $Q(x) \in \mathbb{R}$ is continuously differentiable and positive definite, and $R \in \mathbb{R}^{m\times m}$ is a positive-definite symmetric matrix. For the local cost in Eq. 5–2, which is convex in the control, and control-affine dynamics of the form

$$\dot{x} = f(x) + g(x)u, \tag{5–3}$$

where $f(x) \in \mathbb{R}^n$ and $g(x) \in \mathbb{R}^{n\times m}$, the closed-form expression for optimal control is derived as [52]

$$u^*(x) = -\frac{1}{2}R^{-1}g^T(x)\frac{\partial V^*(x)}{\partial x}^T, \tag{5–4}$$

where it is assumed that the value function $V^*(x)$ is continuously differentiable and satisfies $V^*(0) = 0$.

The Hamiltonian of the system in Eq. 5–3 is given by

$$H(x, u, V_x^*) \triangleq V_x^* F_u + r_u,$$

where $V_x^* \triangleq \frac{\partial V^*}{\partial x} \in \mathbb{R}^{1\times n}$ denotes the gradient of the optimal value function $V^*(x)$, $F_u(x, u) \triangleq f(x) + g(x)u \in \mathbb{R}^n$ denotes the system dynamics with control $u(x)$, and $r_u \triangleq r(x, u)$ denotes the local cost with control $u(x)$. The optimal value function $V^*(x)$ in Eq. 5–1 and the associated optimal policy $u^*(x)$ in Eq. 5–4 satisfy the HJB equation

$$H^*(x, u^*, V_x^*) = V_x^* F_{u^*} + r_{u^*} = 0. \tag{5–5}$$

Replacing $u^*(x)$, $V_x^*(x)$, and $F_{u^*}(x, u^*)$ in Eq. 5–5 by their approximations, $\hat{u}(x)$ (actor), $\hat{V}(x)$(critic), and $\hat{F}_{\hat{u}}(x, \hat{x}, \hat{u})$ (identifier), respectively, the approximate HJB equation is given by

$$\hat{H}^*(x, \hat{x}, \hat{u}, \hat{V}_x) = \hat{V}_x \hat{F}_{\hat{u}} + r_{\hat{u}}, \tag{5–6}$$

where $\hat{x}(t)$ denotes the state of the identifier. Using Eqs. 5–5 and 5–6, the error between the actual and the approximate HJB equation is given by the Bellman residual error

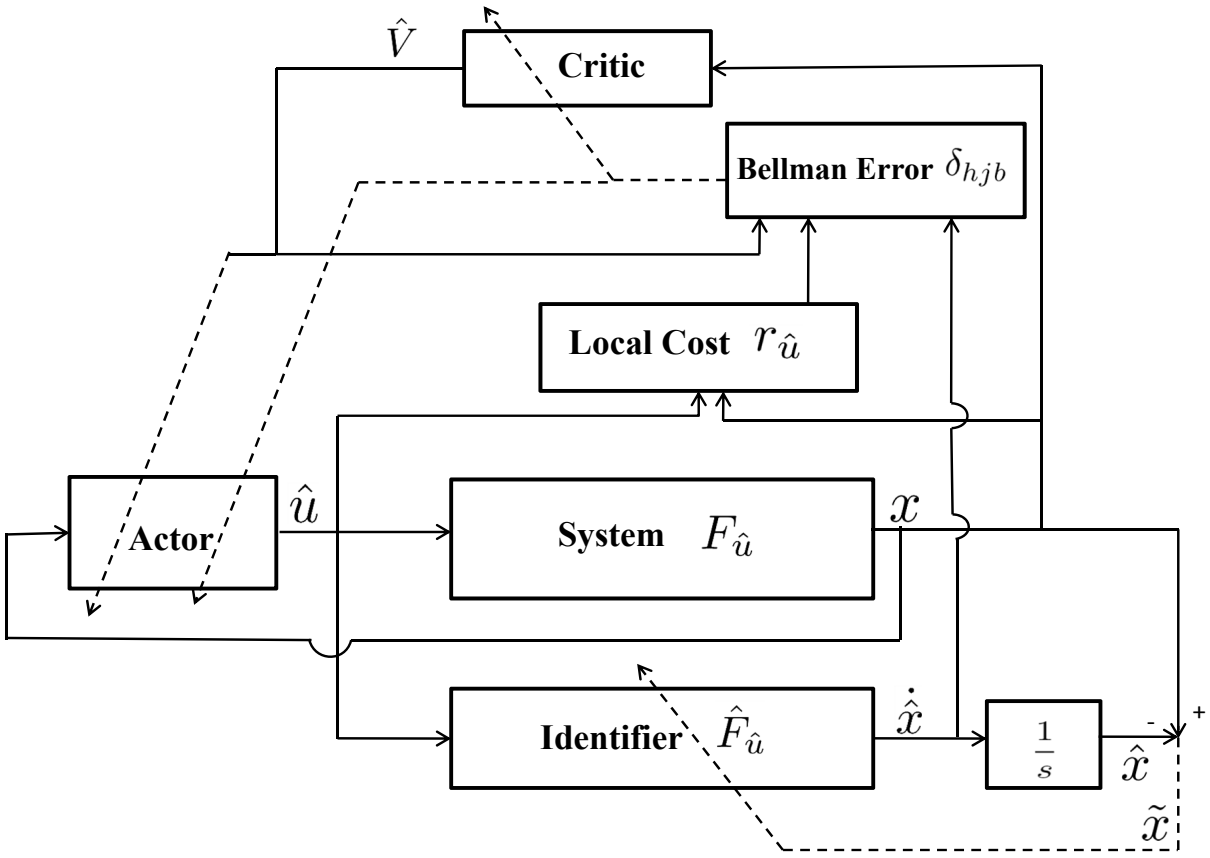

Figure 5-1. Actor-critic-identifier architecture to approximate the HJB.

$\delta_{hjb}(x, \hat{x}, \hat{u}, \hat{V}_x)$, defined as

$$\delta_{hjb} \triangleq \hat{H}^*(x, \hat{x}, \hat{u}, \hat{V}_x) - H^*(x, u^*, V_x^*). \tag{5–7}$$

Since $H^*(x, u^*, V_x^*) \equiv 0$, the Bellman error can be written in a measurable form as

$$\delta_{hjb} = \hat{H}^*(x, \hat{x}, \hat{u}, \hat{V}_x) = \hat{V}_x \hat{F}_{\hat{u}} + r(x, \hat{u}). \tag{5–8}$$

The actor and critic learn based on the Bellman error $\delta_{hjb}(\cdot)$, whereas the identifier estimates the system dynamics online using the identification error $\tilde{x}(t) \triangleq x(t) - \hat{x}(t)$, and hence is decoupled from the design of actor and critic. The block diagram of the ACI architecture is shown in Fig. 5-1.

The following assumptions are made about the control-affine system in Eq. 5–3.

Assumption 5.1. *The functions $f(x)$ and $g(x)$ are second-order differentiable.*

Assumption 5.2. *The input gain matrix $g(x)$ is known and bounded i.e. $0 < \|g(x)\| \leq \bar{g}$, where $\bar{g}$ is a known positive constant.*

Assuming the optimal control, the optimal value function and the system dynamics are continuous and defined on compact sets, NNs can be used to approximate them [15, 103]. Some standard NN assumptions which will be used throughout the work are:

Assumption 5.3. *Given a continuous function $\Upsilon : \mathbb{S} \to \mathbb{R}^n$, where $\mathbb{S}$ is a compact simply connected set, there exists ideal weights W, V such that the function can be represented by a NN as*

$$\Upsilon(x) = W^T \sigma(V^T x) + \varepsilon(x),$$

where $\sigma(\cdot)$ is the nonlinear activation function, and $\varepsilon(x)$ is the function reconstruction error.

Assumption 5.4. *The ideal NN weights are bounded by known positive constants i.e. $\|W\| \leq \bar{W}$, $\|V\| \leq \bar{V}$ [18].*

Assumption 5.5. *The NN activation function $\sigma(\cdot)$ and its derivative with respect to its arguments, $\sigma'(\cdot)$, are bounded.*

Assumption 5.6. *Using the NN universal approximation property [15, 103], the function reconstruction errors and its derivative with respect to its arguments are bounded [18] as $\|\varepsilon(\cdot)\| \leq \bar{\varepsilon}$, $\|\varepsilon'(\cdot)\| \leq \bar{\varepsilon}'$.*

5.2 Actor-Critic Design

Using Assumption 5.3 and Eq. 5–4, the optimal value function and the optimal control can be represented by NNs as

$$\begin{aligned} V^*(x) &= W^T \phi(x) + \varepsilon_v(x), \\ u^*(x) &= -\frac{1}{2} R^{-1} g^T(x) (\phi'(x)^T W + \varepsilon_v'(x)^T), \end{aligned} \tag{5–9}$$

where $W \in \mathbb{R}^N$ are unknown ideal NN weights, N is the number of neurons, $\phi(x) = [\phi_1(x) \ \phi_2(x) \ldots \phi_N(x)]^T \in \mathbb{R}^N$ is a smooth NN activation function such that $\phi_i(0) = 0$ and $\phi_i'(0) = 0 \ \ \forall i = 1...N$, and $\varepsilon_v(\cdot) \in \mathbb{R}$ is the function reconstruction error.

Assumption 5.7. *The NN activation functions $\{\phi_i(x) : i = 1...N\}$ are selected so that as $N \to \infty$, $\phi(x)$ provides a complete independent basis for $V^*(x)$.*

Using Assumption 5.7 and the Weierstrass higher-order approximation Theorem, both $V^*(x)$ and $\frac{\partial V^*(x)}{\partial x}$ can be uniformly approximated by NNs in Eq. 5–9, i.e. as $N \to \infty$, the approximation errors $\varepsilon_v(x)$, $\varepsilon_v'(x) \to 0$ [41]. The critic $\hat{V}(x)$ and the actor $\hat{u}(x)$ approximate the optimal value function and the optimal control in Eq. 5–9, and are given by

$$\hat{V}(x) = \hat{W}_c^T \phi(x); \qquad \hat{u}(x) = -\frac{1}{2} R^{-1} g^T(x) \phi'^T(x) \hat{W}_a, \tag{5–10}$$

where $\hat{W}_c(t) \in \mathbb{R}^N$ and $\hat{W}_a(t) \in \mathbb{R}^N$ are estimates of the ideal weights of the critic and actor NNs, respectively. The weight estimation errors for the critic and actor NNs are defined as $\tilde{W}_c(t) \triangleq W - \hat{W}_c(t) \in \mathbb{R}^N$ and $\tilde{W}_a(t) \triangleq W - \hat{W}_a(t) \in \mathbb{R}^N$, respectively.

Remark 5.1. *Since the optimal control is determined using the gradient of the optimal value function in Eq. 5–9, the critic NN in Eq. 5–10 may be used to determine the actor without using another NN for the actor. However, for ease in deriving weight update laws and subsequent stability analysis, separate NNs are used for the actor and the critic [14].*

The actor and critic NN weights are both updated based on the minimization of the Bellman error $\delta_{hjb}(\cdot)$ in Eq. 5–8, which can be rewritten by substituting $\hat{V}(x)$ from Eq. 5–10 as

$$\delta_{hjb} = \hat{W}_c^T \omega + r(x, \hat{u}), \tag{5–11}$$

where $\omega(x, \hat{x}, \hat{u}) \triangleq \phi'(x) \hat{F}_{\hat{u}}(x, \hat{x}, \hat{u}) \in \mathbb{R}^N$ is the critic NN regressor vector.

5.2.1 Least Squares Update for the Critic

Let $E_c(\delta_{hjb}) \in \mathbb{R}^+$ denote the integral squared Bellman error as

$$E_c = \int_0^t \delta_{hjb}^2(\tau)\, d\tau. \tag{5–12}$$

The least squares (LS) update law for the critic is generated by minimizing Eq. 5–12 as

$$\frac{\partial E_c}{\partial \hat{W}_c} = 2\int_0^t \delta_{hjb}(\tau)\frac{\partial \delta_{hjb}(\tau)}{\partial \hat{W}_c(t)}\, d\tau = 0. \tag{5–13}$$

Using $\frac{\partial \delta_{hjb}}{\partial \hat{W}_c} = \omega^T$ from Eq. 5–11, the batch LS critic weight estimate is determined from Eq. 5–13 as [104]

$$\hat{W}_c(t) = -\left(\int_0^t \omega(\tau)\omega(\tau)^T\, d\tau\right)^{-1} \int_0^t \omega(\tau) r(\tau)\, d\tau, \tag{5–14}$$

provided the inverse $\left(\int_0^t \omega(\tau)\omega(\tau)^T\, d\tau\right)^{-1}$ exists. For online implementation, a normalized recursive formulation of the LS algorithm is developed by taking the time derivative Eq. 5–14 and normalizing as [104]

$$\dot{\hat{W}}_c = -\eta_c \Gamma \frac{\omega}{1+\nu\omega^T\Gamma\omega}\delta_{hjb}, \tag{5–15}$$

where $\nu, \eta_c \in \mathbb{R}$ are constant positive gains, and $\Gamma(t) \triangleq \left(\int_0^t \omega(\tau)\omega(\tau)^T\, d\tau\right)^{-1} \in \mathbb{R}^{N\times N}$ is a symmetric estimation gain matrix generated as

$$\dot{\Gamma} = -\eta_c \Gamma \frac{\omega\omega^T}{1+\nu\omega^T\Gamma\omega}\Gamma; \quad \Gamma(t_r^+) = \Gamma(0) = \varphi_0 I, \tag{5–16}$$

where t_r^+ is the resetting time at which $\lambda_{min}\{\Gamma(t)\} \leq \varphi_1$, $\varphi_0 > \varphi_1 > 0$. The covariance resetting ensures that $\Gamma(t)$ is positive-definite for all time and prevents its value from becoming arbitrarily small in some directions, thus avoiding slow adaptation in some directions (also called the covariance wind-up problem) [104]. From Eq. 5–16, it is clear

that $\dot{\Gamma} \leq 0$, which means that the covariance matrix $\Gamma(t)$ can be bounded as

$$\varphi_1 I \quad \leq \quad \Gamma(t) \leq \varphi_0 I. \tag{5–17}$$

5.2.2 Gradient Update for the Actor

The actor update, like the critic update in Section 5.2.1, is based on the minimization of the Bellman error $\delta_{hjb}(\cdot)$. However, unlike the critic weights, the actor weights appear nonlinearly in $\delta_{hjb}(\cdot)$, making it problematic to develop a LS update law. Hence, a gradient update law is developed for the actor which minimizes the squared Bellman error $E_a(t) \triangleq \frac{1}{2}\delta_{hjb}^2$, whose gradient is given by

$$\frac{\partial E_a}{\partial \hat{W}_a} \quad = \quad \frac{\partial \delta_{hjb}}{\partial \hat{W}_a}\delta_{hjb} = \left(\hat{W}_c^T \phi' \frac{\partial \hat{F}_{\hat{u}}}{\partial \hat{u}} \frac{\partial \hat{u}}{\partial \hat{W}_a} + \hat{W}_a^T \phi' G \phi'^T\right) \delta_{hjb}, \tag{5–18}$$

where Eq. 5–11 is used, and $G(x) \triangleq g(x)R^{-1}g(x)^T \in \mathbb{R}^{n\times n}$ is a symmetric matrix. Using Eq. 5–18, the gradient-based update law for the actor NN is given by

$$\dot{\hat{W}}_a = \quad proj\left\{-\frac{\eta_{a1}}{\sqrt{1+\omega^T\omega}}\left(\hat{W}_c^T \phi' \frac{\partial \hat{F}_{\hat{u}}}{\partial \hat{u}} \frac{\partial \hat{u}}{\partial \hat{W}_a}\right)^T \delta_{hjb} - \frac{\eta_{a1}}{\sqrt{1+\omega^T\omega}} \phi' G \phi'^T \hat{W}_a \delta_{hjb} \right.$$
$$\left. -\eta_{a2}(\hat{W}_a - \hat{W}_c)\right\}, \tag{5–19}$$

where $proj\{\cdot\}$ is a projection operator used to bound the weight estimates [72], [73], $\eta_{a1}, \eta_{a2} \in \mathbb{R}$ are positive adaptation gains, $\frac{1}{\sqrt{1+\omega^T\omega}}$ is the normalization term, and the last term in Eq. 5–19 is added for stability (based on the subsequent stability analysis).

5.3 Identifier Design

The following assumption is made for the identifier design:

Assumption 5.8. *The control input is bounded i.e.* $u(t) \in \mathcal{L}_\infty$.

Remark 5.2. *Using Assumptions 5.2 and 5.5, and the projection algorithm in Eq. 5–19, Assumption 5.8 holds for the control design* $u(t) = \hat{u}(x)$ *in Eq. 5–10.*

Using Assumption 5.3, the dynamic system in Eq. 5–3, with control $\hat{u}(x)$, can be represented using a multi-layer NN as

$$\dot{x} = F_{\hat{u}}(x, \hat{u}) = W_f^T \sigma(V_f^T x) + \varepsilon_f(x) + g(x)\hat{u}, \tag{5–20}$$

where $W_f \in \mathbb{R}^{L_f+1 \times n}$, $V_f \in \mathbb{R}^{n \times L_f}$ are the unknown ideal NN weights, $\sigma_f \triangleq \sigma(V_f^T x) \in \mathbb{R}^{L_f+1}$ is the NN activation function, and $\varepsilon_f(x) \in \mathbb{R}^n$ is the function reconstruction error. The following multi-layer dynamic neural network (MLDNN) identifier is used to approximate the system in Eq. 5–20

$$\dot{\hat{x}} = \hat{F}_{\hat{u}}(x, \hat{x}, \hat{u}) = \hat{W}_f^T \hat{\sigma}_f + g(x)\hat{u} + \mu, \tag{5–21}$$

where $\hat{x}(t) \in \mathbb{R}^n$ is the DNN state, $\hat{\sigma}_f \triangleq \sigma(\hat{V}_f^T \hat{x}) \in \mathbb{R}^{L_f+1}$, $\hat{W}_f(t) \in \mathbb{R}^{L_f+1 \times n}$ and $\hat{V}_f(t) \in \mathbb{R}^{n \times L_f}$ are weight estimates, and $\mu(t) \in \mathbb{R}^n$ denotes the RISE feedback term defined as [47, 71]

$$\mu \triangleq k\tilde{x}(t) - k\tilde{x}(0) + v,$$

where $\tilde{x}(t) \triangleq x(t) - \hat{x}(t) \in \mathbb{R}^n$ is the identification error, and $v(t) \in \mathbb{R}^n$ is the generalized solution (in Filippov's sense [105]) to

$$\dot{v} = (k\alpha + \gamma)\tilde{x} + \beta_1 sgn(\tilde{x}); \quad v(0) = 0,$$

where k, α, γ, $\beta_1 \in \mathbb{R}$ are positive constant control gains, and $sgn(\cdot)$ denotes a vector signum function. The identification error dynamics can be written as

$$\dot{\tilde{x}} = \tilde{F}_u(x, \hat{x}, u) = W_f^T \sigma_f - \hat{W}_f^T \hat{\sigma}_f + \varepsilon_f(x) - \mu, \tag{5–22}$$

where $\tilde{F}_{\hat{u}}(x, \hat{x}, \hat{u}) \triangleq F_{\hat{u}}(x, \hat{u}) - \hat{F}_{\hat{u}}(x, \hat{x}, \hat{u}) \in \mathbb{R}^n$. A filtered identification error is defined as

$$r \triangleq \dot{\tilde{x}} + \alpha\tilde{x}. \tag{5–23}$$

Taking the time derivative of Eq. 5–23 and using Eq. 5–22 yields

$$\begin{aligned}\dot{r} &= W_f^T\sigma_f' V_f^T\dot{x} - \dot{\hat{W}}_f^T\hat{\sigma}_f - \hat{W}_f^T\hat{\sigma}_f'\dot{\hat{V}}_f^T\hat{x} - \hat{W}_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\hat{x}} + \dot{\varepsilon}_f(x) - kr - \gamma\tilde{x} \\ &\quad -\beta_1 sgn(\tilde{x}) + \alpha\dot{\tilde{x}}.\end{aligned} \tag{5–24}$$

Based on Eq. 5–24 and the subsequent stability analysis, the weight update laws for the DNN are designed as

$$\begin{aligned}\dot{\hat{W}}_f &= proj(\Gamma_{wf}\hat{\sigma}_f'\hat{V}_f^T\dot{\hat{x}}\tilde{x}^T), \\ \dot{\hat{V}}_f &= proj(\Gamma_{vf}\dot{\hat{x}}\tilde{x}^T\hat{W}_f^T\hat{\sigma}_f'),\end{aligned} \tag{5–25}$$

where $proj(\cdot)$ is a smooth projection operator [72], [73], and $\Gamma_{wf} \in \mathbb{R}^{L_f+1\times L_f+1}$, $\Gamma_{vf} \in \mathbb{R}^{n\times n}$ are positive constant adaptation gain matrices. The expression in Eq. 5–24 can be rewritten as

$$\dot{r} = \tilde{N} + N_{B1} + \hat{N}_{B2} - kr - \gamma\tilde{x} - \beta_1 sgn(\tilde{x}), \tag{5–26}$$

where the auxiliary signals, $\tilde{N}(x,\tilde{x},r,\hat{W}_f,\hat{V}_f,t)$, $N_{B1}(x,\hat{x},\hat{W}_f,\hat{V}_f,t)$, and $\hat{N}_{B2}(\hat{x},\dot{\hat{x}},\hat{W}_f,\hat{V}_f,t) \in \mathbb{R}^n$ are defined as

$$\tilde{N} \triangleq \alpha\dot{\tilde{x}} - \dot{\hat{W}}_f^T\hat{\sigma}_f - \hat{W}_f^T\hat{\sigma}_f'\dot{\hat{V}}_f^T\hat{x} + \frac{1}{2}W_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\tilde{x}} + \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'V_f^T\dot{\tilde{x}}, \tag{5–27}$$

$$N_{B1} \triangleq W_f^T\sigma_f' V_f^T\dot{x} - \frac{1}{2}W_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{x} - \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'V_f^T\dot{x} + \dot{\varepsilon}_f(x), \tag{5–28}$$

$$\hat{N}_{B2} \triangleq \frac{1}{2}\tilde{W}_f^T\hat{\sigma}_f'\hat{V}_f^T\dot{\hat{x}} + \frac{1}{2}\hat{W}_f^T\hat{\sigma}_f'\tilde{V}_f^T\dot{\hat{x}}, \tag{5–29}$$

where $\tilde{W}_f \triangleq W_f - \hat{W}_f(t) \in \mathbb{R}^{L_f+1\times n}$ and $\tilde{V}_f \triangleq V_f - \hat{V}_f(t) \in \mathbb{R}^{n\times L_f}$. To facilitate the subsequent stability analysis, an auxiliary term $N_{B2}(\hat{x},\dot{x},\hat{W}_f,\hat{V}_f,t) \in \mathbb{R}^n$ is defined by replacing $\dot{\hat{x}}(t)$ in $\hat{N}_{B2}(\cdot)$ by $\dot{x}(t)$, and $\tilde{N}_{B2}(\hat{x},\dot{\tilde{x}},\hat{W}_f,\hat{V}_f,t) \triangleq \hat{N}_{B2}(\cdot) - N_{B2}(\cdot)$. The terms $N_{B1}(\cdot)$ and $N_{B2}(\cdot)$ are grouped as $N_B \triangleq N_{B1} + N_{B2}$. Using Assumptions 5.2, 5.4-5.6, and Eqs. 5–23, 5–25, 5–28 and 5–29, the following bounds can be obtained

$$\left\|\tilde{N}\right\| \leq \rho_1(\|z\|)\|z\|, \tag{5–30}$$

$$\begin{aligned} \|N_{B1}\| &\leq \zeta_1, \quad \|N_{B2}\| \leq \zeta_2, \\ \left\|\dot{N}_B\right\| &\leq \zeta_3 + \zeta_4 \rho_2(\|z\|)\|z\|, \end{aligned} \tag{5–31}$$

$$\left\|\dot{\tilde{x}}^T \tilde{N}_{B2}\right\| \leq \zeta_5 \|\tilde{x}\|^2 + \zeta_6 \|r\|^2, \tag{5–32}$$

where $z \triangleq \left[\tilde{x}^T \; r^T\right]^T \in \mathbb{R}^{2n}$, $\rho_1(\cdot), \rho_2(\cdot) \in \mathbb{R}$ are positive, globally invertible, non-decreasing functions, and $\zeta_i \in \mathbb{R}$, $i = 1, ..., 6$ are computable positive constants. To facilitate the subsequent stability analysis, let $\mathcal{D} \subset \mathbb{R}^{2n+2}$ be a domain containing $y(t) = 0$, where $y(t) \in \mathbb{R}^{2n+2}$ is defined as

$$y \triangleq \left[\tilde{x}^T \;\; r^T \;\; \sqrt{P} \;\; \sqrt{Q}\right]^T, \tag{5–33}$$

where the auxiliary function $P(z,t) \in \mathbb{R}$ is the generalized solution to the differential equation

$$\dot{P} = -L, \qquad P(0) = \beta_1 \sum_{i=1}^{n} |\tilde{x}_i(0)| - \tilde{x}^T(0) N_B(0), \tag{5–34}$$

where the auxiliary function $L(z,t) \in \mathbb{R}$ is defined as

$$L \triangleq r^T(N_{B1} - \beta_1 sgn(\tilde{x})) + \dot{\tilde{x}}^T N_{B2} - \beta_2 \rho_2(\|z\|)\|z\|\|\tilde{x}\|, \tag{5–35}$$

where $\beta_1, \beta_2 \in \mathbb{R}$ are chosen according to the following sufficient conditions to ensure $P(z,t) \geq 0$ [71]

$$\beta_1 > \max(\zeta_1 + \zeta_2, \; \zeta_1 + \frac{\zeta_3}{\alpha}), \qquad \beta_2 > \zeta_4. \tag{5–36}$$

The auxiliary function $Q(\tilde{W}_f, \tilde{V}_f) \in \mathbb{R}$ in Eq. 5–33 is defined as

$$Q \triangleq \frac{1}{4}\alpha \left[tr(\tilde{W}_f^T \Gamma_{wf}^{-1} \tilde{W}_f) + tr(\tilde{V}_f^T \Gamma_{vf}^{-1} \tilde{V}_f)\right],$$

where $tr(\cdot)$ denotes the trace of a matrix.

Theorem 5.1. *For the system in Eq. 5–3, the identifier developed in Eq. 5–21 along with the weight update laws in Eq. 5–25 ensures asymptotic identification of the state and its derivative, in the sense that*

$$\lim_{t\to\infty} \|\tilde{x}(t)\| = 0 \qquad \text{and} \qquad \lim_{t\to\infty} \left\|\dot{\tilde{x}}(t)\right\| = 0,$$

provided the control gains k and γ are chosen sufficiently large based on the initial conditions of the states[1] *, and satisfy the following sufficient conditions*

$$\gamma > \frac{\zeta_5}{\alpha}, \qquad k > \zeta_6, \tag{5–37}$$

where ζ_5 and ζ_6 are introduced in Eq. 5–32, and β_1, β_2 introduced in Eq. 5–35, are chosen according to the sufficient conditions in Eq. 5–36.

Proof. The proof is similar to the proof of Theorem 4.1, the difference being that $g(x)$ is assumed to be exactly known in this chapter. This simplifies the design of the identifier, where $g(x)$ is directly used, unlike in Chapter 4 where its NN estimate is used instead. □

Using the developed identifier in Eq. 5–21, the actor weight update law can now be simplified using Eq. 5–19 as

$$\dot{\hat{W}}_a = proj\left\{-\frac{\eta_{a1}}{\sqrt{1+\omega^T\omega}}\phi' G \phi'^T \left(\hat{W}_a - \hat{W}_c\right)\delta_{hjb} - \eta_{a2}(\hat{W}_a - \hat{W}_c)\right\}. \tag{5–38}$$

5.4 Convergence and Stability Analysis

The unmeasurable form of the Bellman error can be written using Eqs. 5–5-5–8 and Eq. 5–11, as

$$\begin{aligned}\delta_{hjb} &= \hat{W}_c^T\omega - W_c^T\phi' F_{u^*} + \hat{u}^T R\hat{u} - u^{*T} R u^* - \varepsilon'_v F_{u^*}. \\ &= -\tilde{W}_c^T\omega - W^T\phi'\tilde{F}_{\hat{u}} + \frac{1}{4}\tilde{W}_a^T\phi' G\phi'^T\tilde{W}_a - \frac{1}{4}\varepsilon'_v G\varepsilon'^T_v - \varepsilon'_v F_{u^*},\end{aligned} \tag{5–39}$$

[1] See subsequent semi-global stability analysis.

where Eqs. 5–9 and 5–10 are used. The dynamics of the critic weight estimation error $\tilde{W}_c(t)$ can now be developed by substituting Eq. 5–39 in Eq. 5–15, as

$$\dot{\tilde{W}}_c = -\eta_c \Gamma \psi \psi^T \tilde{W}_c + \eta_c \Gamma \frac{\omega}{1+\nu \omega^T \Gamma \omega} \left[-W^T \phi' \tilde{F}_{\hat{u}} + \frac{1}{4} \tilde{W}_a^T \phi' G \phi'^T \tilde{W}_a - \frac{1}{4} \varepsilon_v' G \varepsilon_v'^T - \varepsilon_v' F_{u^*} \right. \\ \left. - \frac{1}{4} \varepsilon_v' G \varepsilon_v'^T - \varepsilon_v' F_{u^*} \right], \tag{5–40}$$

where $\psi(t) \triangleq \frac{\omega(t)}{\sqrt{1+\nu \omega(t)^T \Gamma(t) \omega(t)}} \in \mathbb{R}^N$ is the normalized critic regressor vector, bounded as

$$\|\psi\| \leq \frac{1}{\sqrt{\nu \varphi_1}}, \tag{5–41}$$

where φ_1 is introduced in Eq. 5–17. The error system in Eq. 5–40 can be represented by the following perturbed system

$$\dot{\tilde{W}}_c = \Omega_{nom} + \Delta_{per}, \tag{5–42}$$

where $\Omega_{nom}(\tilde{W}_c, t) \triangleq -\eta_c \Gamma \psi \psi^T \tilde{W}_c \in \mathbb{R}^N$, denotes the nominal system, and $\Delta_{per}(t) \triangleq \eta_c \Gamma \frac{\omega}{1+\nu \omega^T \Gamma \omega} \left[-W^T \phi' \tilde{F}_{\hat{u}} + \frac{1}{4} \tilde{W}_a^T \phi' G \phi'^T \tilde{W}_a - \frac{1}{4} \varepsilon_v' G \varepsilon_v'^T - \varepsilon_v' F_{u^*} \right] \in \mathbb{R}^N$ denotes the perturbation. Using Theorem 2.5.1 in [104], the nominal system

$$\dot{\tilde{W}}_c = -\eta_c \Gamma \psi \psi^T \tilde{W}_c \tag{5–43}$$

is globally exponentially stable, if the bounded signal $\psi(t)$ is PE, i.e.

$$\mu_2 I \geq \int_{t_0}^{t_0+\delta} \psi(\tau) \psi(\tau)^T d\tau \geq \mu_1 I \qquad \forall t_0 \geq 0,$$

for some positive constants $\mu_1, \mu_2, \delta \in \mathbb{R}$. Since $\Omega_{nom}(\tilde{W}_c, t)$ is continuously differentiable and the Jacobian $\frac{\partial \Omega_{nom}}{\partial \tilde{W}_c} = -\eta_c \Gamma \psi \psi^T$ is bounded for the exponentially stable system in Eq. 5–43, the converse Lyapunov Theorem 4.14 in [106] can be used to show that there exists

a function $V_c : \mathbb{R}^N \times [0,\ \infty) \to \mathbb{R}$, which satisfies the following inequalities

$$\begin{aligned} c_1 \left\|\tilde{W}_c\right\|^2 &\leq V_c(\tilde{W}_c, t) \leq c_2 \left\|\tilde{W}_c\right\|^2 \\ \frac{\partial V_c}{\partial t} + \frac{\partial V_c}{\partial \tilde{W}_c}\Omega_{nom}(\tilde{W}_c, t) &\leq -c_3 \left\|\tilde{W}_c\right\|^2 \\ \left\|\frac{\partial V_c}{\partial \tilde{W}_c}\right\| &\leq c_4 \left\|\tilde{W}_c\right\|, \end{aligned} \tag{5–44}$$

for some positive constants $c_1,\ c_2,\ c_3,\ c_4 \in \mathbb{R}$. Using Assumptions 5.2, 5.4-5.6 and 5.8, the projection bounds in Eq. 5–19, the fact that $F_{u^*} \in \mathcal{L}_\infty$ (since $u^*(x)$ is stabilizing), and provided the conditions of Theorem 1 hold (required to prove that $\tilde{F}_{\hat{u}} \in \mathcal{L}_\infty$), the following bounds can be developed:

$$\begin{aligned} \left\|\tilde{W}_a\right\| \leq \kappa_1, \qquad \left\|\phi' G \phi'^T\right\| \leq \kappa_2, \\ \left\|\frac{1}{4}\tilde{W}_a^T \phi' G \phi'^T \tilde{W}_a - W^T \phi' \tilde{F}_{\hat{u}} - \varepsilon_v' F_{u^*}\right\| \leq \kappa_3, \\ \left\|\frac{1}{2}W^T \phi' G \varepsilon_v'^T + \frac{1}{2}\varepsilon_v' G \varepsilon_v'^T + \frac{1}{2}W^T \phi' G \phi'^T \tilde{W}_a + \frac{1}{2}\varepsilon_v' G \phi'^T\right\| \leq \kappa_4, \end{aligned} \tag{5–45}$$

where $\kappa_1, \kappa_2, \kappa_3, \kappa_4 \in \mathbb{R}$ are computable positive constants.

Theorem 5.2. *If Assumptions 5.1-5.8 hold, the normalized critic regressor $\psi(t)$ defined in 5–40 is PE (persistently exciting), and provided Eq. 5–36, Eq. 5–37 and the following sufficient gain condition is satisfied*[2]

$$\frac{c_3}{\eta_{a1}} > \kappa_1 \kappa_2, \tag{5–46}$$

where $\eta_{a1},\ c_3,\ \kappa_1,\ \kappa_2$ are introduced in Eqs. 5–19, 5–44, and 5–45, then the controller in Eq. 5–10, the actor and critic weight update laws in Eqs. 5–15-5–16 and 5–38, and the

[2] Since c_3 is a function of the critic adaptation gain η_c, η_{a1} is the actor adaptation gain, and κ_1, κ_2 are known constants, the sufficient gain condition in Eq. 5–46 can be easily satisfied.

identifier in Eq. 5–21 and 5–25, guarantee that the state of the system $x(t)$, and the actor and critic weight estimation errors $\tilde{W}_a(t)$ and $\tilde{W}_c(t)$ are UUB.

Proof. To investigate the stability of Eq. 5–3 with control $\hat{u}(x)$, and the perturbed system in Eq. 5–42, consider $V_L : \mathcal{X} \times \mathbb{R}^N \times \mathbb{R}^N \times [0, \infty) \to \mathbb{R}$ as the continuously differentiable, positive-definite Lyapunov function candidate defined as

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

where $V^*(x)$ (the optimal value function), is the Lyapunov function for Eq. 5–3, and $V_c(\tilde{W}_c, t)$ is the Lyapunov function for the exponentially stable system in Eq. 5–43. Since $V^*(x)$ is continuously differentiable and positive-definite from Eq. 5–1 and 5–2, there exist class $\mathcal{K}$ functions α_1 and α_2 defined on $[0,\ r]$, where $B_r \subset \mathcal{X}$ (Lemma 4.3 in [106]), such that

$$\alpha_1(\|x\|) \leq V^*(x) \leq \alpha_2(\|x\|) \qquad \forall x \in B_r. \tag{5–47}$$

Using Eqs. 5–44 and 5–47, $V_L(x, \tilde{W}_c, \tilde{W}_a, t)$ can be bounded as

$$\alpha_1(\|x\|) + c_1 \left\|\tilde{W}_c\right\|^2 + \frac{1}{2}\left\|\tilde{W}_a\right\|^2 \leq V_L(x, \tilde{W}_c, \tilde{W}_a, t) \leq \alpha_2(\|x\|) + c_2 \left\|\tilde{W}_c\right\|^2 + \frac{1}{2}\left\|\tilde{W}_a\right\|^2,$$

which can be written as

$$\alpha_3(\|\tilde{z}\|) \leq V_L(x, \tilde{W}_c, \tilde{W}_a, t) \leq \alpha_4(\|\tilde{z}\|) \qquad \forall \tilde{z} \in B_s,$$

where $\tilde{z}(t) \triangleq [x(t)^T\ \tilde{W}_c(t)^T\ \tilde{W}_a(t)^T]^T \in \mathbb{R}^{n+2N}$, α_3 and α_4 are class $\mathcal{K}$ functions defined on $[0,\ s]$, where $B_s \subset \mathcal{X} \times \mathbb{R}^N \times \mathbb{R}^N$. Taking the time derivative of $V_L(\cdot)$ yields

$$\dot{V}_L = \frac{\partial V^*}{\partial x} f + \frac{\partial V^*}{\partial x} g\hat{u} + \frac{\partial V_c}{\partial t} + \frac{\partial V_c}{\partial \tilde{W}_c}\Omega_{nom} + \frac{\partial V_c}{\partial \tilde{W}_c}\Delta_{per} - \tilde{W}_a^T \dot{\hat{W}}_a, \tag{5–48}$$

where the time derivative of $V^*(\cdot)$ is taken along the the trajectories of the system Eq. 5–3 with control $\hat{u}(\cdot)$ and the time derivative of $V_c(\cdot)$ is taken along the along the trajectories of the perturbed system Eq. 5–42. To facilitate the subsequent analysis, the HJB in Eq.

5–5 is rewritten as $\frac{\partial V^*}{\partial x} f = -\frac{\partial V^*}{\partial x} g u^* - Q(x) - u^{*^T} R u^*$. Substituting for $\frac{\partial V^*}{\partial x} f$ in Eq. 5–48, using the fact that $\frac{\partial V^*}{\partial x} g = -2u^{*^T} R$ from Eq. 5–4, and using Eqs. 5–19 and 5–44, Eq. 5–48) can be upper bounded as

$$\begin{aligned} \dot{V}_L &\leq -Q - u^{*^T} R u^* - c_3 \left\| \tilde{W}_c \right\|^2 + c_4 \left\| \tilde{W}_c \right\| \|\Delta_{per}\| + 2u^{*^T} R(u^* - \hat{u}) \\ &+ \eta_{a2} \tilde{W}_a^T (\hat{W}_a - \hat{W}_c) + \frac{\eta_{a1}}{\sqrt{1+\omega^T \omega}} \tilde{W}_a^T \phi' G \phi'^T (\hat{W}_a - \hat{W}_c) \delta_{hjb}. \end{aligned} \tag{5–49}$$

Substituting for u^*, $\hat{u}$, δ_{hjb}, and Δ_{per} using Eqs. 5–4, 5–10, 5–39, and 5–42, respectively, and using Eq. 5–17 and Eq. 5–41 in Eq. 5–49, yields

$$\begin{aligned} \dot{V}_L \leq & -Q - c_3 \left\| \tilde{W}_c \right\|^2 - \eta_{a2} \left\| \tilde{W}_a \right\|^2 + \frac{1}{2} W^T \phi' G \varepsilon_v'^T + \frac{1}{2} \varepsilon_v' G \varepsilon_v'^T + \frac{1}{2} W^T \phi' G \phi'^T \tilde{W}_a \\ & + c_4 \frac{\eta_c \varphi_0}{2\sqrt{\nu \varphi_1}} \left\| -W^T \phi' \tilde{F}_{\hat{u}} + \frac{1}{4} \tilde{W}_a^T \phi' G \phi'^T \tilde{W}_a - \frac{1}{4} \varepsilon_v' G \varepsilon_v'^T - \varepsilon_v' F_{u^*} \right\| \left\| \tilde{W}_c \right\| \\ & + \frac{\eta_{a1}}{\sqrt{1+\omega^T \omega}} \tilde{W}_a^T \phi' G \phi'^T (\tilde{W}_c - \tilde{W}_a) \left(-\tilde{W}_c^T \omega - W^T \phi' \tilde{F}_{\hat{u}} + \frac{1}{4} \tilde{W}_a^T \phi' G \phi'^T \tilde{W}_a \right. \\ & \left. - \frac{1}{4} \varepsilon_v' G \varepsilon_v'^T - \varepsilon_v' F_{u^*} \right) + \frac{1}{2} \varepsilon_v' G \phi'^T \tilde{W}_a + \eta_{a2} \left\| \tilde{W}_a \right\| \left\| \tilde{W}_c \right\|. \end{aligned} \tag{5–50}$$

Using the bounds developed in Eqs. 5–45, 5–50 can be further upper bounded as

$$\begin{aligned} \dot{V}_L &\leq -Q - (c_3 - \eta_{a1} \kappa_1 \kappa_2) \left\| \tilde{W}_c \right\|^2 - \eta_{a2} \left\| \tilde{W}_a \right\|^2 + \eta_{a1} \kappa_1^2 \kappa_2 \kappa_3 + \kappa_4 \\ &+ \left(\frac{c_4 \eta_c \varphi_0}{2\sqrt{\nu \varphi_1}} \kappa_3 + \eta_{a1} \kappa_1 \kappa_2 \kappa_3 + \eta_{a1} \kappa_1^2 \kappa_2 + \eta_{a2} \kappa_1 \right) \left\| \tilde{W}_c \right\|. \end{aligned}$$

Provided $c_3 > \eta_{a1} \kappa_1 \kappa_2$, and completing the square yields

$$\begin{aligned} \dot{V}_L &\leq -Q - (1-\theta)(c_3 - \eta_{a1} \kappa_1 \kappa_2) \left\| \tilde{W}_c \right\|^2 - \eta_{a2} \left\| \tilde{W}_a \right\|^2 + \eta_{a1} \kappa_1^2 \kappa_2 \kappa_3 + \kappa_4 \\ &+ \frac{1}{4\theta(c_3 - \eta_{a1} \kappa_1 \kappa_2)} \left[\frac{c_4 \eta_c \varphi_0}{2\sqrt{\nu \varphi_1}} \kappa_3 + \eta_{a1} \kappa_1 \kappa_2 \kappa_3 + \eta_{a1} \kappa_1^2 \kappa_2 + \eta_{a2} \kappa_1 \right]^2 \end{aligned} \tag{5–51}$$

where $0 < \theta < 1$. Since $Q(x)$ is positive definite, Lemma 4.3 in [106] indicates that there exist class $\mathcal{K}$ functions α_5 and α_6 such that

$$\begin{aligned} \alpha_5(\|\tilde{z}\|) &\leq Q + (1-\theta)(c_3 - \eta_{a1} \kappa_1 \kappa_2) \left\| \tilde{W}_c \right\|^2 + \eta_{a2} \left\| \tilde{W}_a \right\|^2 \\ &\leq \alpha_6(\|\tilde{z}\|) \qquad \forall v \in B_s. \end{aligned} \tag{5–52}$$

Using Eq. 5–52, the expression in Eq. 5–51 can be further upper bounded as

$$\begin{aligned}\dot{V}_L &\leq -\alpha_5(\|\tilde{z}\|) + \eta_{a1}\kappa_1^2\kappa_2\kappa_3 + \kappa_4 \\ &+ \frac{1}{4\theta(c_3 - \eta_{a1}\kappa_1\kappa_2)}\left[\frac{c_4\eta_c\varphi_0}{2\sqrt{\nu\varphi_1}}\kappa_3 + \eta_{a1}\kappa_1\kappa_2\kappa_3 + \eta_{a1}\kappa_1^2\kappa_2 + \eta_{a2}\kappa_1\right]^2,\end{aligned}$$

which proves that $\dot{V}_L(\cdot)$ is negative whenever $\tilde{z}(t)$ lies outside the compact set $\Omega_{\tilde{z}} \triangleq \left\{\tilde{z} : \|\tilde{z}\| \leq \alpha_5^{-1}\left(\frac{1}{4\theta(c_3-\eta_{a1}\kappa_1\kappa_2)}\left[\frac{c_4\eta_c\varphi_0}{2\sqrt{\nu\varphi_1}}\kappa_3 + \eta_{a1}\kappa_1\kappa_2\kappa_3 + \eta_{a1}\kappa_1^2\kappa_2 + \eta_{a2}\kappa_1 + \eta_{a1}\kappa_1^2\kappa_2\kappa_3 + \kappa_4\right]^2\right)\right\}$, and hence, $\|\tilde{z}(t)\|$ is UUB (Theorem 4.18 in [106]). The bounds in Eq. 5–45 depend on the actor NN approximation error ε_v', which can be reduced by increasing the number of neurons N, thereby reducing the size of the residual set $\Omega_{\tilde{z}}$. From Assumption 5.7, as the number of neurons of the actor and critic NNs $N \to \infty$, the reconstruction error $\varepsilon_v' \to 0$. □

Remark 5.3. *Since the actor, critic and identifier are continuously updated, the developed RL algorithm can be compared to fully optimistic PI in machine learning literature [107], where policy evaluation and policy improvement are done after every state transition, unlike traditional PI, where policy improvement is done after convergence of the policy evaluation step. Proving convergence of optimistic PI is complicated and is an active area of research in machine learning [107, 108]. By considering an adaptive control framework, this result investigates the convergence and stability behavior of fully optimistic PI in continuous-time.*

Remark 5.4. *The PE condition in Theorem 2 is equivalent to the exploration paradigm in RL which ensures sufficient sampling of the state space and convergence to the optimal policy [101].*

5.5 Comparison with Related Work

Similar to RL, optimal control involves selection of an optimal policy based on some long-term performance criteria. DP provides a means to solve optimal control problems [52]; however, DP is implemented backward in time, making it offline and computationally expensive for complex systems. Owing to the similarities between optimal control and

RL [3], Werbos [17] introduced RL-based AC methods for optimal control, called ADP. ADP uses NNs to approximately solve DP forward-in-time, thus avoiding the *curse of dimensionality*. A detailed discussion of ADP-based designs is found in [6, 24, 107]. The success of ADP prompted a major research effort towards designing ADP-based optimal feedback controllers. The discrete/iterative nature of the ADP formulation lends itself naturally to the design of discrete-time optimal controllers [7, 10, 67–70, 109].

Extensions of ADP-based controllers to continuous-time systems entails challenges in proving stability, convergence, and ensuring the algorithm is online and model-free. Early solutions to the problem consisted of using a discrete-time formulation of time and state, and then applying an RL algorithm on the discretized system. Discretizing the state space for high dimensional systems requires a large memory space and a computationally prohibitive learning process. Baird [38] proposed *Advantage Updating*, an extension of the Q-learning algorithm which could be implemented in continuous-time and provided faster convergence. Doya [39] used a HJB framework to derive algorithms for value function approximation and policy improvement, based on a continuous-time version of the temporal difference error. Murray et al. [8] also used the HJB framework to develop a *stepwise stable* iterative ADP algorithm for continuous-time input-affine systems with an input quadratic performance measure. In Beard et al. [40], Galerkin's spectral method is used to approximate the solution to the GHJB, using which a stabilizing feedback controller was computed offline. Similar to [40], Abu-Khalaf and Lewis [41] proposed a least-squares successive approximation solution to the GHJB, where an NN is trained offline to learn the GHJB solution. Recent results by [13, 42] have made new inroads by addressing the problem for partially unknown nonlinear systems. However, the inherently iterative nature of the ADP algorithm has prevented the development of rigorous stability proofs of closed-loop controllers for continuous-time uncertain nonlinear systems.

All the aforementioned approaches for continuous-time nonlinear systems are offline and/or require complete knowledge of system dynamics. One of the contributions in

[13] is that only partial knowledge of the system dynamics is required, and a hybrid continuous-time/discrete-time sampled data controller is developed based on PI, where the feedback control operation of the actor occurs at faster time scale than the learning process of the critic. Vamvoudakis and Lewis [14] extended the idea by designing a model-based online algorithm called *synchronous PI* which involved synchronous, continuous-time adaptation of both actor and critic neural networks. Inspired by the work in [14], a novel actor-critic-identifier architecture is proposed in this work to approximately solve the continuous-time infinite horizon optimal control problem for uncertain nonlinear systems; however, unlike [14], the developed method does not require knowledge of the system drift dynamics. The actor and critic NNs approximate the optimal control and the optimal value function, respectively, whereas the identifier DNN estimates the system dynamics online. The integral RL technique in [13] leads to a hybrid continuous-time/discrete-time controller with two time-scale actor and critic learning process, whereas the approach in [14], although continuous-time, requires complete knowledge of system dynamics. A contribution of this work is the use of a novel actor-critic-identifier architecture, which obviates the need to know the system drift dynamics, and where the learning of the actor, critic and identifier is continuous and simultaneous. Moreover, the actor-critic-identifier method utilizes an identification-based online learning scheme, and hence is the first ever indirect adaptive control approach to RL. The idea is similar to the *Heuristic Dynamic Programming* (HDP) algorithm [5], where Werbos suggested the use of a model network along with the actor and critic networks.

In the developed method, the actor and critic NNs use gradient and least squares-based update laws, respectively, to minimize the Bellman error, which is the difference between the exact and the approximate HJB equation. The identifier DNN is a combination of a Hopfield-type [110] component, in parallel configuration with the system [111], and a novel RISE (Robust Integral of Sign of the Error) component. The Hopfield component of

the DNN learns the system dynamics based on online gradient-based weight tuning laws, while the RISE term robustly accounts for the function reconstruction errors, guaranteeing asymptotic estimation of the state and the state derivative. The online estimation of the state derivative allows the actor-critic-identifier architecture to be implemented without knowledge of system drift dynamics; however, knowledge of the input gain matrix is required to implement the control policy. While the design of the actor and critic are coupled through the HJB equation, the design of the identifier is decoupled from actor-critic, and can be considered as a modular component in the actor-critic-identifier architecture. Convergence of the actor-critic-identifier-based algorithm and stability of the closed-loop system are analyzed using Lyapunov-based adaptive control methods, and a *persistence of excitation* (PE) condition is used to guarantee exponential convergence to a bounded region in the neighborhood of the optimal control and UUB stability of the closed-loop system. The PE condition is equivalent to the exploration paradigm in RL [101] and ensures adequate sampling of the system's dynamics, required for convergence to the optimal policy.

5.6 Simulation

5.6.1 Nonlinear System Example

The following nonlinear system is considered [14]

$$\dot{x} = \begin{bmatrix} -x_1 + x_2 \\ -0.5x_1 - 0.5x_2(1 - (cos(2x_1) + 2)^2) \end{bmatrix} + \begin{bmatrix} 0 \\ cos(2x_1) + 2 \end{bmatrix} u, \qquad (5\text{–}53)$$

where $x(t) \triangleq [x_1(t) \ x_2(t)]^T \in \mathbb{R}^2$ and $u(t) \in \mathbb{R}$. The state and control penalties are chosen as

$$Q(x) = x^T \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x; \qquad R = 1.$$

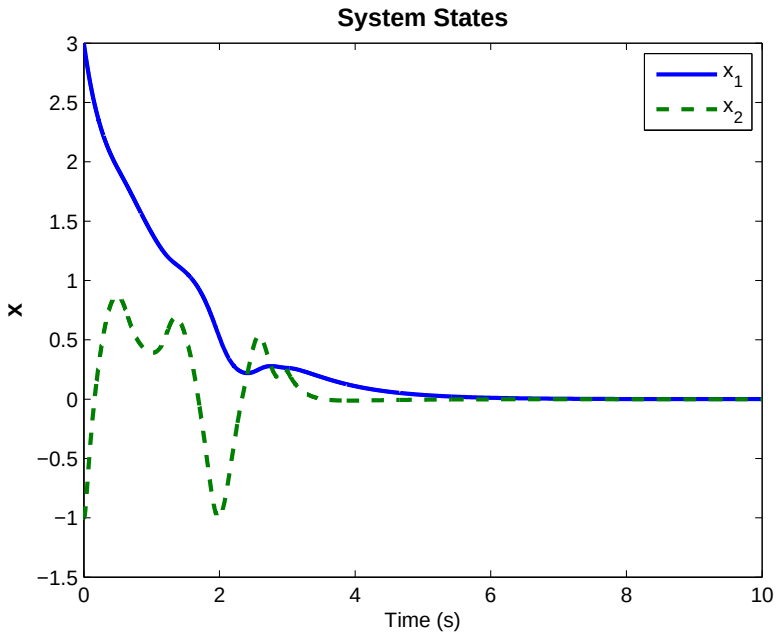

Figure 5-2. System states $x(t)$ with persistently excited input for the first 3 seconds.

The optimal value function and optimal control for the system in Eq. 5–53 are known, and given by [14]

$$V^*(x) = \frac{1}{2}x_1^2 + x_2^2; \qquad u^*(x) = -(cos(2x_1) + 2)x_2.$$

The activation function for the critic NN is selected with $N = 3$ neurons as

$$\phi(x) = [x_1^2 \quad x_1x_2 \quad x_2^2]^T,$$

while the activation function for the identifier DNN is selected as a symmetric sigmoid with $L_f = 5$ neurons in the hidden layer. The identifier gains are selected as

$$k = 800, \quad \alpha = 300, \quad \gamma = 5, \quad \beta_1 = 0.2, \quad \Gamma_{wf} = 0.1\mathbb{I}_{6\times 6}, \quad \Gamma_{vf} = 0.1\mathbb{I}_{2\times 2},$$

and the gains for the actor-critic learning laws are selected as

$$\eta_{a1} = 10, \quad \eta_{a2} = 50, \quad \eta_c = 20, \quad \nu = 0.005.$$

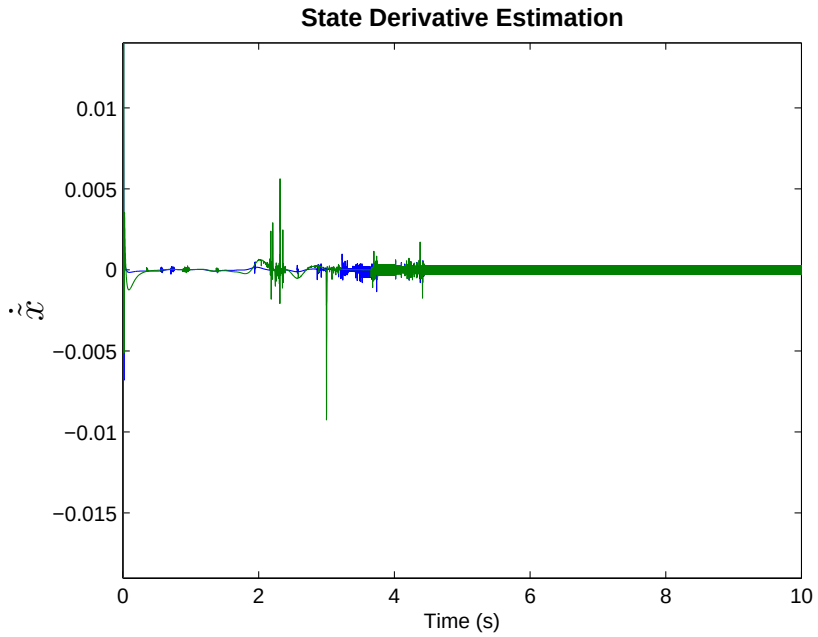

Figure 5-3. Error in estimating the state derivative $\dot{\tilde{x}}(t)$ by the identifier.

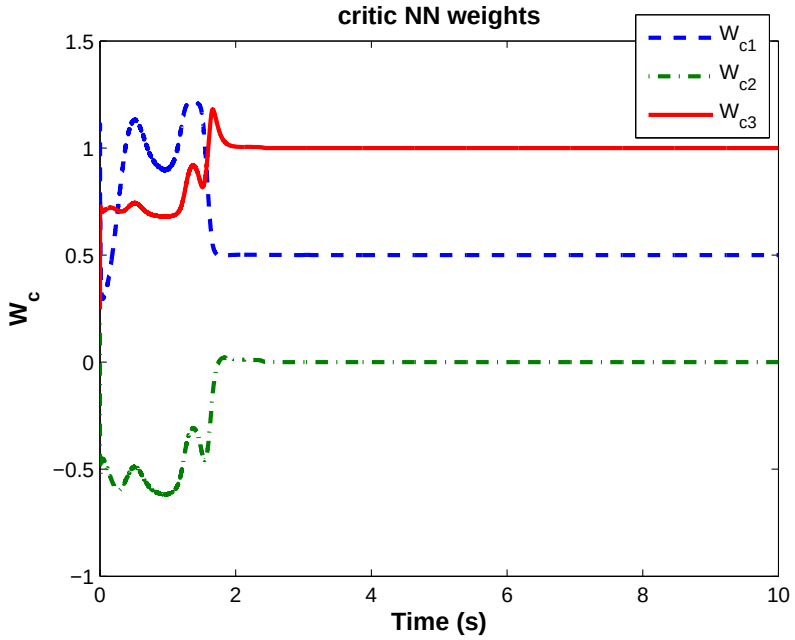

Figure 5-4. Convergence of critic weights $\hat{W}_c(t)$.

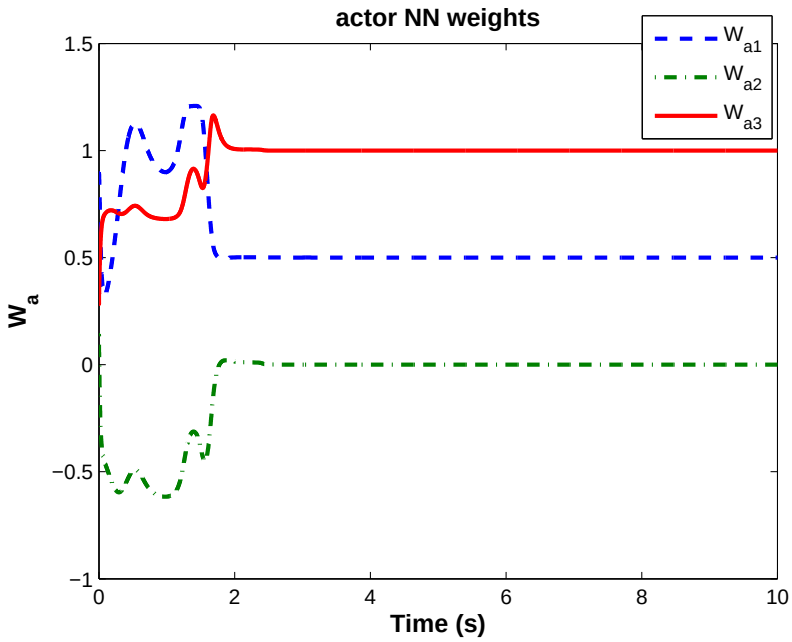

Figure 5-5. Convergence of actor weights $\hat{W}_a(t)$.

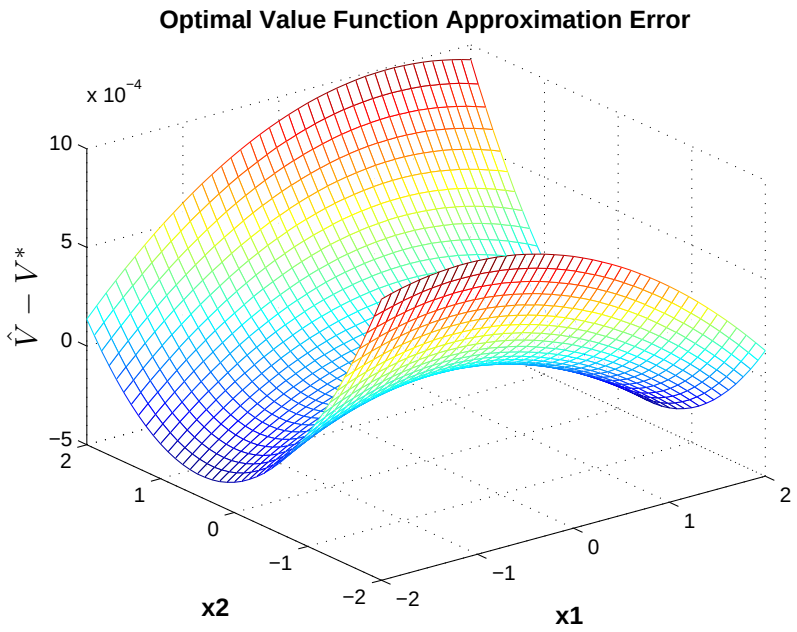

Figure 5-6. Error in approximating the optimal value function by the critic at steady state.

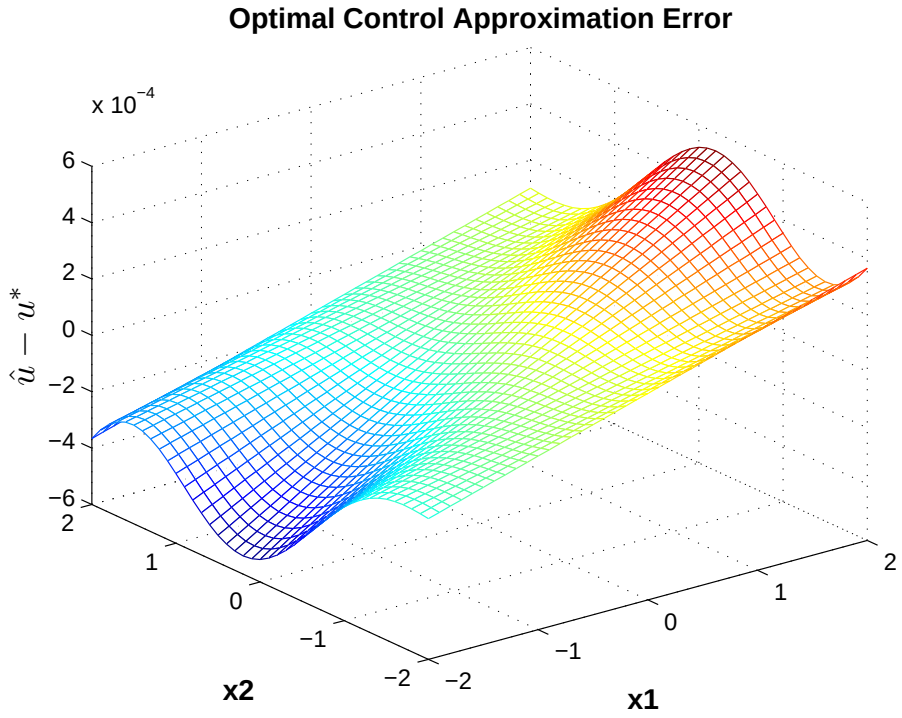

Figure 5-7. Error in approximating the optimal control by the actor at steady state.

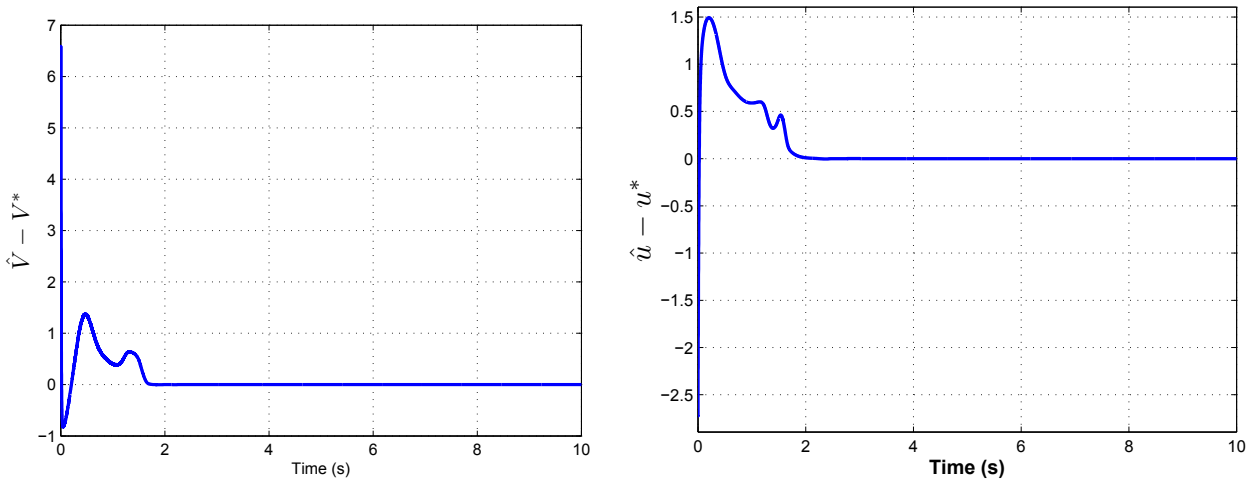

Figure 5-8. Errors in approximating the (a) optimal value function, and (b) optimal control, as a function of time.

The covariance matrix is initialized to $\Gamma(0) = 5000$, all the NN weights are randomly initialized in $[-1,\ 1]$, and the states are initialized to $x(0) = [3,\ -1]$. An implementation issue in using the developed algorithm is to ensure PE of the critic regressor vector. Unlike linear systems, where PE of the regressor translates to sufficient richness of the external input, no verifiable method exists to ensure PE in nonlinear regulation problems. To ensure PE qualitatively, a small exploratory signal consisting of sinusoids of varying frequencies, $n(t) = sin^2(t)cos(t)+sin^2(2t)cos(0.1t)+sin^2(-1.2t)cos(0.5t)+sin^5(t)$, is added to the control $u(t)$ for the first 3 seconds [14]. The evolution of states is shown in Fig. 5-2. The identifier approximates the system dynamics, and the state derivative estimation error is shown in Fig. 5-3. Persistence of excitation ensures that the weights converge converge to their optimal values of $W = [0.5\ \ 0\ \ 1]^T$ in approximately 2 seconds, as seen from the evolution of actor and critic weights in Figs. 5-4 and 5-5. The errors in approximating the optimal value function and optimal control at steady state ($t = 10\ sec.$) are plotted against the states in Figs. 5-6 and 5-7, respectively. Fig. 5-8 shows the error between the optimal value function and approximate optimal value function, and the optimal control and approximate optimal control, as a function of time along the trajectory $x(t)$.

5.6.2 LQR Example

The following linear system is considered [14]

$$\dot{x} = \underbrace{\begin{bmatrix} -1.01887 & 0.90506 & -0.00215 \\ 0.82225 & -1.07741 & -0.17555 \\ 0 & 0 & 1 \end{bmatrix}}_{A} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{B} u.$$

with the following state and the control penalties

$$Q(x) = x^T \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} x; \qquad R = 1.$$

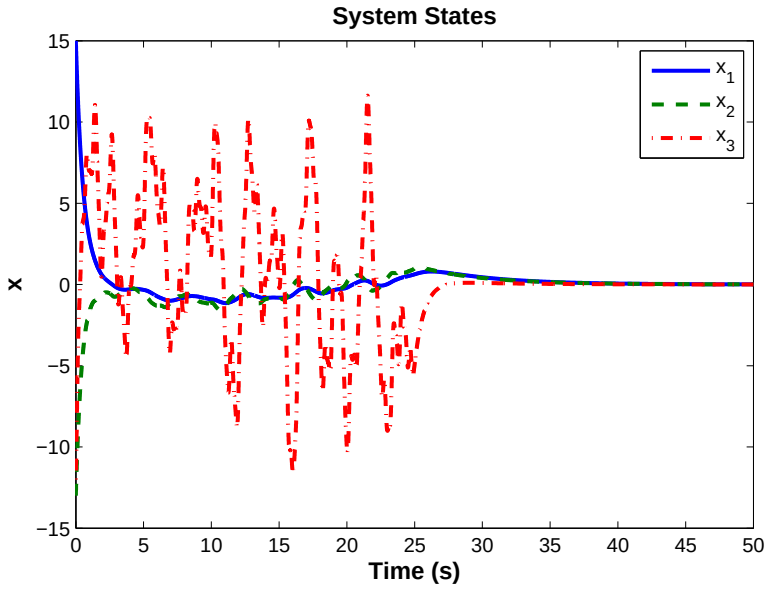

Figure 5-9. System states $x(t)$ with persistently excited input for the first 25 seconds.

The following solution to the ARE can be obtained

$$P = \begin{bmatrix} 1.4245 & 1.1682 & -0.1352 \\ 1.1682 & 1.4349 & -0.1501 \\ -0.1352 & -0.1501 & 2.4329 \end{bmatrix}.$$

The optimal value function is given by $V^*(x) = x^T P x$, and the optimal control is given by

$$u^* = -R^{-1}B^T P x = -\begin{bmatrix} -0.1352 & -0.1501 & 2.4329 \end{bmatrix} x.$$

$$\eta_{a1} = 5, \quad \eta_{a2} = 50, \quad \eta_c = 20, \quad \nu = 0.001.$$

The above LQR design assumes complete knowledge of system dynamics (i.e. A and B), and the ARE is solved offline to obtain P. The proposed actor-critic-identifier architecture is used to solve the LQR problem online without requiring knowledge of the system drift dynamics (i.e. A). The basis for the critic NN is selected by exploiting the

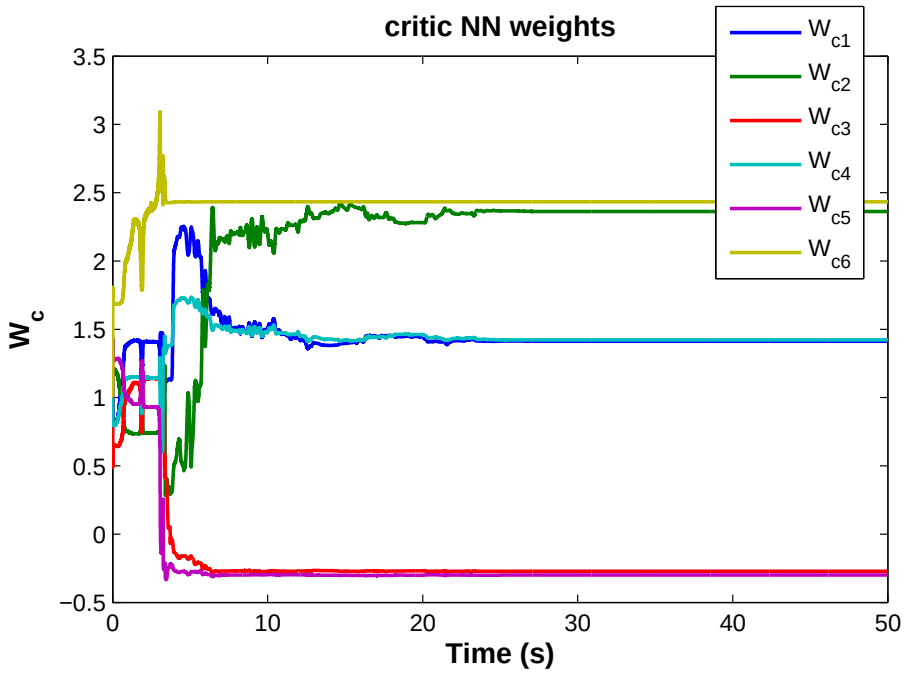

Figure 5-10. Convergence of critic weights $\hat{W}_c(t)$.

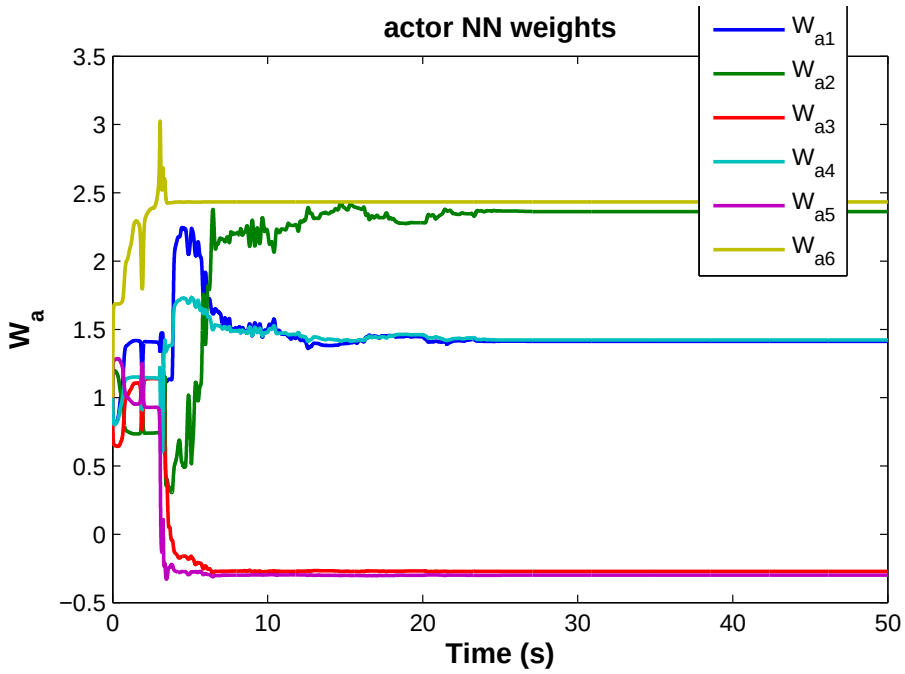

Figure 5-11. Convergence of actor weights $\hat{W}_a(t)$.

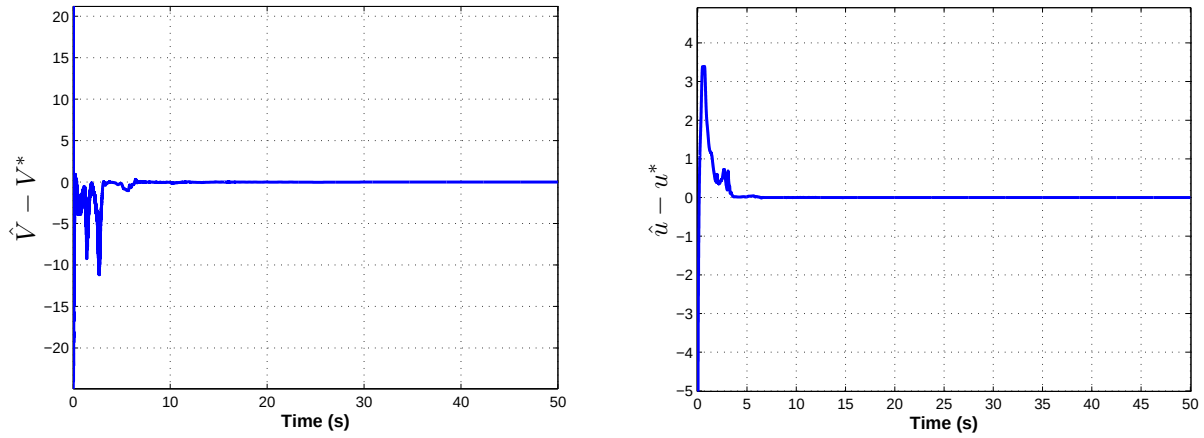

Figure 5-12. Errors in approximating the (a) optimal value function, and (b) optimal control, as a function of time.

structure of the value function as

$$\phi(x) = \begin{bmatrix} x_1^2 & x_1x_2 & x_1x_3 & x_2^2 & x_2x_3 & x_3^2 \end{bmatrix}^T,$$

and the optimal weights are given by

$$W = \begin{bmatrix} 1.4245 & 2.3364 & -0.2704 & 1.4349 & -0.3002 & 2.4329 \end{bmatrix}^T.$$

The same identifier as in the nonlinear example in Section 5.6.1 is used, and the gains for the actor and critic learning laws are selected as The covariance matrix is initialized to $\Gamma(0) = 50000$, all the NN weights are randomly initialized in $[-1,\ 1]$, and the states are initialized to $x(0) = \begin{bmatrix} 15 & -13 & -12 \end{bmatrix}$. To ensure PE, an exploratory signal consisting of sinusoids of varying frequencies, $n(t) = 10(sin(2\pi t) + sin(et) + cos(5t)^5 + sin(10t) + cos(3t) + sin(2t)^2cos(0.1t) + sin(0.5\pi t) + cos(10t) + sin(20t))$, is added to the control $u(t)$ for the first 25 seconds. The evolution of states is shown in Fig. 5-9, and Figs. 5-10 and 5-11 show the convergence of critic and actor weights, respectively. Fig. 5-12 shows the error between the optimal value function and approximate optimal value function, and the optimal control and approximate optimal control, as a function of time along the trajectory $x(t)$.

5.7 Summary

An actor-critic-identifier architecture is proposed to learn the approximate solution to the HJB equation for infinite-horizon optimal control of uncertain nonlinear systems. The online method is the first ever indirect adaptive control approach to continuous-time RL. The learning by the actor, critic and identifier is continuous and simultaneous, and the novel addition of the identifier to the traditional AC architecture eliminates the need to know the system drift dynamics. The actor and critic minimize the Bellman error using gradient and least-squares update laws, respectively, and provide online approximations to the optimal control and the optimal value function, respectively. The identifier estimates the system dynamics online and asymptotically converges to the system state and its derivative. A PE condition is required to ensure exponential convergence to a bounded region in the neighborhood of the optimal control and UUB stability of the closed-loop system. Simulation results demonstrate the performance of the actor-critic-identifier-based method.

CHAPTER 6
CONCLUSION AND FUTURE WORK

This chapter concludes the dissertation by discussing the key ideas developed in each chapter. Limitations and implementation issues of the work are discussed and recommendations are made regarding possible future research directions.

6.1 Dissertation Summary

This work focusses on replicating the success of RL methods in machine learning to control continuous-time nonlinear systems. While in Chapter 3, the RL approach is used to develop robust adaptive controllers which guarantee asymptotic tracking, RL methods are used in Chapter 5 to develop online adaptive optimal controllers. The improvement in performance of the closed-loop system demonstrated through simulations and experiments shows the potential of online data-driven RL methods, where the controller is able to learn the optimal policy by interacting with the environment. The RL approach for optimal control is cast as a parameter estimation and identification problem, and is considered in an adaptive control framework. The adaptive control framework allows rigorous analysis of stability and convergence of the algorithm. For the RL-based optimal control in Chapter 5, a persistence of excitation condition is found to be crucial in ensuring exponential convergence of the parameters to a bounded region in the neighborhood of the optimal control and yields UUB stability of the closed-loop system.

The focus of Chapter 3 is to develop a non-dynamic programming based adaptive critic controller for a class of continuous-time uncertain nonlinear systems with additive bounded disturbances. This work overcomes the limitation of previous work where adaptive critic controllers are either discrete-time and/or yield a uniformly ultimately bounded stability result due to the presence of disturbances and unknown approximation errors. The asymptotic tracking result is made possible by combining a continuous RISE feedback with both the actor and the critic NN structures. The feedforward actor NN approximates the nonlinear system dynamics while the robust feedback (RISE) rejects

the NN functional reconstruction error and disturbances. In addition, the actor NN is trained online using a combination of tracking error, and a reinforcement signal, generated by the critic. Experimental results and t-test analysis demonstrate faster convergence of the tracking error when a reinforcement learning term is included in the NN weight update laws. Although the proposed method guarantees asymptotic tracking, a limitation of the controller is that it does not ensure optimality, which is a common feature (at least approximate optimal control) of DP-based RL controllers.

The development of the state derivative estimator in Chapter 4 is motivated by the need to develop model-free RL-based solutions to the optimal control problem for nonlinear systems. In contrast to purely robust feedback methods in literature, an identification-based robust adaptive approach is developed. The result differs from existing pure robust methods in that the proposed method combines a DNN system identifier with a robust RISE feedback to ensure asymptotic convergence to the state derivative, which is proven using a Lyapunov-based stability analysis. Simulation results in the presence of noise show an improved transient and steady state performance of the developed state derivative identifier in comparison to several other derivative estimation methods. Initially developed for model-free RL-based control, the developed estimator can be used in a wide range of applications, e.g., parameter estimation, fault detection, acceleration feedback, output feedback control, etc.

Due to the difficulty in solving the HJB for optimal control of continuous-time systems, few results exist which solve/circumvent the problem in an online model-free way. The state derivative estimator developed in Chapter 4 paved the way for the development of a novel actor-critic-identifier architecture in Chapter 5 which learns the approximate optimal solution for infinite-horizon optimal control of uncertain nonlinear systems. The method is online, partially model-free, and is the first ever indirect adaptive control approach to continuous-time RL. The actor and critic minimize the Bellman error using gradient and least-squares update laws, respectively, and provide online approximations to

the optimal control and the optimal value function, respectively. The identifier estimates the system dynamics online and asymptotically converges to the system state and its derivative. Another contribution of the result is that the learning by the actor, critic and identifier is continuous and simultaneous, and the novel addition of the identifier to the traditional actor-critic architecture eliminates the need to know the system drift dynamics. A limitation of the method, however, is the requirement of the knowledge of the input gain matrix.

6.2 Future Work

This work illustrates that RL methods can be successfully applied to feedback control. While the methods developed are fairly general and applicable to a wide range of systems, research in this area is still at a nascent stage and several interesting open problems exist. This section discusses the open theoretical problems, implementation issues, and future research directions.

6.2.1 Model-Free RL

RL methods based on TD learning typically do not need a model to learn the optimal policy; they either learn the model online (indirect adaptive approach) or directly learn the parameters of the optimal control (direct adaptive approach). The controller developed in Chapter 5 is based on an indirect adaptive approach, where an identifier is used to approximate the system dynamics online resulting in a model-free formulation of the Bellman error which is used to approximate the value function. Although the approximation of the value function is model-free, the greedy policy used to compute the optimal policy requires knowledge of the input gain matrix. Hence, the developed approach is only partially model-free. A possible approach for completely model-free RL for continuous-time nonlinear systems is to use Q-learning methods [20], a direct adaptive model-free approach to learn optimal policies in MDPs. However, Q-learning-based control design still remains an open problem for continuous-time nonlinear systems. A recent result in [112] points to a possible approach to solve the problem.

6.2.2 Relaxing the Persistence of Excitation Condition

The critic regressor in Chapter 5 is required to satisfy the PE condition for convergence to a neighborhood of the optimal control. As observed in Chapter 5, the PE condition in adaptive control is equivalent to the exploration paradigm, which lies at the heart of RL. Exploration is essential to explore the state space and converge to the global optimal solution. For linear systems, the PE condition translates to the sufficient richness of the external input. However, PE is hard to verify in general for nonlinear systems. Future efforts can focus on relaxing the PE assumption by replacing it with a milder condition on the regressor vector. A recent result in [113] attempts to relax the PE assumption by exploiting prior information about the system but that may go against the spirit of RL which relies on online learning.

6.2.3 Asymptotic RL-Based Optimal Control

Although asymptotic tracking is guaranteed in Chapter 3, the controller is not optimal. In Chapter 5, where an optimal controller is developed, a UUB stability result is achieved. An open problem in RL-based optimal control is asymptotic stability of the closed-loop system in presence of NN approximation errors. One way is to account for approximation errors by combining the optimal control with a robust feedback, e.g., sliding mode or RISE. Although asymptotic stability can be proved by the addition of these robust methods, optimality of the overall controller may be compromised in doing so. Hence, it is not straightforward to extend the robust feedback control tools to optimal control in presence of NN approximation errors.

6.2.4 Better Function Approximation Methods

Generalization and the use of appropriate function approximators for value function approximation is one of the most important issues facing RL, preventing its use in large-scale systems. Function approximation was introduced in RL to alleviate the curse of dimensionality when solving sequential decision problems with large or continuous state spaces [35]. Most RL algorithms for continuous-time control involve parameterization

of the value function and the control. These parameterizations involve selecting an appropriate basis function for the value and the control, a task which can be very hard without any prior knowledge about the system. Linear function approximators, though convenient to use from analysis point of view, have limited approximation capability. Nonlinear approximators like multilayer NNs have better approximation capability but are not amenable for analysis and proving convergence. A challenge for the computational intelligence community is to develop simple yet powerful approximators which are also amenable to mathematical analysis.

6.2.5 Robustness to Disturbances

In practical systems, disturbances are inevitable, e.g., wind gust pushing against an aircraft, contaminant in a chemical process, sudden political upheaval affecting the stock market, etc. The system considered in Chapter 5 is not subjected to any external disturbances, and hence, robustness to external disturbances is not guaranteed. Optimal control of systems subjected to disturbances can be considered in the framework of minimax differential games [114], where the control and disturbance are treated as players with conflicting interests – one minimizes the objective function whereas the other maximizes it, and both reach an optimal compromise (if it exists) which is called the saddle point solution. Recent results in [115, 116] have made inroads into the continuous-time differential game problem. The ACI method developed in Chapter 5 can be extended to solve the differential game problem in an online, partially model-free way.

6.2.6 Output Feedback RL Control

The methods developed in this work assume full state feedback, however, there may be situations where all the states are not available for measurement. In RL jargon, such situations are referred as Partially Observable Markov Decision Processes (POMDPs) [117]. From a controls perspective, in absence of full-state feedback, the problem can be dealt by developing observers and output feedback controllers. An open problem is to extend these methods to RL-based control. A challenge in extending the observer-based

techniques for output feedback RL is that observers typically need a model of the system while RL methods are ideally model-free. A possible alternative is to use non-model based observers, like high gain or sliding mode. The state derivative estimator developed in Chapter 4 can also be extended to the output feedback case.

6.2.7 Extending RL beyond the Infinite-Horizon Regulator

The methods developed in this work are applicable only for infinite-horizon regulation of continuous-time systems. Also, the system considered in 5 is restricted to be autonomous. ADP for time-varying systems and tracking are interesting open problems. Other extensions where future research efforts can be directed are: minimum time, finite-time, and constrained optimal control problems.

APPENDIX A
ASYMPTOTIC TRACKING BY A REINFORCEMENT LEARNING-BASED ADAPTIVE CRITIC CONTROLLER

A.1 Derivation of Sufficient Conditions in Eq. 3–42

Integrating Eq. 3–46, the following expression is obtained

$$\int_0^t L(\tau)d\tau = \int_0^t \left\{ r^T(N_d + N_{B1} - \beta_1 sgn(e_n)) + \dot{e}_n^T N_{B2} - \beta_3 \|e_n\|^2 - \beta_4 |R|^2 \right\} d\tau.$$

Using Eq. 3–4, integrating the first integral by parts, and integrating the second integral yields

$$\begin{aligned}\int_0^t L(\tau)d\tau &= e_n^T N - e_n^T(0)\,N(0) - \int_0^t e_n^T \left(\dot{N}_B + \dot{N}_d\right) d\tau + \beta_1 \sum_{i=1}^m |e_{ni}(0)| - \beta_1 \sum_{i=1}^m |e_{ni}(t)| \\ &\quad + \int_0^t \alpha_n e_n^T (N_d + N_{B1} - \beta_1 sgn(e_n)) d\tau - \int_0^t (\beta_3 \|e_n\|^2 + \beta_4 |R|^2) d\tau.\end{aligned}$$

Using the fact that $\|e_n\| \leq \sum_{i=1}^m |e_{ni}|$, and using the bounds in Eqs. 3–32 and 3–33, yields

$$\begin{aligned}\int_0^t L(\tau)d\tau &\leq \beta_1 \sum_{i=1}^m |e_{ni}(0)| - e_n^T(0)\,N(0) - (\beta_1 - \zeta_1 - \zeta_2 - \zeta_3)\|e_n\| \\ &\quad - \int_0^t \left(\beta_3 - \zeta_7 - \frac{\zeta_8}{2}\right) \|e_n\|^2 d\tau - \int_0^t \left(\beta_4 - \frac{\zeta_8}{2}\right) |R|^2 d\tau \\ &\quad + \int_0^t \alpha_n \|e_n\| \left(\zeta_1 + \zeta_2 + \frac{\zeta_5}{\alpha_n} + \frac{\zeta_6}{\alpha_n} - \beta_1\right) d\tau.\end{aligned}$$

If the sufficient conditions in Eq. 3–42 are satisfied, then the following inequality holds

$$\begin{aligned}\int_0^t L(\tau)d\tau &\leq \beta_1 \sum_{i=1}^m |e_{ni}(0)| - e_n(0)^T N(0). \\ \int_0^t L(\tau)d\tau &\leq P(0). \end{aligned} \tag{A–1}$$

Using Eqs. A–1 and 3–45, it can be shown that $P(z, R, t) \geq 0$.

A.2 Differential Inclusions and Generalized Solutions

Consider a system

$$\dot{x} = f(x,t), \tag{A–2}$$

where $x \in \mathbb{R}^n$ and $f : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}^n$. If the function f is Lipschitz continuous in x and piecewise continuous in t, existence and uniqueness of solutions can be studied and proved in the classical sense (Cauchy-Peano theorem). However, many practical systems with discontinuous right hand sides exist, e.g., Coulomb friction, sliding mode, contact transition etc. For such systems, there may be no solutions in the usual sense, and the notion of solutions has to be generalized to ensure its existence. One of the ways[1] is to study the *generalized solutions* in Filippov's sense using the following differential inclusion

$$\dot{x} \in K[f](x,t), \tag{A–3}$$

where f is Lebesgue measurable and locally bounded, and $K[\cdot]$ is defined as

$$K[f](x,t) \triangleq \bigcap_{\delta>0} \bigcap_{\mu M=0} \overline{co} f(B(x,\delta) - M, t),$$

where $\bigcap_{\mu M=0}$ denotes the intersection of all sets M of Lebesgue measure zero, $\overline{co}$ denotes convex closure. In words, $K[\cdot]$ is the convex closure of the set of all possible limit values of f in small neighborhoods of a given point x. If x is absolutely continuous (i.e. differentiable a.e.) and satisfies Eq. A–3, then it is called a generalized solution (in Filippov's sense) of the differential equation Eq. A–2.

The differential equations of the closed-loop system, Eqs. 3–3, 3–4, 3–20, 3–23, 3–36, 3–38, and 3–45, have discontinuous right hand sides. Specifically, they are continuous except in the set $\{(y,t)|\tilde{x} = 0\}$, which has a Lebesgue measure of 0. Hence, the Filippov's differential inclusion framework is used to ensure existence and unqueness of solutions

[1] If the function f is discontinuous in t and continuous in x, the solution to Eq. A–2 can be studied in the sense of Caratheodory.

(a.e.) for $\dot{y} = F(y,t)$, where F denotes the right hand sides of the differential equations of the closed-loop system. The function F is Lebesgue measurable and locally bounded, and is continuous except in the set $\{(y,t)|\tilde{x} = 0\}$. Stability of solutions based on differential inclusion is studied using non-smooth Lyapunov functions, using the development in [79, 80]. The generalized time derivative of Eq. 3–47 exists almost everywhere (a.e.), and $\dot{V}(y) \in^{a.e.} \dot{\tilde{V}}(y)$ where

$$\dot{\tilde{V}} = \bigcap_{\xi \in \partial V(y)} \xi^T K\left[F\right](y,t), \tag{A–4}$$

where ∂V is the generalized gradient of V [78]. Since the Lyapunov function in Eq. 3–47 is a Lipschitz continuous regular function, the generalized time derivative in Eq. A–4 can be computed as

$$\dot{\tilde{V}} \;=\; \nabla V^T K\left[F\right](y,t).$$

The following relations from [80] are then used to arrive at equation Eq. 3–50:

1. If f and g are locally bounded functions, $K[f+g](x) \subset K[f](x) + K[g](x)$.
2. If $g : \mathbb{R}^m \to \mathbb{R}^{p\times n}$ is C^0 and $f : \mathbb{R}^m \to \mathbb{R}^n$ is locally bounded, $K[gf](x) = g(x)K[f](x)$.
3. If $f : \mathbb{R}^m \to \mathbb{R}^n$ is continuous, $K[f](x) = \{f(x)\}$.
4. $K[sgn(x)] = SGN(x)$, where $SGN(\cdot)$ refers to the set-valued $sgn(\cdot)$ function.

APPENDIX B
ROBUST IDENTIFICATION-BASED STATE DERIVATIVE ESTIMATION FOR NONLINEAR SYSTEMS

B.1 Proof of Inequalities in Eqs. 4–12-4–14

Some preliminary inequalities are proved which will facilitate the proof of inequalities in Eqs. 4–12-4–14. Using the triangle inequality in Eq. 4–2, the following bound can be obtained

$$\begin{aligned} \|\dot{x}\| &\leq \|W_f\| \|\sigma_f\| + \|\varepsilon_f\| + \sum_{i=1}^{m} \left[\|W_{gi}\| \|\sigma_{gi}\| + \|\varepsilon_{gi}\|\right] \|u_i\| + \|d\|, \\ &\leq c_1, \end{aligned} \tag{B–1}$$

where Assumptions 4.2, 4.3, 4.5-4.7 are used and $c_1 \in \mathbb{R}$ is a computable constant. Using triangle inequality in Eq. 4–3, and the fact that $\dot{\tilde{x}} = \dot{x} - r + \alpha\tilde{x}$, the following bound can be obtained

$$\begin{aligned} \left\|\dot{\tilde{x}}\right\| &\leq \|\dot{x}\| + \|r\| + \alpha \|\tilde{x}\|, \\ &\leq c_1 + c_2 \|z\|, \end{aligned} \tag{B–2}$$

where $c_2 \triangleq max\{1,\ \alpha\} \in \mathbb{R}$. Using Assumptions 4.2, 4.6, projection bounds on the weight estimates in Eq. 4–7, and the bounds in Eqs. B–1 and B–2, the following bounds can be developed for the DNN weight update laws in Eq. 4–7

$$\begin{aligned} \left\|\dot{\hat{W}}_f\right\| &\leq c_3 \|\tilde{x}\| + c_4 \|\tilde{x}\| \|z\|, \quad \left\|\dot{\hat{V}}_f\right\| \leq c_5 \|\tilde{x}\| + c_6 \|\tilde{x}\| \|z\|, \\ \left\|\dot{\hat{W}}_{gi}\right\| &\leq c_7 \|\tilde{x}\| + c_8 \|\tilde{x}\| \|z\|, \quad \left\|\dot{\hat{V}}_{gi}\right\| \leq c_9 \|\tilde{x}\| + c_{10} \|\tilde{x}\| \|z\| \quad \forall i = 1...m, \end{aligned} \tag{B–3}$$

where $c_i \in \mathbb{R}\ i = 3...10$ are computable constants. Using Assumptions 4.1-4.3, the derivative of the dynamics in Eq. 4–2 yields

$$\ddot{x} = W_f^T \sigma_f' V_f^T \dot{x} + \varepsilon_f' \dot{x} + \sum_{i=1}^{m} \left(\left[W_{gi}^T \sigma_{gi}' V_{gi}^T \dot{x} + \varepsilon_{gi}' \dot{x}\right] u_i + \left[W_{gi}^T \sigma_{gi} + \varepsilon_{gi}\right] \dot{u}_i \right) + \dot{d}, \tag{B–4}$$

and using the triangle inequality yields the following bound

$$\begin{aligned}\|\ddot{x}\| &\leq \|W_f\| \left\|\sigma_f'\right\| \|V_f\| \|\dot{x}\| + \left\|\varepsilon_f'\right\| \|\dot{x}\| + \sum_{i=1}^{m} \left(\left[\|W_{gi}\| \left\|\sigma_{gi}'\right\| \|V_{gi}\| \|\dot{x}\| + \left\|\varepsilon_{gi}'\right\| \|\dot{x}\| \right] \|u_i\| \right. \\ &\quad \left. + \left[\|W_{gi}\| \|\sigma_{gi}\| + \|\varepsilon_{gi}\| \right] \|\dot{u}_i\| \right) + \left\|\dot{d}\right\|, \\ &\leq c_{11}, \end{aligned} \tag{B–5}$$

where Assumptions 4.2, 4.3, 4.5-4.7, and Eq. B–1 is used, and $c_{11} \in \mathbb{R}$ is a computable constant.

B.1.1 Proof of Inequality in Eq. 4–12

Using triangle inequality in Eq. 4–9 yields

$$\begin{aligned}\left\|\tilde{N}\right\| &\leq \alpha \left\|\dot{\tilde{x}}\right\| + \left\|\dot{\hat{W}}_f\right\| \|\hat{\sigma}_f\| + \left\|\hat{W}_f\right\| \left\|\hat{\sigma}_f'\right\| \left\|\dot{\hat{V}}_f\right\| \|\hat{x}\| + \frac{1}{2} \left\|W_f^T\right\| \left\|\hat{\sigma}_f'\right\| \left\|\hat{V}_f\right\| \left\|\dot{\tilde{x}}\right\| \\ &\quad + \frac{1}{2} \left\|\hat{W}_f\right\| \left\|\hat{\sigma}_f'\right\| \|V_f\| \left\|\dot{\tilde{x}}\right\| + \sum_{i=1}^{m} \left[\left\|\dot{\hat{W}}_{gi}\right\| \|\hat{\sigma}_{gi}\| \|u_i\| + \left\|\hat{W}_{gi}\right\| \left\|\hat{\sigma}_{gi}'\right\| \left\|\dot{\hat{V}}_{gi}\right\| \|\hat{x}\| \|u_i\| \right. \\ &\quad \left. + \frac{1}{2} \left\|\hat{W}_{gi}\right\| \left\|\hat{\sigma}_{gi}'\right\| \|V_{gi}\| \left\|\dot{\tilde{x}}\right\| \|u_i\| + \frac{1}{2} \left\|W_{gi}^T\right\| \left\|\hat{\sigma}_{gi}'\right\| \left\|\hat{V}_{gi}\right\| \left\|\dot{\tilde{x}}\right\| \|u_i\| \right]. \end{aligned} \tag{B–6}$$

Using Eq. 4–5, the fact that $\|x\|, \|r\| \leq \|z\|$, and the bounds developed in Eqs. B–1, B–2, and B–3, the expression in Eq. B–6 can be further upper bounded as

$$\begin{aligned}\left\|\tilde{N}\right\| &\leq \left[\|\Gamma_{wf}\| \left\|\hat{\sigma}_f'\right\| \left\|\hat{V}_f\right\| \|\hat{\sigma}_f\| (c_3 + c_4 \|z\|) + \left\|\hat{W}_f\right\| \left\|\hat{\sigma}_f'\right\| (\|x\| + \|\tilde{x}\|)(c_5 + c_6 \|z\|) \right. \\ &\quad \left. + \alpha + \alpha^2 + \frac{1}{2} c_2 \left\|W_f^T\right\| \left\|\hat{\sigma}_f'\right\| \left\|\hat{V}_f\right\| + \frac{1}{2} c_2 \left\|\hat{W}_f\right\| \left\|\hat{\sigma}_f'\right\| \|V_f\| \right] \|z\| \\ &\quad + \left[\sum_{i=1}^{m} \left\{ \|\hat{\sigma}_{gi}\| \|u_i\| (c_7 + c_8 \|z\|) + c_2 \left\|\hat{W}_{gi}\right\| \left\|\hat{\sigma}_{gi}'\right\| \|V_{gi}\| \|u_i\| \right\} \right] \|z\| \\ &\quad + \frac{1}{2} \left[\sum_{i=1}^{m} \left\|\hat{W}_{gi}\right\| \left\|\hat{\sigma}_{gi}'\right\| \|u_i\| (\|x\| + \|\tilde{x}\|)(c_9 + c_{10} \|z\|) \right] \|z\| \\ &\quad + \frac{1}{2} \left[\sum_{i=1}^{m} \frac{1}{2} c_2 \left\|W_{gi}^T\right\| \left\|\hat{\sigma}_{gi}'\right\| \left\|\hat{V}_{gi}\right\| \|u_i\| \right] \|z\| \\ &\leq \rho_1(\|z\|) \|z\|, \end{aligned}$$

where $\rho_1(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function.

B.1.2 Proof of Inequalities in Eq. 4–13

Using the triangle inequality in Eq. 4–10 yields

$$\begin{aligned}\|N_{B1}\| \leq & \sum_{i=1}^{m}\left[\|W_{gi}\|\,\|\sigma_{gi}\|\,\|\dot{u}_i\| + c_1\|W_{gi}\|\,\left\|\sigma'_{gi}\right\|\,\|V_{gi}\|\,\|u_i\| + \|\dot{\varepsilon}_{gi}\|\,\|u_i\| + \|\varepsilon_{gi}\|\,\|\dot{u}_i\|\right] \\ & + c_1\|W_f\|\,\left\|\sigma'_f\right\|\,\|V_f\| + \|\dot{\varepsilon}_f\| + \left\|\dot{d}\right\| + + \frac{1}{2}c_1\|W_f\|\,\left\|\hat{\sigma}'_f\right\|\,\left\|\hat{V}_f\right\| \\ & + \sum_{i=1}^{m}\left[\frac{1}{2}\left\|\hat{W}_{gi}\right\|\,\left\|\hat{\sigma}'_{gi}\right\|\,\|V_{gi}\|\,\|u_i\| + \frac{1}{2}\|W_{gi}\|\,\left\|\hat{\sigma}'_{gi}\right\|\,\left\|\hat{V}_{gi}\right\|\,\|u_i\| + \left\|\hat{W}_{gi}\right\|\,\|\hat{\sigma}_{gi}\|\,\|\dot{u}_i\|\right] \\ & + \frac{1}{2}c_1\left\|\hat{W}_f\right\|\,\left\|\hat{\sigma}'_f\right\|\,\|V_f\| \\ \leq & \ \zeta_1,\end{aligned} \tag{B–7}$$

where Assumptions 4.2, 4.3, 4.5-4.7, and projection bounds on the weight estimates in Eq. 4–7 are used, and $\zeta_1 \in \mathbb{R}$ is a positive constant computed using the upper bounds of the terms in Eq. B–7. By replacing $\dot{\hat{x}}(t)$ by $\dot{x}(t)$ in Eq. 4–11, the expression for $N_{B2}(\cdot)$ can be obtained as

$$N_{B2} \triangleq \sum_{i=1}^{m}\left[\frac{1}{2}\tilde{W}_{gi}^T\hat{\sigma}'_{gi}\hat{V}_{gi}^T\dot{x}u_i + \frac{1}{2}\hat{W}_{gi}^T\hat{\sigma}'_{gi}\tilde{V}_{gi}^T\dot{x}u_i\right] + \frac{1}{2}\tilde{W}_f^T\hat{\sigma}'_f\hat{V}_f^T\dot{x} + \frac{1}{2}\hat{W}_f^T\hat{\sigma}'_f\tilde{V}_f^T\dot{x}. \tag{B–8}$$

Using the triangle inequality in Eq. B–8 yields

$$\begin{aligned}\|N_{B2}\| \leq & \sum_{i=1}^{m}\left[\frac{1}{2}c_1\left\|\tilde{W}_{gi}\right\|\,\left\|\hat{\sigma}'_{gi}\right\|\,\left\|\hat{V}_{gi}\right\|\,\|u_i\| + \frac{1}{2}c_1\left\|\hat{W}_{gi}\right\|\,\left\|\hat{\sigma}'_{gi}\right\|\,\left\|\tilde{V}_{gi}\right\|\,\|u_i\|\right] \\ & + \frac{1}{2}c_1\left\|\tilde{W}_f\right\|\,\left\|\hat{\sigma}'_f\right\|\,\left\|\hat{V}_f\right\| + \frac{1}{2}c_1\left\|\hat{W}_f\right\|\,\left\|\hat{\sigma}'_f\right\|\,\left\|\tilde{V}_f\right\| \\ \leq & \ \zeta_2,\end{aligned} \tag{B–9}$$

where Assumptions 4.2, 4.3, 4.5-4.7 and projection bounds on the weight estimates in Eq. 4–7 are used, and $\zeta_2 \in \mathbb{R}$ is a positive constant computed using the upper bounds of the terms in Eq. B–9. Taking the derivative of $N_B \triangleq N_{B1} + N_{B2}$, and using Eqs. 4–11 and B–8 yields $\dot{N}_B(\cdot)$, which can be split as

$$\dot{N}_B \triangleq \dot{N}_{Ba} + \dot{N}_{Bb}, \tag{B–10}$$

where

$$\begin{aligned}
\dot{N}_{Ba} \triangleq & \sum_{i=1}^{m} \left[W_{gi}^T \sigma_{gi}' V_{gi}^T \dot{x} \dot{u}_i + W_{gi}^T \sigma_{gi} \ddot{u}_i + W_{gi}^T \sigma_{gi}' V_{gi}^T \dot{x} \dot{u}_i + W_{gi}^T \sigma_{gi}' V_{gi}^T \ddot{x} u_i + W_{gi}^T \dot{\sigma}_{gi}' V_{gi}^T \dot{x} u_i \right] \\
& + \sum_{i=1}^{m} \left[\ddot{\varepsilon}_{gi} u_i + 2 \dot{\varepsilon}_{gi} \dot{u}_i + \varepsilon_{gi} \ddot{u}_i - \frac{1}{2} \hat{W}_{gi}^T \hat{\sigma}_{gi}' V_{gi}^T \ddot{x} u_i - \frac{1}{2} \hat{W}_{gi}^T \hat{\sigma}_{gi}' V_{gi}^T \dot{x} \dot{u}_i - \frac{1}{2} W_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \ddot{x} u_i \right] \\
& + \sum_{i=1}^{m} \left[-\frac{1}{2} W_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{x} \dot{u}_i - \hat{W}_{gi}^T \hat{\sigma}_{gi} \ddot{u}_i + \frac{1}{2} \tilde{W}_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \ddot{x} u_i + \frac{1}{2} \tilde{W}_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{x} \ddot{u}_i \right] \qquad \text{(B–11)} \\
& + \ddot{\varepsilon}_f + \ddot{d} + W_f^T \dot{\sigma}_f' V_f^T \dot{x} + W_f^T \sigma_f' V_f^T \ddot{x} - \frac{1}{2} W_f^T \hat{\sigma}_f' \hat{V}_f^T \ddot{x} - \frac{1}{2} \hat{W}_f^T \hat{\sigma}_f' V_f^T \ddot{x} \\
& + \frac{1}{2} \tilde{W}_f^T \hat{\sigma}_f' \hat{V}_f^T \ddot{x} + \frac{1}{2} \hat{W}_f^T \hat{\sigma}_f' \tilde{V}_f^T \ddot{x},
\end{aligned}$$

$$\begin{aligned}
\dot{N}_{Bb} \triangleq & - \sum_{i=1}^{m} \left[\frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' V_{gi}^T \dot{x} u_i + \frac{1}{2} \hat{W}_{gi}^T \dot{\hat{\sigma}}_{gi}' V_{gi}^T \dot{x} u_i + \frac{1}{2} W_{gi}^T \dot{\hat{\sigma}}_{gi}' \hat{V}_{gi}^T \dot{x} u_i + \frac{1}{2} W_{gi}^T \hat{\sigma}_{gi}' \dot{\hat{V}}_{gi}^T \dot{x} u_i \right] \\
& - \frac{1}{2} W_f^T \hat{\sigma}_f' \dot{\hat{V}}_f^T \dot{x} - \frac{1}{2} \dot{\hat{W}}_f^T \hat{\sigma}_f' V_f^T \dot{x} - \frac{1}{2} \hat{W}_f^T \dot{\hat{\sigma}}_f' V_f^T \dot{x} + \frac{1}{2} \dot{\hat{W}}_f^T \hat{\sigma}_f' \hat{V}_f^T \dot{x} + \frac{1}{2} \tilde{W}_f^T \dot{\hat{\sigma}}_f' \hat{V}_f^T \dot{x} \\
& + \sum_{i=1}^{m} \left[-\dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi} \dot{u}_i - \hat{W}_{gi}^T \dot{\hat{\sigma}}_{gi} \dot{u}_i - \frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' \hat{V}_{gi}^T \dot{x} u_i + \frac{1}{2} \tilde{W}_{gi}^T \dot{\hat{\sigma}}_{gi}' \hat{V}_{gi}^T \dot{x} u_i + \frac{1}{2} \tilde{W}_{gi}^T \hat{\sigma}_{gi}' \dot{\hat{V}}_{gi}^T \dot{x} u_i \right] \\
& + \sum_{i=1}^{m} \left[\frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' \tilde{V}_{gi}^T \dot{x} u_i + \frac{1}{2} \dot{\hat{W}}_{gi}^T \dot{\hat{\sigma}}_{gi}' \tilde{V}_{gi}^T \dot{x} u_i - \frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' \dot{\hat{V}}_{gi}^T \dot{x} u_i + \frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' \tilde{V}_{gi}^T \ddot{x} u_i \right. \qquad \text{(B–12)} \\
& \left. + \frac{1}{2} \dot{\hat{W}}_{gi}^T \hat{\sigma}_{gi}' \tilde{V}_{gi}^T \dot{x} \dot{u}_i \right] + \frac{1}{2} \tilde{W}_f^T \hat{\sigma}_f' \dot{\hat{V}}_f^T \dot{x} + \frac{1}{2} \dot{\hat{W}}_f^T \hat{\sigma}_f' \tilde{V}_f^T \dot{x} + \frac{1}{2} \hat{W}_f^T \dot{\hat{\sigma}}_f' \tilde{V}_f^T \dot{x} - \frac{1}{2} \hat{W}_f^T \hat{\sigma}_f' \dot{\hat{V}}_f^T \dot{x},
\end{aligned}$$

where $\dot{\hat{\sigma}}_{gi}'$ denotes the time derivative of $\hat{\sigma}_{gi}'$. To develop upper bounds for Eqs. B–11 and B–12, the following bound will be used

$$\begin{aligned}
\left\| \dot{\hat{\sigma}}_{gi}' \right\| & \leq \left\| \hat{\sigma}_{gi}'' \right\| \left\| \dot{\hat{V}}_{gi} \right\| \|\hat{x}\| + \left\| \hat{\sigma}_{gi}'' \right\| \left\| \hat{V}_{gi} \right\| \left\| \dot{\hat{x}} \right\| \\
& \leq \left(c_9 \|\tilde{x}\| + c_{10} \|\tilde{x}\| \|z\| \right) \left\| \hat{\sigma}_{gi}'' \right\| \left(\|x\| + \|\tilde{x}\| \right) + \left(c_1 + c_2 \|z\| \right) \left\| \hat{\sigma}_{gi}'' \right\| \left\| \hat{V}_{gi} \right\| \\
& \leq c_{12} + \rho_0(\|z\|) \|z\|,
\end{aligned}$$

where Eqs. B–2 and B–3 are used, $c_{12} \in \mathbb{R}^+$, and $\rho_0(\cdot) \in \mathbb{R}$ is a positive, globally invertible, non-decreasing function. Using the bound in Eq. B–5, $\left\| \dot{N}_{Ba} \right\|$ in Eq. B–11 can

be upper bounded using the triangle inequality as

$$\begin{aligned}
\left\|\dot{N}_{Ba}\right\| \leq & \sum_{i=1}^{m}\left[c_1\left\|W_{gi}\right\|\left\|\sigma'_{gi}\right\|\left\|V_{gi}\right\|\left\|\dot{u}_i\right\|+\left\|W_{gi}\right\|\left\|\sigma_{gi}\right\|\left\|\ddot{u}_i\right\|+c_1\left\|W_{gi}\right\|\left\|\sigma'_{gi}\right\|\left\|V_{gi}\right\|\left\|\dot{u}_i\right\|\right] \\
& +\sum_{i=1}^{m}\left[c_{11}\left\|W_{gi}\right\|\left\|\sigma'_{gi}\right\|\left\|V_{gi}\right\|\left\|u_i\right\|+c_1\left\|W_{gi}\right\|\left\|\dot{\sigma}'_{gi}\right\|\left\|V_{gi}\right\|\left\|u_i\right\|+\left\|\ddot{\varepsilon}_{gi}\right\|\left\|u_i\right\|\right] \\
& +\sum_{i=1}^{m}\left[2\left\|\dot{\varepsilon}_{gi}\right\|\left\|\dot{u}_i\right\|\left\|\dot{\varepsilon}_{gi}\right\|\left\|\ddot{u}_i\right\|+\frac{1}{2}c_{11}\left\|\hat{W}_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|V_{gi}\right\|\left\|u_i\right\|\right] \\
& +\sum_{i=1}^{m}\left[\frac{1}{2}c_1\left\|\hat{W}_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|V_{gi}\right\|\left\|\dot{u}_i\right\|+\frac{1}{2}c_{11}\left\|W_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|\hat{V}_{gi}\right\|\left\|u_i\right\|\right] \\
& +\left\|\ddot{\varepsilon}_f\right\|+\left\|\ddot{d}\right\|+c_1\left\|W_f\right\|\left\|\dot{\sigma}'_f\right\|\left\|V_f\right\|+c_{11}\left\|W_f\right\|\left\|\sigma'_f\right\|\left\|V_f\right\|+\frac{1}{2}c_{11}\left\|W_f\right\|\left\|\hat{\sigma}'_f\right\|\left\|\hat{V}_f\right\| \\
& +\frac{1}{2}c_{11}\left\|\hat{W}_f\right\|\left\|\hat{\sigma}'_f\right\|\left\|V_f\right\|+\sum_{i=1}^{m}\left[\frac{1}{2}\left\|W_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|\hat{V}_{gi}\right\|\left\|\dot{x}\right\|\left\|\dot{u}_i\right\|\left\|\hat{W}_{gi}\right\|\left\|\hat{\sigma}_{gi}\right\|\left\|\ddot{u}_i\right\|\right] \\
& +\sum_{i=1}^{m}\frac{1}{2}c_{11}\left\|\tilde{W}_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|\hat{V}_{gi}\right\|\left\|u_i\right\|+\frac{1}{2}\left\|\tilde{W}_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|\hat{V}_{gi}\right\|\left\|\dot{x}\right\|\left\|\ddot{u}_i\right\| \\
& +\frac{1}{2}c_{11}\left\|\tilde{W}_f\right\|\left\|\hat{\sigma}'_f\right\|\left\|\hat{V}_f\right\|+\frac{1}{2}c_{11}\left\|\hat{W}_f\right\|\left\|\hat{\sigma}'_f\right\|\left\|\tilde{V}_f\right\|.
\end{aligned}$$

Using Assumptions 4.2, 4.3, 4.5-4.7, all the terms in the above expression can be bounded by a constant, and hence the following bound can be developed

$$\left\|\dot{N}_{Ba}\right\| \leq \zeta_{31}, \tag{B–13}$$

where $\zeta_{31} \in \mathbb{R}^{+}$is a computable constant. The bound on Eq. B–14 is developed as

$$\begin{aligned}
\left\|\dot{N}_{Bb}\right\| \leq & \sum_{i=1}^{m}\frac{1}{2}c_1(c_7\left\|\tilde{x}\right\|+c_8\left\|\tilde{x}\right\|\left\|z\right\|)\left\|\hat{\sigma}'_{gi}\right\|\left\|V_{gi}\right\|\left\|u_i\right\| \\
& +\sum_{i=1}^{m}\frac{1}{2}c_1(c_{12}+\rho_0(\left\|z\right\|)\left\|z\right\|)\left\|\hat{W}_{gi}\right\|\left\|V_{gi}\right\|\left\|u_i\right\| \\
& +\sum_{i=1}^{m}\frac{1}{2}c_1(c_{12}+\rho_0(\left\|z\right\|)\left\|z\right\|)\left\|W_{gi}\right\|\left\|\hat{V}_{gi}\right\|\left\|u_i\right\| \\
& +\sum_{i=1}^{m}\frac{1}{2}c_1(c_9\left\|\tilde{x}\right\|+c_{10}\left\|\tilde{x}\right\|\left\|z\right\|)\left\|W_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\left\|u_i\right\| \\
& +\frac{1}{2}c_1(c_5\left\|\tilde{x}\right\|+c_6\left\|\tilde{x}\right\|\left\|z\right\|)\left\|W_f\right\|\left\|\hat{\sigma}'_f\right\|+\frac{1}{2}c_1(c_3\left\|\tilde{x}\right\|+c_4\left\|\tilde{x}\right\|\left\|z\right\|)\left\|\hat{\sigma}'_f\right\|\left\|V_f\right\|
\end{aligned}$$

$$
\begin{aligned}
&+\frac{1}{2}c_1(c_{12}+\rho_0(\|z\|)\,\|z\|)\left\|\hat{W}_f\right\|\|V_f\|+\frac{1}{2}c_1(c_3\|\tilde{x}\|+c_4\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_f\right\|\left\|\hat{V}_f\right\| \\
&+\sum_{i=1}^{m}\left[(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}_{gi}\right\|\|\dot{u}_i\|+(c_{12}+\rho_0(\|z\|)\,\|z\|)\left\|\hat{W}_{gi}\right\|\|\dot{u}_i\|\right] \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_{gi}\right\|\left\|\hat{V}_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_{12}+\rho_0(\|z\|)\,\|z\|)\left\|\tilde{W}_{gi}\right\|\left\|\hat{V}_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_9\|\tilde{x}\|+c_{10}\|\tilde{x}\|\,\|z\|)\left\|\tilde{W}_{gi}\right\|\left\|\hat{\sigma}'_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_{gi}\right\|\left\|\tilde{V}_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_{12}+\rho_0(\|z\|)\,\|z\|)(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\tilde{V}_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_{11}(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_{gi}\right\|\left\|\tilde{V}_{gi}\right\|\|u_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_{gi}\right\|\left\|\tilde{V}_{gi}\right\|\|\dot{u}_i\| \\
&+\sum_{i=1}^{m}\frac{1}{2}c_1(c_7\|\tilde{x}\|+c_8\|\tilde{x}\|\,\|z\|)(c_9\|\tilde{x}\|+c_{10}\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_{gi}\right\|\|u_i\| \\
&+\frac{1}{2}c_1(c_{12}+\rho_0(\|z\|)\,\|z\|)\left\|\tilde{W}_f\right\|\left\|\hat{V}_f\right\|+\frac{1}{2}c_1(c_5\|\tilde{x}\|+c_6\|\tilde{x}\|\,\|z\|)\left\|\tilde{W}_f\right\|\left\|\hat{\sigma}'_f\right\| \\
&+\frac{1}{2}c_1(c_3\|\tilde{x}\|+c_4\|\tilde{x}\|\,\|z\|)\left\|\hat{\sigma}'_f\right\|\left\|\tilde{V}_f\right\|+\frac{1}{2}c_1(c_{12}+\rho_0(\|z\|)\,\|z\|)\left\|\hat{W}_f\right\|\left\|\tilde{V}_f\right\| \\
&+\frac{1}{2}c_1(c_5\|\tilde{x}\|+c_6\|\tilde{x}\|\,\|z\|)\left\|\hat{W}_f\right\|\left\|\hat{\sigma}'_f\right\|,
\end{aligned}
$$

which can be simplified by combining terms bounded by constants and terms bounded by a function of states, as

$$
\left\|\dot{N}_{Bb}\right\| \;\leq\; \zeta_{32}+\zeta_4\rho_2(\|z\|)\,\|z\|\,, \tag{B–14}
$$

where $\zeta_{32},\zeta_4\in\mathbb{R}^+$are computable constants, and $\rho_2(\cdot)\in\mathbb{R}$ is a positive, globally invertible, non-decreasing function. From Eqs. B–10, B–13, and B–14, the following bound can be

obtained

$$\left\|\dot{N}_B\right\| \leq \zeta_3 + \zeta_4 \rho_2(\|z\|)\|z\|.$$

B.1.3 Proof of Inequality in Eq. 4–14

Using the definition $\tilde{N}_{B2} \triangleq \hat{N}_{B2} - N_{B2}$

$$\begin{aligned}\dot{\tilde{x}}^T \tilde{N}_{B2} &= \dot{\tilde{x}}^T(\hat{N}_{B2} - N_{B2}) \\ &= \dot{\tilde{x}}^T \sum_{i=1}^{m}\left[\frac{1}{2}\tilde{W}_{gi}^T \hat{\sigma}'_{gi}\hat{V}_{gi}^T \dot{\tilde{x}}u_i + \frac{1}{2}\hat{W}_{gi}^T\hat{\sigma}'_{gi}\tilde{V}_{gi}^T\dot{\tilde{x}}u_i\right] + \frac{1}{2}\dot{\tilde{x}}^T\tilde{W}_f^T\hat{\sigma}'_f\hat{V}_f^T\dot{\tilde{x}} \\ &\quad + \frac{1}{2}\dot{\tilde{x}}^T\hat{W}_f^T\hat{\sigma}'_f\tilde{V}_f^T\dot{\tilde{x}},\end{aligned}$$

which can be upper bounded using the triangle inequality as

$$\begin{aligned}\left\|\dot{\tilde{x}}^T\tilde{N}_{B2}\right\| \leq \frac{1}{2}\|\dot{\tilde{x}}\|^2 \Bigg\{ &\sum_{i=1}^{m}\left[\left\|\tilde{W}_{gi}\right\|\|\hat{\sigma}'_{gi}\|\left\|\hat{V}_{gi}\right\|\|u_i\| + \left\|\hat{W}_{gi}\right\|\|\hat{\sigma}'_{gi}\|\left\|\tilde{V}_{gi}\right\|\|u_i\|\right] \\ &+ \left\|\tilde{W}_f\right\|\|\hat{\sigma}'_f\|\left\|\hat{V}_f\right\| + \left\|\hat{W}_f\right\|\|\hat{\sigma}'_f\|\left\|\tilde{V}_f\right\|\Bigg\}. \end{aligned} \tag{B–15}$$

Using the fact that $\|\dot{\tilde{x}}\|^2 = \|r - \alpha\tilde{x}\|^2 = (r-\alpha\tilde{x})^T(r-\alpha\tilde{x}) \leq \|r\|^2 + \alpha^2\|\tilde{x}\|^2 + 2\alpha\|r\|\|\tilde{x}\| \leq (1+\alpha)\|r\|^2 + \alpha(1+\alpha)\|\tilde{x}\|^2$, Eq. B–15 can be further upper bounded as

$$\begin{aligned}\left\|\dot{\tilde{x}}^T\tilde{N}_{B2}\right\| \leq \frac{1}{2}(1+\alpha)\left[\|r\|^2 + \alpha\|\tilde{x}\|^2\right]\Bigg\{ &\left\|\tilde{W}_f\right\|\|\hat{\sigma}'_f\|\left\|\hat{V}_f\right\| + \left\|\hat{W}_f\right\|\|\hat{\sigma}'_f\|\left\|\tilde{V}_f\right\| \\ &+ \sum_{i=1}^{m}\left[\left\|\tilde{W}_{gi}\right\|\|\hat{\sigma}'_{gi}\|\left\|\hat{V}_{gi}\right\|\|u_i\| + \left\|\hat{W}_{gi}\right\|\|\hat{\sigma}'_{gi}\|\left\|\tilde{V}_{gi}\right\|\|u_i\|\right]\Bigg\}.\end{aligned}$$

Using the Assumptions 4.2, 4.3, 4.5-4.7, the following bound can obtained

$$\left\|\dot{\tilde{x}}^T\tilde{N}_{B2}\right\| \leq \zeta_5\|\tilde{x}\|^2 + \zeta_6\|r\|^2, \tag{B–16}$$

where $\zeta_5, \zeta_6 \in \mathbb{R}^+$ are computable constants.

B.2 Derivation of Sufficient Conditions in Eq. 4–18

Integrating Eq. 4–17 yields

$$\begin{aligned}\int_0^t L(\tau)d\tau &= \int_0^t \left\{r^T\left[N_{B1}(\tau)-\beta_1 sgn(\tilde{x})\right]+\dot{\tilde{x}}(\tau)^T N_{B2}(\tau)-\beta_2\rho_2(\|z\|)\|z\|\,\|\tilde{x}\|\right\}d\tau.\\ &= \tilde{x}^T N_B-\tilde{x}^T(0)N_B(0)-\int_0^t \tilde{x}^T\dot{N}_B d\tau+\beta_1\sum_{i=1}^n|\tilde{x}_i(0)|-\beta_1\sum_{i=1}^n|\tilde{x}_i(t)|\\ &\quad+\int_0^t \alpha\tilde{x}^T(N_{B1}-\beta_1 sgn(\tilde{x}))d\tau-\int_0^t\beta_2\rho_2(\|z\|)\|z\|\,\|\tilde{x}\|\,d\tau,\end{aligned}$$

where Eq. 4–8 is used. Using the fact that $\|\tilde{x}\|_2 \le \sum_{i=1}^n |\tilde{x}_i|$, and using the bounds in Eq. 4–13, yields

$$\begin{aligned}\int_0^t L(\tau)d\tau &\le \beta_1\sum_{i=1}^n|\tilde{x}_i(0)|-\tilde{x}^T(0)N_B(0)-(\beta_1-\zeta_1-\zeta_2)\|\tilde{x}\|\\ &\quad-\int_0^t\alpha(\beta_1-\zeta_1-\frac{\zeta_3}{\alpha})\|\tilde{x}\|\,d\tau-\int_0^t(\beta_2-\zeta_4)\rho_2(\|z\|)\|z\|\,\|\tilde{x}\|\,d\tau.\end{aligned}$$

If the sufficient conditions in Eq. 4–18 are satisfied, then the following inequality holds

$$\int_0^t L(\tau)d\tau \le \beta_1\sum_{i=1}^n|\tilde{x}_i(0)|-\tilde{x}^T(0)N_B(0)=P(0) \tag{B–17}$$

Using Eqs. 4–16 and B–17, it can be shown that $P(z,t)\ge 0$.

REFERENCES

[1] R. Sutton, "Learning to predict by the methods of temporal differences," *Mach. Learn.*, vol. 3, no. 1, pp. 9–44, 1988.

[2] R. Sutton and A. Barto, *Introduction to reinforcement learning*. MIT Press Cambridge, MA, USA, 1998.

[3] R. Sutton, A. Barto, and R. Williams, "Reinforcement learning is direct adaptive optimal control," *IEEE Contr. Syst. Mag.*, vol. 12, no. 2, pp. 19–22, 1992.

[4] B. Widrow, N. Gupta, and S. Maitra, "Punish/reward: Learning with a critic in adaptive threshold systems," *IEEE Trans. Syst. Man Cybern.*, vol. 3, no. 5, pp. 455–465, 1973.

[5] P. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control: Neural, Fuzzy, and Adaptive Approaches*, D. A. White and D. A. Sofge, Eds. New York: Van Nostrand Reinhold, 1992.

[6] D. V. Prokhorov and I. Wunsch, D. C., "Adaptive critic designs," *IEEE Trans. Neural Networks*, vol. 8, pp. 997–1007, 1997.

[7] S. Ferrari and R. Stengel, "An adaptive critic global controller," in *Proc. Am. Control Conf.*, vol. 4, 2002.

[8] J. Murray, C. Cox, G. Lendaris, and R. Saeks, "Adaptive dynamic programming," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 32, no. 2, pp. 140–153, 2002.

[9] X. Liu and S. Balakrishnan, "Convergence analysis of adaptive critic based optimal control," in *Proc. Am. Control Conf.*, vol. 3, 2000.

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

[11] R. Padhi, S. Balakrishnan, and T. Randolph, "Adaptive-critic based optimal neuro control synthesis for distributed parameter systems," *Automatica*, vol. 37, no. 8, pp. 1223–1234, 2001.

[12] T. Hanselmann, L. Noakes, and A. Zaknich, "Continuous-time adaptive critics," *IEEE Trans. Neural Networks*, vol. 18, no. 3, pp. 631–647, 2007.

[13] D. Vrabie and F. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, vol. 22, no. 3, pp. 237 – 246, 2009.

[14] K. Vamvoudakis and F. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, pp. 878–888, 2010.

[15] G. Cybenko, "Approximation by superpositions of a sigmoidal function," *Math. Control Signals Syst.*, vol. 2, pp. 303–314, 1989.

[16] A. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, 1993.

[17] P. Werbos, "A menu of designs for reinforcement learning over time," *Neural networks for control*, pp. 67–95, 1990.

[18] 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.

[19] A. Barto, R. Sutton, and C. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst. Man Cybern.*, vol. 13, no. 5, pp. 834–846, 1983.

[20] C. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279–292, 1992.

[21] P. Werbos, "Building and understanding adaptive systems: A statistical/numerical approach to factory automation and brain research," *IEEE Trans. Syst. Man Cybern.*, vol. 17, no. 1, pp. 7–20, 1987.

[22] R. Bellman, *Dynamic Programming*. Dover Publications, Inc., 2003.

[23] J. Si and Y. Wang, "On-line learning control by association and reinforcement," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 264–276, 2001.

[24] J. Si, A. Barto, W. Powell, and D. Wunsch, Eds., *Handbook of Learning and Approximate Dynamic Programming*. Wiley-IEEE Press, 2004.

[25] G. Venayagamoorthy, R. Harley, and D. Wunsch, "Comparison of heuristic dynamic programming and dual heuristic programming adaptive critics for neurocontrol of a turbogenerator," *IEEE Trans. Neural Networks*, vol. 13, no. 3, pp. 764–773, 2002.

[26] ——, "Dual heuristic programming excitation neurocontrol for generators in a multimachine power system," *IEEE Trans. Ind. Appl.*, vol. 39, no. 2, pp. 382–394, 2003.

[27] S. Ferrari and R. Stengel, "Online adaptive critic flight control," *Journal of Guidance Control and Dynamics*, vol. 27, no. 5, pp. 777–786, 2004.

[28] S. Jagannathan and G. Galan, "Adaptive critic neural network-based object grasping control using a three-finger gripper," *IEEE Trans. Neural Networks*, vol. 15, no. 2, pp. 395–407, 2004.

[29] D. Han and S. Balakrishnan, "State-constrained agile missile control with adaptive-critic-based neural networks," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 4, pp. 481–489, 2002.

[30] C. Anderson, D. Hittle, M. Kretchmar, and P. Young, "Robust reinforcement learning for heating, ventilation, and air conditioning control of buildings," in *Handbook of Learning and Approximate Dynamic Programming*, J. Si, A. G. Barto, W. B. Powell, and D. Wunsch, Eds. Wiley-IEEE Press, August 2004, pp. 517–529.

[31] T. Landelius, "Reinforcement learning and distributed local model synthesis," Ph.D. dissertation, Linkoping University, Sweden, 1997.

[32] D. Prokhorov, R. Santiago, and D. Wunsch, "Adaptive critic designs: A case study for neurocontrol," *Neural Networks*, vol. 8, no. 9, pp. 1367–1372, 1995.

[33] S. Bradtke, B. Ydstie, and A. Barto, "Adaptive linear quadratic control using policy iteration," in *Proc. Am. Control Conf.* IEEE, 1994, pp. 3475–3479.

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

[35] D. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific, 2007.

[36] D. White and D. Sofge, *Handbook of intelligent control: neural, fuzzy, and adaptive approaches*. Van Nostrand Reinhold Company, 1992.

[37] D. Kleinman, "On an iterative technique for riccati equation computations," *IEEE Trans. Autom. Contr.*, vol. 13, no. 1, pp. 114–115, 1968.

[38] L. Baird, "Advantage updating," Wright Lab, Wright-Patterson Air Force Base, OH, Tech. Rep., 1993.

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

[40] R. Beard, G. Saridis, and J. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, pp. 2159–2178, 1997.

[41] M. Abu-Khalaf and F. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.

[42] D. Vrabie, M. Abu-Khalaf, F. Lewis, and Y. Wang, "Continuous-time ADP for linear systems with partially unknown dynamics," in *Proc. IEEE Int. Symp. Approx. Dyn. Program. Reinf. Learn.*, 2007, pp. 247–253.

[43] J. Campos and F. Lewis, "Adaptive critic neural network for feedforward compensation," in *Proc. Am. Control Conf.*, vol. 4, 1999.

[44] O. Kuljaca and F. Lewis, "Adaptive critic design using non-linear network structures," *Int. J. Adapt Control Signal Process.*, vol. 17, no. 6, pp. 431–445, 2003.

[45] Y. Kim and F. Lewis, *High-level feedback control with neural networks.* World Scientific Pub Co Inc, 1998.

[46] P. M. Patre, W. MacKunis, C. Makkar, and W. E. Dixon, "Asymptotic tracking for systems with structured and unstructured uncertainties," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 2, pp. 373–379, 2008.

[47] B. Xian, D. M. Dawson, M. S. de Queiroz, and J. Chen, "A continuous asymptotic tracking control strategy for uncertain nonlinear systems," *IEEE Trans. Autom. Control*, vol. 49, pp. 1206–1211, 2004.

[48] R. Howard, *Dynamic programming and Markov processes.* Technology Press of Massachusetts Institute of Technology (Cambridge), 1960.

[49] J. Tsitsiklis, "On the convergence of optimistic policy iteration," *The Journal of Machine Learning Research*, vol. 3, pp. 59–72, 2003.

[50] R. Sutton, "Generalization in reinforcement learning: Successful examples using sparse coarse coding," *Advances in neural information processing systems*, pp. 1038–1044, 1996.

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

[52] D. Kirk, *Optimal Control Theory: An Introduction.* Dover Pubns, 2004.

[53] M. Crandall and P. Lions, "Viscosity solutions of Hamilton-Jacobi equations," *Transactions of the American Mathematical Society*, vol. 277, no. 1, pp. 1–42, 1983.

[54] M. Bardi and I. Dolcetta, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations.* Springer, 1997.

[55] J. Betts, *Practical methods for optimal control using nonlinear programming.* Society for Industrial Mathematics, 2001, no. 3.

[56] Q. Gong, W. Kang, and I. Ross, "A pseudospectral method for the optimal control of constrained feedback linearizable systems," *IEEE Trans. Autom. Contr.*, vol. 51, no. 7, pp. 1115–1129, 2006.

[57] R. Freeman and P. Kokotovic, "Optimal nonlinear controllers for feedback linearizable systems," in *Proc. Am. Control Conf.*, Jun. 1995, pp. 2722–2726.

[58] K. Dupree, P. Patre, Z. Wilcox, and W. Dixon, "Asymptotic optimal control of uncertain nonlinear euler-lagrange systems," *Automatica*, 2010.

[59] M. Sepulchre, R. Jankovic and P. V. Kokotovic, *Constructive Nonlinear Control*. New York: Springer-Verlag, 1997.

[60] M. Krstic and Z.-H. Li, "Inverse optimal design of input-to-state stabilizing nonlinear controllers," *IEEE Trans. Autom. Control*, vol. 43, no. 3, pp. 336–350, March 1998.

[61] M. Krstic, "Inverse optimal adaptive control-the interplay between update laws, control laws, and Lyapunov functions," in *Proc. Am. Control Conf.*, 2009, pp. 1250–1255.

[62] D. Mayne and H. Michalska, "Receding horizon control of nonlinear systems," *IEEE Trans. Autom. Contr.*, vol. 35, no. 7, pp. 814–824, 1990.

[63] M. Morari and J. Lee, "Model predictive control: past, present and future," *Computers & Chemical Engineering*, vol. 23, no. 4-5, pp. 667–682, 1999.

[64] B. Foss, T. Johansen, and A. Sørensen, "Nonlinear predictive control using local models–applied to a batch fermentation process," *Control Engineering Practice*, vol. 3, no. 3, pp. 389–396, 1995.

[65] J. Richalet, A. Rault, J. Testud, and J. Papon, "Model predictive heuristic control:: Applications to industrial processes," *Automatica*, vol. 14, no. 5, pp. 413–428, 1978.

[66] G. Saridis and C. Lee, "An approximation theory of optimal control for trainable manipulators," vol. 9, no. 3, 1979.

[67] S. Balakrishnan, "Adaptive-critic-based neural networks for aircraft optimal control," *J. Guid. Contr. Dynam.*, vol. 19, no. 4, pp. 893–898, 1996.

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

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

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

[71] P. M. Patre, W. MacKunis, K. Kaiser, and W. E. Dixon, "Asymptotic tracking for uncertain dynamic systems via a multilayer neural network feedforward and RISE feedback control structure," *IEEE Trans. Autom. Control*, vol. 53, no. 9, pp. 2180–2185, 2008.

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

[73] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos, *Nonlinear and Adaptive Control Design*. John Wiley & Sons, 1995.

[74] A. Filippov, "Differential equations with discontinuous right-hand side," *Am. Math. Soc. Transl.*, vol. 42 no. 2, pp. 199–231, 1964.

[75] ——, *Differential equations with discontinuous right-hand side*. Netherlands: Kluwer Academic Publishers, 1988.

[76] G. V. Smirnov, *Introduction to the theory of differential inclusions*. American Mathematical Society, 2002.

[77] J. P. Aubin and H. Frankowska, *Set-valued analysis*. Birkhuser, 2008.

[78] F. H. Clarke, *Optimization and nonsmooth analysis*. SIAM, 1990.

[79] D. Shevitz and B. Paden, "Lyapunov stability theory of nonsmooth systems," *IEEE Trans. Autom. Control*, vol. 39 no. 9, pp. 1910–1914, 1994.

[80] B. Paden and S. Sastry, "A calculus for computing Filippov's differential inclusion with application to the variable structure control of robot manipulators," *IEEE Trans. Circuits Syst.*, vol. 34 no. 1, pp. 73–82, 1987.

[81] F. L. Lewis, "Nonlinear network structures for feedback control," *Asian J. Control*, vol. 1, no. 4, pp. 205–228, 1999.

[82] M. Niethammer, P. Menold, and F. Allgower, "Parameter and derivative estimation for nonlinear continuous-time system identification," in *5th IFAC Symposium Nonlinear Control Systems (NOLCOS01), Russia*, 2001.

[83] T. Floquet, J. Barbot, W. Perruquetti, and M. Djemai, "On the robust fault detection via a sliding mode disturbance observer," *Int. J. Control*, vol. 77, no. 7, pp. 622–629, 2004.

[84] W. Xu, J. Han, and S. Tso, "Experimental study of contact transition control incorporating joint acceleration feedback," vol. 5, no. 3, pp. 292–301, 2000.

[85] P. Schmidt and R. Lorenz, "Design principles and implementation of acceleration feedback to improve performance of dc drives," *IEEE Trans. Ind. Appl.*, vol. 28, no. 3, pp. 594–599, 1992.

[86] N. Olgac, H. Elmali, M. Hosek, and M. Renzulli, "Active vibration control of distributed systems using delayed resonator with acceleration feedback," *J. Dyn. Syst. Meas. Contr.*, vol. 119, p. 380, 1997.

[87] K. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, vol. 1, no. 1, pp. 4–27, 1990.

[88] M. Polycarpou and P. Ioannou, "Identification and control of nonlinear systems using neural network models: Design and stability analysis," *Systems Report 91-09-01, University of Southern California*, 1991.

[89] G. A. Rovithakis and M. A. Christodoulou, "Adaptive control of unknown plants using dynamical neural networks," *IEEE Trans. Syst. Man Cybern.*, vol. 24, pp. 400–412, 1994.

[90] A. Poznyak, W. Yu, E. Sanchez, and J. Perez, "Nonlinear adaptive trajectory tracking using dynamic neural networks," *IEEE Trans. Neural Networks*, vol. 10, no. 6, pp. 1402–1411, 2002.

[91] W. Yu, A. Poznyak, and X. Li, "Multilayer dynamic neural networks for non-linear system on-line identification," *Int. J. Control*, vol. 74, no. 18, pp. 1858–1864, 2001.

[92] R. Sanner and J. Slotine, "Stable recursive identification using radial basis function networks," in *American Control Conference*, 1992, pp. 1829–1833.

[93] S. Lu and T. Basar, "Robust nonlinear system identification using neural-network models," *IEEE Trans. Neural Networks*, vol. 9, no. 3, pp. 407–429, 2002.

[94] J. Huang and F. Lewis, "Neural-network predictive control for nonlinear dynamic systems with time-delay," *IEEE Trans. Neural Networks*, vol. 14, no. 2, pp. 377–389, 2003.

[95] S. Ibrir, "Online exact differentiation and notion of asymptotic algebraic observers," *IEEE Trans. Automat. Contr.*, vol. 48, no. 11, pp. 2055–2060, 2003.

[96] L. Vasiljevic and H. Khalil, "Error bounds in differentiation of noisy signals by high-gain observers," *Systems & Control Letters*, vol. 57, no. 10, pp. 856–862, 2008.

[97] A. Levant, "Robust exact differentiation via sliding mode technique," *Automatica*, vol. 34, no. 3, pp. 379–384, 1998.

[98] M. Gupta, L. Jin, and N. Homma, *Static and dynamic neural networks: from fundamentals to advanced theory*. Wiley-IEEE Press, 2003.

[99] K. Funahashi and Y. Nakamura, "Approximation of dynamic systems by continuous-time recurrent neural networks," *Neural Networks*, vol. 6, pp. 801–806, 1993.

[100] H. Khalil and F. Esfandiari, "Semiglobal stabilization of a class of nonlinear systems using output feedback," *IEEE Trans. Autom. Control*, vol. 38, no. 9, pp. 1412–1415, 1993.

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

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

[103] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural Networks*, vol. 2, pp. 359–366, 1985.

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

[105] A. F. Fillipov, *Differential Equations with Discontinuous Righthand Sides*. Kluwer Academic Publishers, 1988, pp. 48-122.

[106] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.

[107] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*. Athena Scientific, 1996.

[108] L. Busoniu, R. Babuska, B. De Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*. CRC Press, 2010.

[109] G. Lendaris, L. Schultz, and T. Shannon, "Adaptive critic design for intelligent steering and speed control of a 2-axle vehicle," in *Int. Joint Conf. Neural Netw.*, 2000, pp. 73–78.

[110] J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," *Proc. Nat. Acad. Sci. U.S.A.*, vol. 81, no. 10, p. 3088, 1984.

[111] A. Poznyak, E. Sanchez, and W. Yu, *Differential neural networks for robust nonlinear control: identification, state estimation and trajectory tracking*. World Scientific Pub Co Inc, 2001.

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

[113] G. Chowdhary and E. Johnson, "Concurrent learning for convergence in adaptive control without persistency of excitation," in *Proc. IEEE Conf. Decis. Control*. IEEE, 2010, pp. 3674–3679.

[114] T. Basar and G. J. Olsder, *Dynamic Noncooperative Game Theory*. SIAM, 1999.

[115] D. Vrabie and F. Lewis, "Integral reinforcement learning for online computation of feedback nash strategies of nonzero-sum differential games," in *Proc. IEEE Conf. Decis. Control*. IEEE, 2010, pp. 3066–3071.

[116] K. Vamvoudakis and F. Lewis, "Multi-player non-zero-sum games: Online adaptive learning solution of coupled hamilton-jacobi equations," *Automatica*, 2011.

[117] T. Jaakkola, S. Singh, and M. Jordan, "Reinforcement learning algorithm for partially observable markov decision problems," *Advances in neural information processing systems*, pp. 345–352, 1995.

BIOGRAPHICAL SKETCH

Shubhendu Bhasin was born in Delhi, India in 1982. He received his Bachelor of Engineering degree in manufacturing processes and automation engineering from Netaji Subhas Institute of Technology, University of Delhi, India in 2004. From August 2004 to March 2006, he worked at Tata Elxsi Ltd., Bangalore, as a Design and Development Engineer in their embedded systems division. Thereafter, he joined Conexant Systems Ltd., Noida, where he worked as a Software Engineer till July 2006. He then joined the Nonlinear Controls and Robotics (NCR) research lab in the Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, to pursue his M.S. and doctoral research under the advisement of Dr. Warren E. Dixon. He received his M.S. in mechanical engineering in Spring of 2009 and PhD in Summer of 2011. His research interests include reinforcement learning-based control, approximate dynamic programming, differential games, robust and adaptive control of mechanical systems.