SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING USING BARRIER TRANSFORMATION

By

S M NAHID MAHMUD

Bachelor of Science in Mechanical Engineering

Islamic University of Technology

Gazipur, Bangladesh

December, 2015

Submitted to the Faculty of the
Graduate College of
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
MAY, 2021

SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING USING BARRIER TRANSFORMATION

Thesis Approved:

Dr. Rushikesh Kamalapurkar

Thesis Advisor

Dr. He Bai

Dr. Gary Yen

ACKNOWLEDGMENTS

All praises are due to Allah, the most glorified, the most high, for giving me enough strength throughout the course of my graduate education. I want to express my sincere gratitude towards my advisor, Dr. Rushikesh Kamalapurkar, for his guidance, patience, support, and encouragement. Without his valuable insights and contributions, none of this thesis work would have been possible. I want to express my hearty gratitude to Dr. Gary Yen and Dr. He Bai for being a part of my defense committee. I am also very grateful to Dr. Nivison, and Dr. Bell for their valuable suggestions which have helped me to write this thesis. Besides, I would like to thank all SCC and CoRAL research group members for fostering a positive and engaging learning environment. Thanks to my father Mahtab Uddin and my mother Nasima Akhter, my younger brother Niaz, relatives, friends, the fellow graduate students of MAE, OSU, Bangladeshi students association (BSA, OSU), respected faculties and staffs of OSU, and those who are directly or indirectly involved with this thesis work for their encouragement. Finally, I am grateful to the MAE department and graduate college of Oklahoma State University for providing me the opportunity and financial assistance to pursue my MS degree in this prestigious institution. I would also like to extend thanks to Air Force Research Laboratories for their financial support under award number FA8651-19-2-0009.[1]

[1]Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

Name: S M Nahid Mahmud

Date of Degree: DECEMBER, 2020

Title of Study: SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING USING BARRIER TRANSFORMATION

Major Field: Mechanical and Aerospace Engineering

Abstract:

The ability to learn and execute optimal control policies safely is critical to the realization of complex autonomy, especially where task restarts are not available and/or when the systems are safety-critical. Safety requirements are often expressed in terms of state and/or control constraints. Methods such as barrier transformation and control barrier functions have been successfully used for safe learning in systems under state constraints and/or control constraints, in conjunction with model-based reinforcement learning to learn the optimal control policy. However, existing barrier-based safe learning methods rely on fully known models and full state feedback. In this thesis, two different safe model-based reinforcement learning techniques are developed. One of the techniques utilizes a novel filtered concurrent learning method to realize simultaneous learning and control in the presence of model uncertainties for safety-critical systems, and the other technique utilizes a novel dynamic state estimator to realize simultaneous learning and control for safety-critical systems with a partially observable state. The applicability of the developed techniques is demonstrated through simulations, and to illustrate their effectiveness, comparative simulations are presented wherever alternate methods exist to solve the problem under consideration. The thesis concludes with a discussion about the limitations of the developed techniques. Extensions of the developed techniques are also proposed along with the possible approaches to achieve them.

Contents

Chapter Page

Chapter Page

List of Tables

Table Page

List of Figures

Chapter I

INTRODUCTION

1.1 Motivation

Since the beginning of time, humans have attempted to imitate natural techniques in order to solve human design problems. In nineteenth century, Charles Darwin showed that species correct their behaviors based on interactions with the environment in order to stay safe and/or to avail benefits [1]. For example, Ivan Pavlov used inducing conditional reflexes with simple reward or punishment to teach dogs behavior patterns [2]. Therefore, it can be said that learning the correct behavior to survive from interactions with the environment is a highly desirable characteristic of a species/cognitive agent. A cognitive agent can be described as an agent that acquires knowledge and understanding through thinking, experience, and the senses to produce a specific result.

To humans, one of the most coveted design tasks is to perform assigned tasks precisely while remaining safe. Ultimately, this entailed the development of cognitive agents/autonomous agents. Repeatability, accuracy, and lack of physical fatigue are crucial advantages of autonomous agents over humans. Additionally, autonomous systems can provide advantages in settings where humans may be in danger, such as war zones and hostile environments. In order to maximize the likelihood of success and reduce the number of casualties, using autonomous systems for complex, high-risk tasks has long been a goal of humans.

In this thesis, the interaction between an agent and it's environment is modeled using actions, states, and rewards. The environment will be interpreted as the sur-

roundings or circumstances under which the agent operates. Furthermore, we will refer to an enticing stimulus, delivered to an agent to change its actions as a reward. Similarly, a penalty can be described as an aversive stimulus applied to an agent to change its actions.

Typically, any action taken by an agent affects the state of the system (i.e., the agent and the environment), and the agent is rewarded (or penalized) for it. Learning, in this context, amounts to the synthesis of a behavior policy/strategy, is defined as a map from the state space to the action space to complete a given task. Most natural and artificial methods to learn policies involve "trial and error" where policies are learned and refined by implementing them and observing the resulting rewards. While "trial and error" or "learning from failure" is an integral part of the learning process, safety-critical systems require learning techniques where the errors and failures result in, perhaps suboptimal, but safe behavior. As a result, safely learning a correct policy which ensures both safety and correct action is a critical capability for an agent to possess.

What exactly is *safety*, *correct action*, and *correct policy*? Depending on the objectives of the agent-environment interaction, safety can be described in several ways. Intuitively, safety connotes the ability to avoid danger. In robotics, guidance, and control applications, safety is often expressed in terms of state and/or action space constraints. Correct action is often described as the action that maximizes the cumulative reward or minimizes the cumulative cost. In robotics, guidance, and control applications, the cumulative cost is often interpreted as a Bolza cost, i.e., the combination of a Lagrange cost and a Meyer cost. The Lagrange cost is the cumulative penalty accumulated along a path traversed by the agent, and the Meyer cost is the penalty at the boundary. Policies with lower total costs are considered better, and policies that minimize the total cost are considered optimal.

In robotics, guidance, and control applications, correctness and safety of a policy

are quantified in terms of a Bolza cost and state-space and/or action space constraints, respectively. In addition to safety and optimality, stability is another critical characteristic of an autonomous system. Stability is often described as the agent's desired response with no intolerable variation in response to parameter changes. In summary, the goal of the thesis is to develop learning techniques that enable an agent to learn an policy to achieve a task while maintaining safety and stability doing learning and execution.

In robotics, stability has traditionally received far less attention than safety [3]. In general, policy-based trajectory planners use the agent's exact dynamical model and knowledge of the environment to ensure that the agent/robot is safe by maintaining defined constraints (state space and/or action space constraints) and planning obstacle avoidance maneuvers [4]. Sample-based methods generate policy by extracting samples (also known as useful information) from an agent's state and/or action space and then using the samples to design trajectories. To ensure safety, sample-based policies take into account an agent's dynamics. Sample-based policies face a tradeoff in that they must strive to support either safety and persistent feasibility (i.e., the existence of a solution that meets the constraints on state space and/or action space) or performance (i.e., optimality) [5–7]. Another safe trajectory planner is Nonlinear Model Predictive Control (NMPC), where the system dynamics are used for planning and obstacles are treated as constraints in an optimization program over the control inputs of a robot. Policies generated using NMPC face the same tradeoffs as sample-based policies [8–15]. Reachability-based methods, another type of trajectory planners, precalculate a reachable set using the robot's motion, then use these reachable sets to ensure collision avoidance at runtime. Reachability-based policies enable strict safety guarantees and some persistent feasibility guarantees but the precomputing of the reachable sets are often inefficient as they over-approximate the reachable sets [16–21].

Control design techniques based on Lyapunov functions and control Lyapunov functions have been used to ensure stability in control systems, and control barrier functions have been used to resolve safety concerns. Recently, control barrier functions have been merged with control Lyapunov functions to create control synthesis techniques that ensure both stability and safety. However, to achieve the correct policy, these methods must ensure that the effort is minimized, i.e., they must solve an optimization problem. In most cases, these optimization problems must consider a large number of state space and/or action space constraints which leads to the dilemma of choosing between optimally and safety [22–24].

The barrier function-based system transformation (BT) method solves this problem. A complete state constrained and/or action space-constrained optimal control problem is converted into a similar, unconstrained optimization problem using this transformation process. The state constraints can be guaranteed if the initial state is within the prescribed bound, which guarantees safety [25]. Thus, we seek a method that can be used in conjunction with BT to obtain the correct policy that stabilizes the agent while keeping it safe and minimizing its Bolza cost.

Finding the optimal policy that minimizes the total Lagrange and Meyer cost (Bolza cost) is known as the Bolza optimal control problem. Obtaining an analytical solution to the Bolza problem is often infeasible if the system dynamics are nonlinear. On the other hand, numerous numerical solution techniques are available to solve Bolza problems; however, numerical solution techniques require exact model knowledge and are realized via open-loop implementation of offline solutions. Open-loop implementations are sensitive to disturbances, changes in objectives, and changes in the system dynamics; hence, we seek online closed-loop solutions of optimal control problems to solve this drawback [26–34].

We can find these closed-loop solutions using the value function. Typically value function is described with the respect of a given policy, i.e., how good it is for an agent

to be in a given state under a given policy. The notion of "how good" is expressed in terms of the total accumulated cost. In other words, a value function evaluated at a given state and under a given policy is defined as the total accumulated cost starting from the given state under the given policy. Under the general conditions we can now say that the optimal policy value function will be our optimal policy. Therefore, to solve the online closed-loop optimal control problem, we need to determine the optimal value function.

In the past, value function-based dynamic programming (DP) techniques such as policy iteration (PI) and value iteration (VI) have been developed as useful tools for optimal control synthesis for systems with finite state and action spaces. However, as the state space's size increases, computing both PI and VI become practically infeasible [29, 33, 35–37]. To tackle this problem, approximate dynamic programming (ADP) techniques can be used. ADP algorithms approximate the classical PI and VI algorithms to compute approximate optimal value function using a parametric approximation of the policy or the value function, i.e., if the policy or the value function can be parameterized with sufficient accuracy using a small number of parameters, the optimal control problem reduces to an approximation problem in the parameter space. Furthermore, this formulation lends itself to an online solution approach using reinforcement learning (RL) where the parameters are adjusted on the-fly using input-output data [38–42]. Despite the drawbacks such as needing: 1) sufficient exploration of the state-action space and 2) some insight into the dynamics of the system, RL has given rise to practical techniques that can synthesize nearly optimal policies to control nonlinear systems that have large state and action spaces and unknown or partially known dynamics.

In online implementations of RL, the control policy derived from the approximate value function is used to control the system; hence, obtaining a good approximation of the value function is critical to the closed-loop system's stability. Similar to adaptive

control, the sufficient exploration condition manifests itself as a persistence of excitation (PE) condition when RL is implemented online. In general, it is difficult to guarantee PE a priori; hence, to ensure PE, a probing signal is applied to the controller using trial and error. In the stability analysis, the probing signal is ignored; hence, the closed-loop implementation's stability cannot be guaranteed. However, model-based RL (MBRL) schemes has been developed which uses finite exciation (FE) to relax the PE condition. Using FE facilitated by model-based extrapolation, stability and convergence of online RL can established under a PE-like condition that, while impossible to guarantee a priori, can be verified online [43, 44]. On the other hand, MBRL methods are prone to failure due to inaccurate models such as models with parametric uncertainties and/or partially observable models. Online MBRL methods that handle modeling uncertainties are motivated by tasks that require systems to operate in dynamic environments with changing objectives and system models, and accurate models of the system and environment are generally not available in complex tasks due to sparsity of data.

In this thesis, a novel MBRL technique combined with BT has been develop for models with parametric uncertainties to achieve the correct policy. To address, the partial observability of the models, another MBRL technique combined with BT has been developed for continuous nonlinear control affine systems in the Brunovsky form. The applicability of the developed methods is demonstrated through simulations, and to illustrate their effectiveness, comparative simulations are presented wherever alternate methods exist to solve the problem under consideration. The thesis concludes with a discussion about the limitations of the developed technique, and further extensions of the technique are proposed, along with the possible approaches to achieve them.

1.2 Literature Review

One way to generate safe trajectories for the agent is to use different sample-based methods such as rapidly-exploring random trees (RRT), probabilistic road maps (PRM), fast marching trees (FMT), and so on [5–7]. Sample-based techniques map trajectories by sampling from the control input and/or state space of a dynamical system. It results temporal and/or spatial discretization of the system's dynamic model. A finer discretization usually allows for stronger claims about the safety of such methods, but at the expense of increased computational cost and, resulting a reduction in performance [4, 21]. With respect to an arbitrary cost function, sample-based methods may generate optimal trajectories but do not ensure safety. To ensure safety, the sample-based methods incorporate the dynamics of an agent [5,21,45], and to perform obstacle avoidance, obstacles are buffered to compensate for the robot's shape [4, 21], resulting the reduction of performance by reducing the free space available for planning. On the other hand, trajectory planners need to achieve persistent feasibility, (Planning is persistently feasible if there always exists a safe trajectory or stopping maneuver before the robot completes executing the previously planned trajectory [21]) to be feasible in real life. Ensuring persistent feasibility demands additional computational cost which causes reduction of performance. [45–47] shows that linearizing the robot's model results rapid results but one may lose safety guarantees. This means that sample-based methods suffer from the tradeoff between safety guarantee and performance, i.e., optimality.

Nonlinear Model Predictive Control (NMPC) techniques, another type of safe trajectory planner, experience the same tradeoff as sample-based methods. NMPC techniques map trajectories by formulating an optimization program over a system's control inputs, with the dynamics and obstacles treated as constraints. In general, NMPC techniques discretize time, and linearize the dynamical model of the system to make the optimization problem feasible [8–11, 15, 21]. To avoid linearization (still

requires discretization), pseudo-spectral methods approximate the NMPC program with polynomial functions [12]. An alternative to these types of discretizations and simplifications of the dynamics is Sequential Action Control (SAC) [13, 14]. The obstacle avoidance NMPC techniques have been shown in [15]. On the other hand, various methods such as fine discretization [15], linearization of the dynamics [10], tracking a precomputed a dynamically feasible reference trajectory [48], exploitation of environement structure [11], usage of SAC [13, 14], computation of the viability kernels (assuming the environment is known) [49] have been attempted to ensure persistent feasibility.

Reachability-based techniques uses precomputed reachable sets to synthesize safe tracking controllers to ensure collision avoidance, and/or considering state constraints, and/or control constraints at runtime [21]. In literature, a number of Reachability-based techniques exist to compute reachable sets such as sums-of-square (SOS) programming [17, 19], Hamilton-Jacobi-Bellman (HJB) reachability [16, 50], zonotope reachable sets [18, 20]. By computing overapproximations of the reachable sets of robots in state space, the SOS and zonotope attempts safety [19, 20]. The HJB approach, on the other hand, poses its offline reachability analysis as a differential game between a complex model (i.e., high fidelity) of a system and a simplified planning model. The numerical solution of this offline reachability analysis is not provably overapproximative [51]. For the SOS approach, with a finite library of reachable sets, one attempts to compose the reachable sets sequentially at runtime [19, 21] to address persistent feasibility, though it is unclear how to continue when no reachable sets are available. Zonotope approach is used to valid a single maneuvur though it is unknown how to promise the existence of valid manuevurs [18] during the whole run time. For the HJB approach, one can simultaneously plan exploration trajectories and trajectories that return the system to a previously known safe location [52]; however, due to the reachability analysis' underlying conservatism, which restricts the

system's free space making it stuck at the same position for a long time. A more recent reachability-based approach in [21] has taken a system decomposition approach to improve the tractability of computing reachable sets, resulting strict safety. Persistent feasibility is achieved by prescribing a minimum sensor horizon and a minimum duration for the planned trajectories. However, this approach is still burdened with calculating expensive reachable sets.

To avoid the calculation of reachable sets, [22, 23] reintroduced the concept of control barrier functions. Originally, the concept of control barrier functions was developed by the inspiration of set invariance concept introduced in the 1940s. In 1942, Nagumo provided necessary and sufficient conditions for set invariance [53]. Later, [54] showed details about the safety in terms of set variance. Later, in the 2000s, barrier certificates were introduced as a convenient tool to formally prove safety of nonlinear and hybrid systems [55–57]. The barrier certificates were motivated by its use in the optimization literature where barrier functions are added to cost functions to avoid undesirable regions. A barrier function is a continuous function whose value on a point increases to infinity as the point approaches the boundary of the feasible region of an optimization problem [58]. A barrier certificate is a function of state satisfying some conditions on both the function itself and its time derivative along the flow of the system, and a barrier between potential system trajectories and the given unsafe region denotes that a given system is safe [55]. In achieving this, one do not need to compute the reachable set neither we need to have explicit computation of system flows.

Later, barrier certificate approach has been extended to a Lyapunov-like approach known as Barrier Lyapunov function in [59], but the definition of Barrier Lyapunov function is different than the one considered in the current literature, while the conditions of Barrier Lyapunov function ensure safety over the entire set(not just on the boundary), they also enforce invariance of every level set. Meanwhile, the work on

viability theory [60,61] extended the above-mentioned approaches to open dynamical systems. This facilitated a move from invariant sets to controlled invariant sets, which are sets that can be made invariant by designing a controller appropriately. Inspired by the barrier certificate, the notion of control barrier function was first introduced in [62]. Later, [22, 23] redefined the concept of control barrier function to minimize the restriction by providing necessary and sufficient conditions. Safe stabilizing controllers can be synthesised by the control barrier function method by embedding set invariance conditions within an optimal problem. The problem reduces to solving a quadratic program (QP) at each time stage to obtain the optimal control if the control system is affine in controls and the cost is quadratic [22]. This QP-based approach works myopically, which means the safe control is just a function of the current state [24, 63], which means this method can guarantee local safety at each time point, but the safety restriction is satisfied based on how often the QP is solved [64]. This creates a problem of selecting step sizes during solving QP, a step size that is too small can lead to additional computation, whereas a step size that is too big can lead to risky actions. Moreover, if QP based approach is designed too conservatively, it may use unnecessary intervention when the situation is not dangerous; if QP based approach is too optimistic, it may allow the state to get too close to the boundary of the safe set and have to invoke large intervention to prevent the state from approaching to the bound of the set, and it may become infeasible, and fails. To increase the feasibility of the QP a relaxation variable is added, which can easily become infeasible in the presence of conflicting control, stability, and safety constraints [65]. While increasing feasibility, this relaxation no longer guarantees convergence to the desired equilibrium point [66]. To address these issues, [24] proposes to reformulate constrained QP as an unconstrained optimal control problem with new augmented instantaneous cost.

Since developing analytical solutions for nonlinear systems much more difficult,

numerical solutions are sought for solving optimal control problems of general nonlinear systems [67]. Formulating the optimal control problem in terms of a Hamiltonian and then numerically solving a two-point boundary value problem for the state and co-state equations is a typical approach [26, 27]. Another option is to directly transpose the optimal control problem into a nonlinear programming problem and then solve the resulting nonlinear program [68–73]. By avoiding the need to solve the Hamilton-Jacobi-Bellman equation, the nonlinear optimal control problem can be solved using inverse optimal control [74–81]. However, these numerical solution techniques require exact model knowledge and are realized via open-loop implementation of offline solutions. Open loop implementations are sensitive to disturbances, changes in objectives, and changes in the system dynamics; hence, online closed-loop solutions of optimal control problems need to be sought. One way to find closed-loop solutions is to use value functions [24] which can be obtained by the DP techniques. The literature on DP techniques focused on the theory of optimality is substantial [28–34]. The need for exact model knowledge limits the applicability of conventional DP techniques like PI and VI. Model-free reinforcement learning techniques such as Q-learning [31] and temporal difference learning [29,36] avoid the need for exact model knowledge. These methods, however, require that the states and actions be on finite sets. Despite the fact that the theory was developed for finite state spaces of any scale, model-free reinforcement learning techniques can only be applied in small state spaces. Under the umbrella of neuro-dynamic programming [33, 36–42], extensions of simulation-based reinforcement learning algorithms have been studied for systems with countable state and action-spaces.

Both PI and VI become computationally infeasible as the size of the space grows. The need for excessive computation can be avoided if the approximate optimal value function instead of the exact optimal value function is computed. To obtain an approximation to the optimal value function using PI, the generalized Hamilton-Jacobi-

Bellman equation must be solved approximately in each iteration [35]. Several methods to approximate the solutions to the generalized Hamilton-Jacobi-Bellman equation have been studied in the literature. The generalized Hamilton-Jacobi-Bellman equation can be solved numerically using perturbation techniques [82–84], finite difference [85–87] and finite element [88–90] techniques, or using approximation methods such as Galerkin projections [91,92]. In this thesis, a linear-in-the-parameters approximation scheme developed in [93, 94] has been used to approximate value function. The characteristics of the approximation scheme, also known as the Universal Approximation theorem, can be established using the Stone-Weierstrass theorem [94,95]. This theorem states that a single layer neural network can simultaneously approximate a function and its derivative given a sufficiently large number of basis functions. The function approximation error, along with its derivative can be made arbitrarily small by increasing the number of basis functions used in the approximation. To ensure system stability during the learning phase, a two-network approach is utilized, where in addition to the value function, the policy is also approximated using a parametric approximation. The critic learns the value of a policy by updating the weights, and the actor improves the current policy by updating the weights.

The two-network approach known as the actor-critic architecture is one of the most widely used architectures to implement generalized PI algorithms [28,30,36,42,96,97]. Actor-critic methods were first developed in [98] for systems with finite state and action-spaces, and in [28] for systems with continuous state and action-spaces using neural networks to implement the actor and the critic. The actor can learn directly to minimize the estimated cost-to-go, where the estimate of the cost-to-go is obtained by the critic [28, 42, 97–100]. The actor can also be tuned to minimize the Bellman error (also known as the temporal-difference error) [101]. The critic network can be tuned using the method of temporal differences [28, 29, 36, 39, 40, 42, 102] or using heuristic dynamic programming [30, 37, 103] or its variants [97, 104, 105].

The iterative nature of actor-critic methods makes them particularly suitable for offline computation and for discrete-time systems [106–117]. A continuous-time formulation of actor-critic methods was first developed in [118]. In [118], the actor and the critic weights are tuned continuously using an adaptive update law designed as a differential equation. While no stability or convergence results are provided in [118], the developed algorithms can be readily utilized to simultaneously learn and utilize an approximate optimal feedback controller in real-time for nonlinear systems. A sequential (one network is tuned at a time) actor-critic method that does not require complete knowledge of the internal dynamics of the system is presented in [119]. Convergence properties of actor-critic methods for continuous-time systems where both the networks are concurrently tuned are examined in [120], and a Lyapunov-based analysis that concurrently examines convergence and stability properties of an online implementation of the actor-critic method is developed in [121].

In online implementations of reinforcement learning, the control policy derived from the approximate value function is used to control the system; hence, obtaining a good approximation of the value function is critical to the stability of the closed-loop system. Obtaining a good approximation of the value function online requires convergence of the weights of the actor-critic to their ideal values. Hence, similar to adaptive control, the sufficient exploration condition manifests itself as a persistence of excitation condition when reinforcement learning is implemented online.

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

satisfied [122–124].

In general, the controller does not ensure the persistence of excitation condition. Thus, in an online implementation, an ad-hoc exploration signal is often added to the controller [36, 125, 126]. Since the exploration signal is not considered in the the stability analysis, it is difficult the ensure stability of the online implementation. Moreover, the added probing signal causes large control effort expenditure and there is no means to know when it is sufficient to remove the probing signal. More recent works [43] have leveraged techniques from concurrent learning adaptive control [127] in the form of BE extrapolation which allows the BE to be evaluated at unexplored regions of the statespace. This extrapolation results in a virtual excitation of the system which facilitates weight estimate convergence [43].

The unconstrained optimal control problem posed by [24] is solved using ADP where the proximity penalty approach is cast into the framework of control barrier functions. The proximity approach was first introduced in the context of obstacle in [128] and [129], where an additional term that penalizes proximity to obstacles was added to the cost function. Since the added proximity penalty in [128] was finite, the ADP feedback could not guarantee obstacle avoidance, and an auxiliary controller was needed. In [129], a barrier-like function was used to ensure unbounded growth of the proximity penalty near the obstacle boundary. While this approach results in avoidance guarantees, it relies on the relatively strong assumptions that the value function is continuously differentiable over a compact set that contains the obstacles and penalty-induced discontinuities in the cost function. Therefore, while the control barrier function approved results in safety guarantees, the existence of a smooth value function, in spite of a nonsmooth cost function, needs to be assumed. Furthermore, to facilitate parametric approximation of the value function, the existence of a forward invariant compact set in the interior of the safe set needs to be established. Since the invariant set needs to be in the interior of the safe set, the penalty becomes

superfluous, and safety can be achieved through conventional Lyapunov methods.

This thesis is inspired by another approach to safe ADP, recently developed in [130], based on the idea of transforming a state and input constrained nonlinear optimal control problem into an unconstrained one with a type of saturation function was introduced in [131, 132]. In [130], input and state constrained optimal control problems are solved using ADP where the state constrained optimal control problem is transformed, using a barrier transformation (BT), into an equivalent, unconstrained optimization problem. In contrast to [24], mere stability of the transformed system is sufficient for the original system.

A MBRL approach to address the state-constrained optimal control problem appears in [133], where the results in [130] are extended to soften the restrictive persistence of excitation requirement. While the transformation in [130] and [133] results in verifiable safe feedback controllers, it requires exact knowledge of the system model, which is often difficult to obtain. [134], [135] proposed concurrent learning algorithm (CL) for online model learning, where information-rich past data is stored and concurrently used along with gradient based parameter update laws. Unlike the PE condition, an online verifiable rank condition on the stored data is sufficient for parameter convergence. Later, [136] proposed a filtered concurrent learning (FCL) algorithm based on the framework of composite adaptive control proposed in [10]. In addition to the low pass filtering, as performed in [10], the proposed method uses an integral of the filtered outputs to obviate the restrictive PE condition. In this thesis inspired by [136], a novel filtered concurrent learning technique for online model learning is developed, and later integrated with the BT method to yield a novel MBRL solution to the online state-constrained optimal feedback control problem under parametric uncertainty.

Apart from parametric uncertainties in exact system model, another significant drawback of the MBRL methods is that they require full state feedback measure-

ments, and as such, cannot be used if the system is partially observable. MBRL in partially observable systems has long been a focus of study in RL [137, 138], where partially observable Markov decision processes (POMDPs) have been utilized to realize MBRL using output feedback. In [139] an output-feedback MBRL method is developed for a class of nonlinear systems where the problem is formulated as a state estimation problem, and for a specific class of systems, an online solution is obtained that guarantees stability during the learning phase. To the best of the authors' knowledge, online RL solutions to safety-constrained optimal control problems in partially observable nonlinear continuous-time systems are not available in the literature.

1.3 Outline of the thesis

Chapter 1 serves as the introduction. This chapter focuses on the concerns and weaknesses of existing methods; motivating the thesis's development as well as offering a comprehensive overview of the state of the art.

Chapter 2 contains a brief review of available techniques used in the application of BT RL to deterministic continuous-time systems. This chapter also includes a brief review on the available methods used in the state of the art.

Chapter 3 presents the development of a safety aware model-based reinforcement learning technique using BT for the deterministic continuous-time systems with parametric uncertainties. This chapter implements a novel online MBRL based controller which uses BFs, BE extrapolation and a novel FCL method. A known BF transformation is applied to a constrained optimal control problem to generate an unconstrained optimal control problem in the transformed coordinates. MBRL is used to solve the problem online in the transformed coordinates in conjunction with the novel FCL to learn the unknown model parameters. Regulation of the system states to a neighborhood of the origin and convergence of the estimated policy to a neighborhood of the optimal policy is determined using a Lyapunov based stability analysis, and

simulations are presented to demonstrate the performance of the developed controller.

Chapter 4 implements the development of a safety aware model based reinforcement learning technique using BT for output-feedback optimal control of a class of deterministic continuous-time nonlinear systems. A novel online MBRL based controller which uses BFs, BE extrapolation and a novel state estimator method has been developed. This new state estimator takes the observable output feedback of the system using the BFs, and implements in the original coordination. Later, regulation of the transformed system states to a neighborhood of the origin and convergence of the estimated policy to a neighborhood of the optimal policy is determined using a Lyapunov based stability analysis, and a relation between the convergence of the original state systems and the converge of the transformed state systems has been shown. Simulations are performed to demonstrate the applicability and the effectiveness of the developed method.

Chapter 5 concludes the thesis. A summary of the thesis is provided along with a discussion on open problems and future research directions.

Proofs of the theorems and lemmas from chapters 3 and 4 are available in the appendix.

1.4 Contributions

This section details the contributions of this thesis over the state-of-the-art.

1.4.1 Safety-aware ADP for systems with Parametric Uncertainty

The main contributions of this chapter:

- Novel implementation of BT in deterministic nonlinear systems with parametric uncertainties.
- Novel FCL-based system identification for deterministic barrier transformed

systems with parametric uncertainties. Theoretical result guarantees that the estimated unknown parameters of the barrier transformed systems with parametric uncertainties converges to the real parameters.

- The inclusion of FCL makes the full state feedback controller robust to modeling errors and guarantees closed-loop stability under a finite (as opposed to persistent) excitation condition.

- Novel implementation of simulated experience in deterministic barrier transformed nonlinear systems with parametric uncertainties using FCL-based system identification.

- Detailed stability analysis to establish simultaneous online identification of barrier transformed system dynamics and online approximate learning of the optimal controller in barrier transformed coordinate, while maintaining barrier transformed system stability. The stability analysis shows that provided the system dynamics can be approximated fast enough, and with sufficient accuracy, simulation of experience based on the estimated model implemented via approximate BE extrapolation can be utilized to approximately solve an infinite-horizon optimal regulation problem online are provided.

- Novel theoretical result to guarantee that the optimal stabilizing controller developed for the barrier transformed system also stabilize the original system, and if the initial state is within the prescribed bound, the state constraints and/or control constraints can be guaranteed.

- Simulation results that demonstrate the approximate solution of an infinite-horizon optimal regulation problem online for an inherently unstable control-affine nonlinear system with uncertain drift dynamics without the addition of an external ad-hoc probing signal.

In summary, for the first time ever, a safety aware model based reinforcement learning method using BT has been developed for the system with parametric uncertainties.

1.4.2 Safety-aware ADP for partially observable systems

The main contributions of this chapter:

- Novel state estimator for deterministic barrier transformed partial observable systems. Theoretical result to guarantee that the estimated deterministic barrier transformed state converge to the real barrier transformed state.

- Detailed stability analysis to establish simultaneous online estimation of the state and online learning of an approximate optimal controller in barrier transformed coordinate, while maintaining system stability.

- Novel theoretical result to guarantee that the optimal stabilizing controller developed for the deterministic barrier transformed partial observable system also stabilize the original deterministic partial observable system, and if the initial state is within the prescribed bound, the state constraints and/or control constraints can be guaranteed.

- Simulation results that demonstrate the approximate solution of an infinite-horizon optimal regulation problem online for an inherently unstable control-affine nonlinear system with uncertain drift dynamics without the addition of an external ad-hoc probing signal.

In summary, a novel safety aware model based reinforcement learning method using BT has been developed for deterministic partially observable systems.

Chapter II

PRELIMINARIES

The focus of this thesis is to develop frameworks to guarantee safety while obtaining online approximate solutions to infinite horizon total-cost optimal control problems for nonlinear, partially observable, deterministic systems. This chapter serves as a brief introduction to safety certification methods, and model-based reinforcement learning methods that have been used to facilitate the development.

2.1 Notation

Throughout the thesis, unless otherwise specified, the notation $\mathbb{R}^n$ represents the $n-$dimensional Euclidean space, and the elements of $\mathbb{R}^n$ are interpreted as column vectors, $(\cdot)^T$ denotes the vector transpose operator. For any arbitary, $a \in \mathbb{R}$, $\mathbb{R}_{\geq a}$ denotes the interval $[a, \infty)$, and $\mathbb{R}_{>a}$ denotes the interval (a, ∞). Unless otherwise specified, an interval is assumed to be right-open. If any arbitary $a \in \mathbb{R}^m$ and $b \in \mathbb{R}^n$, then $[a; b]$ denotes the concatenated vector $\begin{bmatrix} a \\ b \end{bmatrix} \in \mathbb{R}^{m+n}$, and $[a, b]$ denotes the concatenated vector $\begin{bmatrix} a & b \end{bmatrix} \in \mathbb{R}^{1\times(m+n)}$. The notations I_n and 0_n denote the $n \times n$ identity matrix and the zero element of $\mathbb{R}^n$, respectively. The notation $f \in C^N(X, Y)$, $N \in \mathbb{R}_{\geq 0}$, denotes that the function $f : X \to Y$ is N-times continuously differentiable. Function names corresponding to state and control trajectories are reused to denote elements in the range of the function. For example, the notation $u(\cdot)$ is used to denote the function $u : \mathbb{R}_{\geq t_0} \to \mathbb{R}^m$, the notation u is used to denote an arbitrary element of $\mathbb{R}^m$, and the notation $u(t)$ is used to denote the value of the function $u(\cdot)$

evaluated at time t. The notation $f \in O(g)$ denotes that there exists $c, M \in \mathbb{R}_{>0}$ such that $|f(x)| \leq c|g(x)|,\ \forall\ x > M$

.

2.2 Method for Safety Certifications

2.2.1 Problem Formulation

A nonlinear control affine system as follows

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

where $x \in \Omega \subseteq \mathbb{R}^n$ denotes the system state, $u \in U \subset \mathbb{R}^m$ denotes the control input, $f : \Omega \to \mathbb{R}^n$ denotes the drift dynamics, and $g : \Omega \to \mathbb{R}^{n\times m}$ denotes the control effectiveness matrix. To ensure that the control problem is well posed, it is assumed that f and g are Lipschitz continuous on a set Ω that contains the origin as an interior point, $f(0) = 0$, and $\nabla f(x)$ is continuous and bounded for every bounded $x \in \Omega$.

2.2.2 Barrier Transformation

Definition 1 *Let the function $b : \mathbb{R} \to \mathbb{R}$, is referred to as barrier function (BF), be defined as*

$$b_{(a_i,A_i)}(y_i) \coloneqq \log \frac{A_i(a_i - y_i)}{a_i(A_i - y_i)}, \quad \forall i = 1, 2, \ldots, n, \tag{2}$$

where a_i and A_i are two constants satisfying $a_i < 0 < A_i$.

Let define $b_{(a,A)} : \mathbb{R}^n \to \mathbb{R}^n$ as $b_{(a,A)}(x) \coloneqq [b_{(a_1,A_1)}(x_1); \ldots; b_{(a_n,A_n)}(x_n)]$ with $a = [a_1; \ldots; a_n]$ and $A = [A_1; \ldots; A_n]$. Moreover, the inverse of (2) exists on interval (a_i, A_i), and is given by

$$b^{-1}_{(a_i,A_i)}(y_i) = a_i A_i \frac{e^{y_i} - 1}{a_i e^{y_i} - A_i}, \quad \forall y_i \in \mathbb{R}. \tag{3}$$

Derivative of (3) with respect to y_i yields

$$\frac{\mathrm{d}b^{-1}_{(a_i,A_i)}(y_i)}{\mathrm{d}y_i} = \frac{A_i a_i^2 - a_i A_i^2}{a_i^2 e^{y_i} - 2a_i A_i + A_i^2 e^{-y_i}}. \tag{4}$$

Consider the BF based state transformation

$$s_i \coloneqq b_{(a_i,A_i)}(x_i), \quad x_i = b^{-1}_{(a_i,A_i)}(s_i). \tag{5}$$

The time derivative of the transformed state can be computed using (4) and the chain rule as

$$\frac{\mathrm{d}s_i}{\mathrm{d}t} = \frac{\dot{x}_i}{\frac{\mathrm{d}b^{-1}_{(a_i,A_i)}(z)}{\mathrm{d}z}|_{z=s_i}}, \tag{6}$$

which yields the transformed dynamics

$$\dot{s}_i = \frac{f_i(x) + g_i(x)u}{\frac{\mathrm{d}b^{-1}_{(a_i,A_i)}(z)}{\mathrm{d}z}|_{z=s_i}} = F_i(s) + G_i(s)u, \tag{7}$$

where

$$F_i(s) = \frac{a_i^2 e^{s_i} - 2a_iA_i + A_i^2 e^{-s_i}}{A_i a_i^2 - a_i A_i^2} f_i\Big([b^{-1}_{(a_1,A_1)}(s_1);\ldots;b^{-1}_{(a_n,A_n)}(s_n)]\Big), \tag{8}$$

$$G_i(s) = \frac{a_i^2 e^{s_i} - 2a_iA_i + A_i^2 e^{-s_i}}{A_i a_i^2 - a_i A_i^2} g_i\Big([b^{-1}_{(a_1,A_1)}(s_1);\ldots;b^{-1}_{(a_n,A_n)}(s_n)]\Big). \tag{9}$$

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

$$\dot{s} = F(s) + G(s)u, \tag{10}$$

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

The method used in this thesis to solve unconstrained infinite-horizon total cost optimal control problems for non linear systems is discussed in the next section.

2.3 Unconstrained infinite-horizon optimal control problem

2.3.1 Problem Formulation

The focus of this section is on unconstrained infinite-horizon total cost optimal control problems for nonlinear systems that are affine in the controller and cost functions that are quadratic in the controller. That is, optimal control problems where the system dynamics are of the form

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

where $x \in \Omega \subseteq \mathbb{R}^n$ denotes the system state, $u \in \mathbb{R}^m$ denotes the control input, $f : \Omega \to \mathbb{R}^n$ denotes the drift dynamics, and $g : \Omega \to \mathbb{R}^{n \times m}$ denotes the control effectiveness matrix. To ensure that the control problem is well posed, it is assumed that f and g are Lipschitz continuous on a set Ω that contains the origin as an interior point such that $f(0) = 0$ and $\nabla f(x)$ is continuous and bounded for every bounded $x \in \Omega$. The notation $\phi(t; t_0, x^0, u(\cdot))$ denotes a trajectory of the system in (11) at time t under the control signal u with the initial condition $x^0 \in \Omega$ and initial time $t_0 \in \mathbb{R}_{\geq 0}$.

The cost functional is of the form

$$J\left(t_0, x^0, u\left(\cdot\right)\right) = \int_{t_0}^{\infty} c\left(x\left(\tau; t_0, x^0, u\left(\cdot\right)\right), u\left(\tau\right)\right) \mathrm{d}\tau, \tag{12}$$

where the local cost $c : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ is defined as

$$c\left(x, u\right) \triangleq Q\left(x\right) + u^T R u, \tag{13}$$

where state penalty function, $Q : \mathbb{R}^n \to \mathbb{R}$ is a positive definite function, and control penalty matrix (or, reward), $R \in \mathbb{R}^{m \times m}$ is a symmetric positive definite matrix.

To ensure that the optimal control problem is well-posed, the minimization problem is constrained to the set of admissible controllers, and the existence of at least one admissible controller is assumed.

Definition 2 *Admissible Control [91]: Given the system (f, g), a control u is defined to be admissible with respect to the state penalty function Q on $\mathbb{R}$, if u is continuous on Ω, $u(0) = 0$, u stabilizes (f, g) on Ω, and $J < \infty$, $\forall x \in \Omega$.*

2.3.2 Exact Solution

If the functions f, g, and Q are stationary (time-invariant) and the time-horizon is infinite, then the optimal control input is a stationary state-feedback policy $u(t) = \zeta(x(t))$ for some function $\zeta : \mathbb{R}^n \to \mathbb{R}^m$ [140].

Definition 3 *Let $f : S \to \mathbb{R}$ be a real-valued function. Let f be bounded below on S.*

The infimum of f on S is defined as $\inf_{x \in S} f(x) \coloneqq k \in \mathbb{R}$ such that: (1): $\forall x \in S : k \leq f(x)$, (2): $\forall \epsilon \in \mathbb{R} > 0 : \exists x \in S : f(x) < k + \epsilon$.

The optimal value function $V^* : \mathbb{R}^n \to \mathbb{R}_{\geq 0}$ can be expressed as

$$V^*(x) := \inf_{u(\cdot) \in \mathcal{U}_{[t,\infty)}} \int_t^\infty c(\phi(\tau, x, u_{[t,\tau)}(\cdot)), u(\cdot)) d\tau, \tag{14}$$

for all $x \in \Omega$, where u_I and $\mathcal{U}_I$ are obtained by restricting the domains of u and functions in $\mathcal{U}_I$ to the interval $I \subseteq \mathbb{R}$, respectively. Assuming that an optimal controller exists, let the optimal value function, denoted by $V^* : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, be defined as

$$V^*(x) := \min_{u(\cdot) \in \mathcal{U}_{[t,\infty)}} \int_t^\infty c(\phi(\tau, x, u_{[t,\tau)}(\cdot)), u(\cdot)) d\tau, \tag{15}$$

[44, theorem 1.5] shows that for a nonlinear system described by (11), $V^*(x) \in C^1(\mathbb{R}^n, \mathbb{R})$ is the optimal value function corresponding to the cost functional (12) if and only if it satisfies the Hamilton-Jacobi-Bellman equation

$$\min_{u \in \mathbb{R}^m} \Big(\nabla V(x) \left(f(x) + g(x)u \right) + Q(x) + u^T R u \Big) = 0, \tag{16}$$

where $\nabla(\cdot)$ denotes the derivative of $(\cdot)$ with respect to its first argument with the boundary condition $V(0) = 0$. Provided the HJB in (16) admits a continuously differentiable solution, it constitutes a necessary and sufficient condition for optimality, i.e., if the optimal value function in (14) is continuously differentiable, then it is the unique solution to the HJB in (16) [141]. The optimal control policy $u^* : \mathbb{R}^n \to \mathbb{R}^m$ can be determined from (16) as [44]

$$u^*(x) = -\frac{1}{2} R^{-1} g^T(x) \left(\nabla V^*(x) \right)^T, \quad \forall x \in \Omega. \tag{17}$$

The HJB in (16) can be expressed in the open-loop form as

$$\nabla V^*(x) \left(f(x) + g(x)u^* \right) + Q(x) + u^{*T} R u^* = 0, \quad \forall x \in \Omega. \tag{18}$$

Using (17) in (16) can be expressed in the closed-loop

$$\nabla V^*(x) f(x) - \frac{1}{4}\nabla V^*(x) R^{-1} g^T(x) \left(\nabla V^*(x)\right)^T + Q(x) = 0, \quad \forall x \in \Omega. \tag{19}$$

The optimal policy can now be obtained using (17) if the HJB in (19) can be solved for the optimal value function V^*.

2.3.3 Value Function Approximation

In general, an analytical solution of the HJB equation is infeasible; hence, an approximate solution is sought. The actor-critic (also known as adaptive-critic) architecture is one of the most widely used architectures to implement generalized policy iteration algorithms [28,36,42,96–100]. The actor can learn to directly minimize the estimated cost-to-go, where the estimate of the cost-to-go is obtained by the critic. In an approximate actor-critic-based solution, the optimal value function V^* is replaced by a parametric estimate $\hat{V}\left(x, \hat{W}_c\right)$. and the optimal policy u^* by a parametric estimate $\hat{u}\left(x, \hat{W}_a\right)$ where $\hat{W}_c \in \mathbb{R}^L$ and $\hat{W}_a \in \mathbb{R}^L$ denote vectors of estimates of the ideal parameters. The objective of the critic is to learn the parameters $\hat{W}_c$, and the objective of the actor is to learn the parameters $\hat{W}_a$. Substituting the estimates $\hat{V}$ and $\hat{u}$ for V^* and u^* in (18), respectively, a residual error $\delta : \mathbb{R}^n \times \mathbb{R}^L \times \mathbb{R}^L \to \mathbb{R}$, called the Bellman Error, BE, is defined as

$$\delta(x, \hat{W}_c, \hat{W}_a) \coloneqq \nabla \hat{V}(x, \hat{W}_c) \left(f(x) + g(x)\hat{u}(x, \hat{W}_a)\right) + Q(x) + \hat{u}(x, \hat{W}_a)^T R \hat{u}(x, \hat{W}_a). \tag{20}$$

To solve the optimal control problem, the critic aims to find a set of parameters $\hat{W}_c$ and the actor aims to find a set of parameters $\hat{W}_a$ such that

$$\delta(x, \hat{W}_c, \hat{W}_a) = 0, \tag{21}$$

and

$$u^*(x, \hat{W}_a) = -\frac{1}{2} R^{-1} g^T(x) \left(\nabla \hat{V}\left(x, \hat{W}_c\right)\right)^T, \quad \forall x \in \Omega. \tag{22}$$

Due to the lack of an exact basis for value function approximation, an approximate set of parameters that minimizes the BE is pursued. In particular, to ensure uniform approximation of the value function and the policy over an operating domain $\Omega \subset \mathbb{R}^n$, it is desirable to find parameters that minimize the integral error $E_s : \mathbb{R}^L \times \mathbb{R}^L \to \mathbb{R}$ defined as

$$E_s(\hat{W}_c, \hat{W}_a) \coloneqq \int_{x \in \Omega} \delta^2 \left(x, \hat{W}_c, \hat{W}_a \right) \mathrm{d}x. \tag{23}$$

In an online implementation of the deterministic actor-critic method, it is desirable to update the parameter estimates $\hat{W}_c$ and $\hat{W}_a$ online to minimize the instantaneous error $E_s \left(\hat{W}_c(t), \hat{W}_a(t) \right)$ or the cumulative instantaneous error

$$E(t) \coloneqq \int_0^t E_s \left(\hat{W}_c(\tau), \hat{W}_a(\tau) \right) \mathrm{d}\tau, \tag{24}$$

while the system in (11) is being controlled using the control law, $u(t) = \hat{u} \left(x(t), \hat{W}_a(t) \right)$.

2.3.4 RL-based Online Implementation

Exact model knowledge is needed to compute the Bellman error in (20) and the integral error in (23). In addition, computing the integral error in (36) is generally infeasible. In reinforcement learning-based approximate online optimal control, the Hamilton-Jacobi-Bellman equation along with an estimate of the state derivative [125, 142], or an integral form of the Hamilton-Jacobi-Bellman equation [143] is utilized to approximately evaluate the Bellman error along the system trajectory.

The Bellman error, evaluated at a point, provides an indirect measure of the quality of the estimated value function evaluated at that point. Therefore, the unknown value function parameters are updated based on evaluation of the Bellman error along the system trajectory. Such weight update strategies create two challenges for analyzing convergence. The system states need to satisfy the persistence of excitation condition, and the system trajectory needs to visit enough points in the state-space to generate a good approximation of the value function over the entire

domain of operation. These challenges are typically addressed in the related literature [121, 142, 144–151] by adding an exploration signal to the control input to ensure sufficient exploration of the domain of operation. However, no analytical methods exist to compute the appropriate exploration signal when the system dynamics are nonlinear.

For notational brevity, the dependence of all the functions on the system states and time is suppressed in the stability analysis subsections unless required for clarity of exposition.

2.4 Linear-in-the-parameters approximation of the value function

While the critic updates the estimates $\hat{W}_c(\cdot)$, the actor simultaneously updates the parameter estimates $\hat{W}_a(\cdot)$ using a gradient-based approach so that the quantity $\hat{u}\left(x, \hat{W}_a\right) + \frac{1}{2}R^{-1}g^T(x)\left(\nabla\hat{V}\left(x, \hat{W}_c\right)\right)^T$ decreases. The weight updates are performed online and in real-time while the system is being controlled using the control law $u = \hat{u}\left(x, \hat{W}_a\right)$. In general, ensuring stability during the learning process is difficult. The use of two separate sets of parameters to estimate the value function and the policy is actually needed solely to preserve stability during the learning process.

For feasibility of analysis, the optimal value function is approximated using a linear-in-the-parameters approximation

$$\hat{V}\left(x, \hat{W}_c\right) \coloneqq \hat{W}_c^T\sigma(x), \tag{25}$$

where $\sigma : \mathbb{R}^n \to \mathbb{R}^L$ is a continuously differentiable nonlinear activation function such that $\sigma(0) = 0$ and $\nabla\sigma(0) = 0$, and $\hat{W}_c \in \mathbb{R}^L$, where L denotes the number of unknown parameters in the approximation of the value function. Based on (17), the optimal policy is approximated using the linear-in-the-parameters approximation

$$\hat{u}\left(x, \hat{W}_a\right) \coloneqq -\frac{1}{2}R^{-1}g(x)^T\nabla\sigma^T(x)\hat{W}_a. \tag{26}$$

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

$$\dot{\hat{W}}_c = -\eta_c \Gamma \frac{\omega}{\rho} \hat{\delta}_t, \tag{27}$$

$$\dot{\Gamma} = \left(\beta \Gamma - \eta_c \frac{\Gamma \omega \omega^T \Gamma}{\rho^2} \right) \mathbf{1}_{\{\|\Gamma\| \leq \overline{\Gamma}\}}, \tag{28}$$

where $\Gamma : \mathbb{R}_{\geq t_0} \to \mathbb{R}^{L \times L}$ is a time-varying least-squares gain matrix, $\left\|\Gamma\left(t_0\right)\right\| \leq \overline{\Gamma}$, $\omega \coloneqq \nabla \sigma\left(x\right) \dot{x}$, $\rho \coloneqq 1 + \nu \omega^T \Gamma \omega \in \mathbb{R}$, $\nu \in \mathbb{R}$ is a positive constant gain, $\overline{\Gamma} > 0 \in \mathbb{R}$ is a saturation constant, $\beta > 0 \in \mathbb{R}$ is a constant forgetting factor, and $\eta_c > 0 \in \mathbb{R}$ is a constant adaptation gain.

The actor weights are updated based on the subsequent stability analysis as

$$\dot{\hat{W}}_a = -\eta_{a1} \left(\hat{W}_a - \hat{W}_c \right) - \eta_{a2} \hat{W}_a + \frac{\eta_c G_\sigma \hat{W}_a \omega^T}{4\rho} \hat{W}_c, \tag{29}$$

where $\eta_{a1}, \eta_{a2} \in \mathbb{R}$ are positive constant adaptation gains,
$G_\sigma \coloneqq \nabla \sigma\left(x\right) g\left(x\right) R^{-1} g^T\left(x\right) \nabla \sigma^T\left(x\right)$.

The stability analysis indicates that the sufficient exploration condition takes the form of a PE condition that requires the existence of positive constants $\underline{\psi}$ and T such that the regressor vector satisfies

$$\underline{\psi} I_L \geq \int_t^{t+T} \frac{\omega(\tau)\omega(\tau)^T}{\rho(\tau)} \mathrm{d}\tau, \forall t \in \mathbb{R}_{\geq t_0} \tag{30}$$

Let $\tilde{W}_c \coloneqq W - \hat{W}_c$ and $\tilde{W}_a \triangleq W - \hat{W}_a$ denote the vectors of parameter estimation errors, where $W \in \mathbb{R}^L$ denotes the constant vector of ideal parameters. Provided (30) is satisfied, and under sufficient conditions on the learning gains and the constants $\underline{\psi}$ and T, the candidate Lyapunov function

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

can be used to establish convergence of x, $\tilde{W}_c$, and $\tilde{W}_a$ to a neighborhood of zero as $t \to \infty$, when the system in (11) is controlled using the control law

$$u = \hat{u}\left(x, \hat{W}_a\right), \tag{31}$$

and the parameter estimates $\hat{W}_c(\cdot)$ and $\hat{W}_a(\cdot)$ are updated using the update laws in (27) and (29), respectively.

Chapter III

SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING WITH PARAMETRIC UNCERTAINTIES

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

3.1 Problem Formulation

3.1.1 Control objective

Consider a continuous-time affine nonlinear dynamical system

$$\dot{x} = f(x)\theta + g(x)u, \tag{32}$$

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

The objective is to design a controller u for the system in (32) such that starting

from a given feasible initial condition x^0, the trajectories $x(\cdot)$ decay to the origin and satisfy $x_i(t) \in (a_i, A_i), \forall t \geq 0$, where $i = 1, 2, \ldots, n$ and $a_i < 0 < A_i$. While MBRL methods such as those detailed in [44] guarantee stability of the closed-loop with state constraints are typically difficult to establish without extensive trial and error. In the following, a BT is used to guarantee state constraints.

3.1.2 Barrier Transformation

Let the function $b : \mathbb{R} \to \mathbb{R}$, is referred to as barrier function (BF), be defined as

$$b_{(a_i,A_i)}(y_i) \coloneqq \log \frac{A_i(a_i - y_i)}{a_i(A_i - y_i)}, \quad \forall i = 1, 2, \ldots, n, \tag{33}$$

Let define $b_{(a,A)} : \mathbb{R}^n \to \mathbb{R}^n$ as $b_{(a,A)}(x) \coloneqq [b_{(a_1,A_1)}(x_1); \ldots; b_{(a_n,A_n)}(x_n)]$ with $a = [a_1; \ldots; a_n]$ and $A = [A_1; \ldots; A_n]$. Moreover, the inverse of (33) on the interval (a_i, A_i), is given by

$$b^{-1}_{(a_i,A_i)}(y_i) = a_i A_i \frac{e^{y_i} - 1}{a_i e^{y_i} - A_i}. \tag{34}$$

Taking the derivative of ((34)) with respect to y_i yields

$$\frac{\mathrm{d}b^{-1}_{(a_i,A_i)}(y_i)}{\mathrm{d}y_i} = \frac{A_i a_i^2 - a_i A_i^2}{a_i^2 e^{y_i} - 2a_i A_i + A_i^2 e^{-y_i}}. \tag{35}$$

Consider the BF based state transformation

$$s_i \coloneqq b_{(a_i,A_i)}(x_i), \quad x_i = b^{-1}_{(a_i,A_i)}(s_i). \tag{36}$$

In the following derivation, whenever clear from the context, the subscripts a_i and A_i of the BF and its inverse are suppressed for brevity. The time derivative of the transformed state can be computed using the chain rule as $\dot{s}_i = \frac{\dot{x}_i}{\frac{\partial b^{-1}_{(a_i,A_i)}(z_i)}{\partial z}|_{z=s_i}}$ which yields the transformed dynamics

$$\dot{s}_i = \frac{f_i(x)\theta + g_i(x)u}{\frac{\mathrm{d}b^{-1}_{(a_i,A_i)}(z_i)}{dz}|_{z=s_i}} = F_i(s)\theta + G_i(s)u, \tag{37}$$

where

$$F_i(s) = \frac{a_i^2 e^{s_i} - 2a_i A_i + A_i^2 e^{-s_i}}{A_i a_i^2 - a_i A_i^2} f_i\Big([b^{-1}(s_1); \ldots; b^{-1}(s_n)]\Big), \tag{38}$$

$$G_i(s)=\frac{a_i^2 e^{s_i}-2a_i A_i+A_i^2 e^{-s_i}}{A_i a_i^2-a_i A_i^2} g_i\Big([b^{-1}(s_1);...;b^{-1}(s_n)]\Big). \tag{39}$$

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

$$\dot{s} = F(s) + G(s)u = y(s)\theta + G(s)u, \tag{40}$$

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

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

Lemma 3.1.1 *If $t \mapsto \Phi\big(t, b(x^0), \zeta\big)$ is a Carathéodory solution to (40), starting from the initial condition $b(x^0)$, under the feedback policy $(s,t) \mapsto \zeta(s,t)$, and if $t \mapsto \Lambda(t, x^0, \zeta)$ is a solution to (32), starting from the initial condition x^0, under the controller $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$, then $\Lambda(t, x^0, \zeta) = b^{-1}\big(\Phi(t, b(x^0), \zeta)\big)$ for almost all $t \in \mathbb{R}_{\geq 0}$.*

Proof. see Lemma 3.1.1 in Appendix A. ■

It is immediate from Lemma 3.1.1 that if the trajectories of (40) are bounded and decay to a neighborhood of the origin under a feedback policy $(s,t) \mapsto \zeta(s,t)$, then the feedback policy $(x,t) \mapsto \zeta\big(b(x), t\big)$, when applied to the original system in (32), achieves the control objective stated in section (3.1.1). To develop a BT MBRL method that is robust to parametric uncertainties, the following section develops a novel identifier inspired by the filtered concurrent learning (FCL) method presented in [136].

3.2 Parameter Estimation

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

$$\dot{Y} = \begin{cases} y(s), & \left\|Y_f\right\| \leq \overline{Y_f}, \\ 0, & \text{otherwise}, \end{cases} \quad Y(0) = 0, \tag{41}$$

$$\dot{Y}_f = \begin{cases} Y^T Y, & \left\|Y_f\right\| \leq \overline{Y_f}, \\ 0, & \text{otherwise}, \end{cases} \quad Y_f(0) = 0, \tag{42}$$

$$\dot{G}_f = \begin{cases} G(s)u, & \left\|Y_f\right\| \leq \overline{Y_f}, \\ 0, & \text{otherwise}, \end{cases}, \quad G_f(0) = 0, \tag{43}$$

$$\dot{X}_f = \begin{cases} Y^T(s - s^0 - G_f), & \left\|Y_f\right\| \leq \overline{Y_f}, \\ 0, & \text{otherwise}, \end{cases} \quad X_f(0) = 0, \tag{44}$$

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

$$\dot{\hat{\theta}} = \beta_1 Y_f^T (X_f - Y_f \hat{\theta}), \quad \hat{\theta}(0) = \theta^0, \tag{45}$$

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

Equations (40) - (45) constitute a nonsmooth system of differential equations

$$\dot{z} = h(z,u) = \begin{cases} h_1(z,u), & \left\|Y_f\right\| \leq \overline{Y_f}, \\ h_2(z,u), & \text{otherwise}, \end{cases} \tag{46}$$

where $z = [s;\ \text{vec}(Y);\ \text{vec}(Y_f);\ G_f;\ X_f;\ \hat{\theta}]$, $h_1(z,u) = [F(s) + G(s)u;\ \text{vec}(y(s));$ $\text{vec}(Y^T Y); G(s)u;\ Y^T(s - s^o - G_f);\ \beta_1 Y_f^T (X_f - Y_f \hat{\theta})]$, and $h_2(z,u) = [F(s) + G(s)u;$ $0;\ 0;\ 0;\ 0;\ \beta_1 Y_f^T (X_f - Y_f \hat{\theta})]$. Since $\|Y_f\|$ is non-decreasing in time, it can be shown that (46) admits Carathéodory solutions.

Lemma 3.2.1 *If $\|Y_f\|$ is non-decreasing in time then* (46) *admits Carathéodory solutions.*

Proof. see Lemma 3.2.1 in Appendix A. ■

Note that (42), expressed in the integral form

$$Y_f(t) = \int_0^{t_3} Y^T(\tau)Y(\tau)d\tau, \tag{47}$$

where $t_3 \coloneqq \inf_t\{t \geq 0 \quad | \quad \|Y_f(t)\| \leq \overline{Y_f}\}$, along with (44), expressed in the integral form

$$X_f(t)=\int_0^{t_3} Y^T(\tau)\big(s(\tau)-s^0-G_f(\tau)\big)d\tau, \tag{48}$$

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

$$\dot{\tilde{\theta}} = -\beta_1 Y_f^T Y_f \tilde{\theta}. \tag{49}$$

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

Assumption 3.2.1 *There exists a time instance $T > 0$ such that $Y_f(T)$ is full rank.*

Note that the minimum eigenvalue of Y_f is trivially non-decreasing for $t \geq t_3$ since $Y_f(t)$ is constant $\forall t \geq t_3$. For $t_4 \leq t_5 \leq t_3$, $Y_f(t_5) = Y_f(t_4) + \int_{t_4}^{t_5} Y^T(\tau)Y(\tau)d\tau$. Since $Y_f(t_4)$ is positive semidefinite, and so is the integral $\int_{t_4}^{t_5} Y^T(\tau)Y(\tau)d\tau$, we conclude that $\lambda_{\min}(Y_f(t_5)) \geq \lambda_{\min}(Y_f(t_4))$, As a result, $t \mapsto \lambda_{\min}(Y_f(t))$ is non-decreasing.

Therefore, if Assumption 3.2.1 is satisfied at $t = T$, then $Y_f(t)$ is also full rank for all $t \geq T$. Similar to other MBRL methods that rely on system identification ([44, Chapter 4]) the following assumption is needed to ensure boundedness of the state trajectories over the interval $[0, T]$.

Assumption 3.2.2 *A feedback controller $\psi : \mathbb{R}^n \to \mathbb{R}^q$ that keeps the trajectories of* (40) *inside a known bounded set over the interval $[0, T)$, without requiring the knowledge of θ, is available.*

If a feedback controller that satisfies Assumption 3.2.2 is not available, then, under the additional assumption that the trajectories of (40) are exciting over the interval $[0, T)$, such a controller can be learned, online while maintaining system stability, using model-free reinforcement learning techniques such as [142, 150, 152].

Remark 3.2.1 *While the analysis of the developed technique dictates that a different stabilizing controller should be used over the time interval $[0, T)$, typically, similar to the examples from section 3.5.1 and section 3.5.2, the transient response of the developed controller provides sufficient excitation so that T is small (in the examples provided in section 3.5.1 and section 3.5.2, T is the order of 10^{-5} and 10^{-6}, respectively), and the stabilizing controller is not needed in practice.*

3.3 Model-Based Reinforcement Learning

Lemma 3.1.1 implies that if a feedback controller that practically stabilizes the transformed system in (40) is designed, then the same feedback controller, applied to the original system by inverting the BT also achieves the control objective stated in Section 3.1.1. In the following, a controller that practically stabilizes (40) is designed as

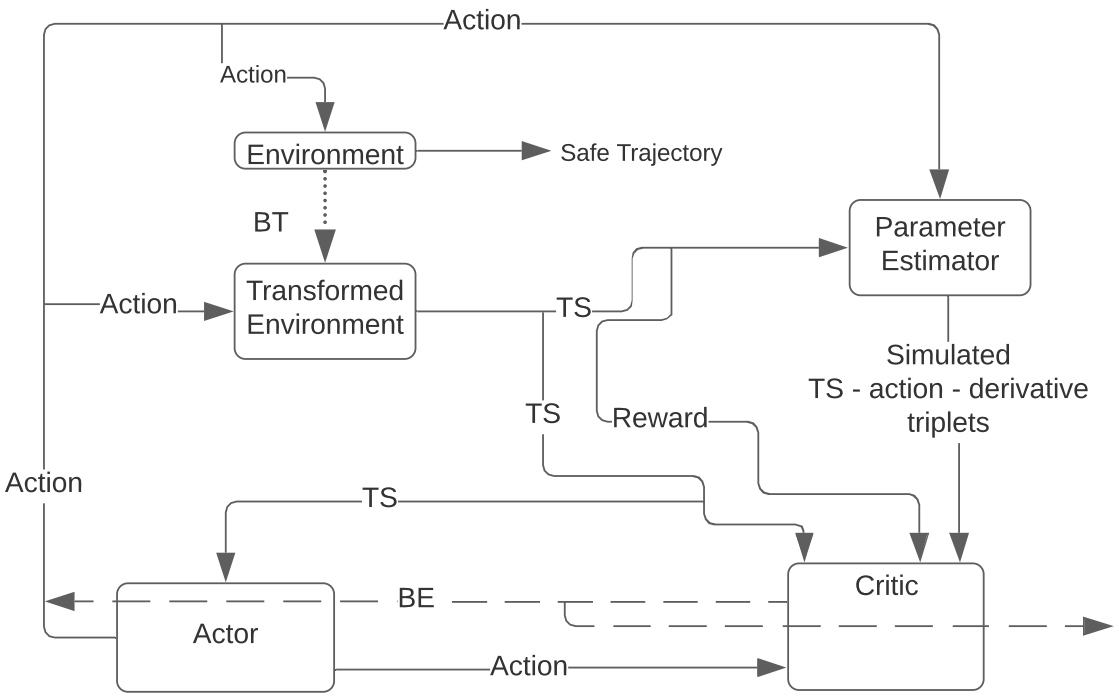

Figure 1: Developed BT MBRL framework (after $Y_f(T)$ is full rank [Assumption 3.2.1]). This control system consists of simulation-based BT-actor-critic-estimator architecture. In addition to the transformed state-action measurements, the critic also utilizes states, actions, and the corresponding state-derivatives to learn the value function. In the figure, BT: Barrier Transformation; TS: Transformed State; BE: Bellman Error. Dotted line means one time initialization, and dashed lines mean learning action.

an estimate of the controller that minimizes the infinite horizon cost[1]

$$J(u(\cdot)) \coloneqq \int_0^\infty r(\phi(\tau, s^0, u(\cdot)), u(\tau))d\tau, \tag{50}$$

over the set $\mathcal{U}$ of piecewise continuous functions $t \mapsto u(t)$, subject to (40), where $\phi(\tau, s^0, u(\cdot))$ denotes the trajectory of ((40)), evaluated at time τ, starting from the state s^0 and under the controller $u(\cdot)$, $r(s, u) \coloneqq s^T Q s + u^T R u$, and $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{q\times q}$ are symmetric positive definite (PD) matrices. Assuming that an optimal controller exists, let the optimal value function, denoted by $V^* : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{R}$, be defined as

$$V^*(s) \coloneqq \min_{u(\cdot)\in\mathcal{U}_{[t,\infty)}} \int_t^\infty r(\phi(\tau, s, u_{[t,\tau)}(\cdot)), u(\cdot))d\tau, \tag{51}$$

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

$$\min_{u\in\mathbb{R}^q} \Big(\nabla_s V(s)\left(F(s)+G(s)u\right)+s^T Q s+u^T R u\Big)=0, \tag{52}$$

where $\nabla_{(\cdot)} \coloneqq \frac{\partial}{\partial(\cdot)}$. Furthermore, the optimal controller is given by the feedback policy $u(t) = u^*(\phi(t, s, u_{[0,t)}))$ where $u^* : \mathbb{R}^n \to \mathbb{R}^q$ defined as

$$u^*(s) \coloneqq -\frac{1}{2} R^{-1} G(s)^T (\nabla_s V^*(s))^T. \tag{53}$$

3.3.1 Value function approximation

Since computation of analytical solutions of the HJB equation is generally infeasible, especially for systems with uncertainty, parametric approximation methods are used to approximate the value function V^* and the optimal policy u^*. The optimal value function is expressed as

$$V^*(s) = W^T \sigma(s) + \epsilon(s), \tag{54}$$

[1]For applications with bounded control inputs, a non-quadratic penalty function similar to [153, Eq. 17] can be incorporated in (50).

where $W \in \mathbb{R}^L$ is an unknown vector of bounded weights, $\sigma : \mathbb{R}^n \to \mathbb{R}^L$ is a vector of continuously differentiable nonlinear activation functions such that $\sigma(0) = 0$ and $\nabla_s \sigma(0) = 0$, $L \in \mathbb{N}$ is the number of basis functions, and $\epsilon : \mathbb{R}^n \to \mathbb{R}$ is the reconstruction error. Exploiting the universal function approximation property of single layer neural networks, it can be concluded that given any compact set $\chi \subset \mathbb{R}^n$ and a positive constant $\bar{\epsilon} \in \mathbb{R}$, there exists a number of basis functions $L \in \mathbb{N}$, and known positive constants $\bar{W}$ and $\bar{\sigma}$ such that $\|W\| \leq \bar{W}$, $\sup_{s\in\chi} \left\|\epsilon(s)\right\| \leq \bar{\epsilon}$, $\sup_{s\in\chi} \left\|\nabla_s \epsilon(s)\right\| \leq \bar{\epsilon}$, $\sup_{s\in\chi} \left\|\sigma(s)\right\| \leq \bar{\sigma}$, and $\sup_{s\in\chi} \left\|\nabla_s \sigma(s)\right\| \leq \bar{\sigma}$ [154]. Using ((52)), a representation of the optimal controller using the same basis as the optimal value function is derived as

$$u^*(s) = -\frac{1}{2} R^{-1} G^T(s) \left(\nabla_s \sigma^T(s) W + \nabla_s \epsilon^T(s)\right). \tag{55}$$

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

$$\hat{V}\left(s, \hat{W}_c\right) \coloneqq \hat{W}_c^T \sigma(s), \tag{56}$$

$$\hat{u}\left(s, \hat{W}_a\right) \coloneqq -\frac{1}{2} R^{-1} G^T(s) \nabla_s \sigma^T(s) \hat{W}_a, \tag{57}$$

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

3.3.2 Bellman Error

Substituting (56) and (57) into (52) results in a residual term, $\hat{\delta} : \mathbb{R}^n \times \mathbb{R}^L \times \mathbb{R}^L \times \mathbb{R}^p \to \mathbb{R}$, which is referred to as Bellman Error (BE), defined as

$$\hat{\delta}(s, \hat{W}_c, \hat{W}_a, \hat{\theta}) \coloneqq \nabla_s \hat{V}(s, \hat{W}_c) \left(y(s)\hat{\theta} + G(s)\hat{u}(s, \hat{W}_a)\right) + \hat{u}(s, \hat{W}_a)^T R \hat{u}(s, \hat{W}_a) + s^T Q s. \tag{58}$$

Traditionally, online RL methods require a persistence of excitation (PE) condition to be able learn the approximate control policy [148, 149, 155]. Guaranteeing PE a priori and verifying PE online are both typically impossible. However, using virtual excitation facilitated by model-based BE extrapolation, stability and convergence of online RL can established under a PE-like condition that, while impossible to guarantee a priori, can be verified online (by monitoring the minimum eigenvalue of a matrix in the subsequent Assumption 3.3.1 [43]. Using the system model, the BE can be evaluated at any arbitrary point in the state space. Virtual excitation can then be implemented by selecting a set of states $\left\{s_k \mid k = 1, \cdots, N\right\}$ and evaluating the BE at this set of states to yield

$$\hat{\delta}_k(s_k, \hat{W}_c, \hat{W}_a, \hat{\theta}) \coloneqq \nabla_{s_k} \hat{V}(s_k, \hat{W}_c) \left(y_k \hat{\theta} + G_k \hat{u}(s_k, \hat{W}_a)\right) + \hat{u}(s_k, \hat{W}_a)^T R \hat{u}(s_k, \hat{W}_a) + s_k^T Q s_k, \quad (59)$$

where, $\nabla_{s_k} \coloneqq \frac{\partial}{\partial s_k}$, $y_k \coloneqq y(s_k)$ and $G_k \coloneqq G(s_k)$. Defining the actor and critic weight estimation errors as $\tilde{W}_c \coloneqq W - \hat{W}_c$ and $\tilde{W}_a \coloneqq W - \hat{W}_a$ and substituting the estimates (54) and (55) into (52), and subtracting from (58) yields the analytical BE that can be expressed in terms of the weight estimation errors as[2]

$$\hat{\delta} = -\omega^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_\sigma \tilde{W}_a - W^T \nabla_s \sigma y \tilde{\theta} + \Delta, \quad (60)$$

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

Similarly, (59) implies that

$$\hat{\delta}_k = -\omega_k^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_{\sigma_k} \tilde{W}_a - W^T \nabla_{s_k} \sigma_k y_k \tilde{\theta} + \Delta_k, \quad (61)$$

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

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

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

3.3.3 Update laws for Actor and Critic weights

The actor and the critic weights are held at their initial values over the interval $[0, T)$ and starting at $t = T$, using the instantaneous BE $\hat{\delta}$ from (58) and extrapolated BEs $\hat{\delta}_k$ from (59), the weights are updated according to

$$\dot{\hat{W}}_c = -k_{c_1}\Gamma\frac{\omega}{\rho}\hat{\delta} - \frac{k_{c_2}}{N}\Gamma\sum_{k=1}^{N}\frac{\omega_k}{\rho_k}\hat{\delta}_k, \tag{62}$$

$$\dot{\Gamma} = \beta\Gamma - k_{c_1}\Gamma\frac{\omega\omega^T}{\rho^2}\Gamma - \frac{k_{c_2}}{N}\Gamma\sum_{k=1}^{N}\frac{\omega_k\omega_k^T}{\rho_k^2}\Gamma, \tag{63}$$

$$\begin{aligned}\dot{\hat{W}}_a = &-k_{a_1}\left(\hat{W}_a - \hat{W}_c\right) - k_{a_2}\hat{W}_a \\ &+ \frac{k_{c_1}G_\sigma^T\hat{W}_a\omega^T}{4\rho}\hat{W}_c + \sum_{k=1}^{N}\frac{k_{c_2}G_{\sigma_k}^T\hat{W}_a\omega_k^T}{4N\rho_k}\hat{W}_c,\end{aligned} \tag{64}$$

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

$$u(t) = \begin{cases}\psi(s(t)), & 0 < t < T, \\ \hat{u}\left(s(t), \hat{W}_a(t)\right), & t \geq T,\end{cases} \tag{65}$$

where the controller ψ was introduced in Assumption 3.2.1. The following verifiable PE-like rank condition is then utilized in the stability analysis.

Assumption 3.3.1 *There exists a constant $\underline{c}_3 > 0$ such that the set of points $\left\{ s_k \in \mathbb{R}^n \mid k = 1, \ldots, N \right\}$ satisfies*

$$\underline{c}_3 I_L \leq \inf_{t \in \mathbb{R}_{\geq T}} \left(\frac{1}{N} \sum_{k=1}^{N} \frac{\omega_k(t)\, \omega_k^T(t)}{\rho_k^2(t)} \right). \tag{66}$$

Since ω_k is a function of the weight estimates $\hat{\theta}$ and $\hat{W}_a$, Assumption 3.3.1 cannot be guaranteed a priori. However, unlike the PE condition, Assumption 3.3.1 can be verified online. Furthermore, since $\lambda_{\min}\left(\sum_{k=1}^{N} \frac{\omega_k(t)\omega_k^T(t)}{\rho_k^2(t)}\right)$ is non-decreasing in the number of samples, N, Assumption 3.3.1 can be met, heuristically, by increasing the number of samples.

3.4 Stability Analysis

Theorem 3.4.1 *Provided Assumptions (3.2.1, 3.2.2, and 3.3.1) hold and the gains are selected large enough based on* (72) - (75)*, then the system state s, weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$, and parameter estimation error $\tilde{\theta}$ are uniformly ultimately bounded.*

Proof. Under Assumption 1, the state trajectories are bounded over the interval $[0, T)$. Over the interval $[T, \infty)$, let $B_r \subset \mathbb{R}^{n+2L+p}$ denote a closed ball with radius r centered at the origin. Let $\chi \coloneqq B_r \cap \mathbb{R}^n$. Let the notation $\overline{\left\|(\cdot)\right\|}$ be defined as $\overline{\|h\|} \coloneqq \sup_{s^o \in \chi} \left\|h\left(s^o\right)\right\|$, for some continuous function $h : \mathbb{R}^n \to \mathbb{R}^m$. To facilitate the analysis, let $\left\{ \varpi_j \in \mathbb{R}_{>0} \mid j = 1, \cdots, 7 \right\}$ be constants such that $\varpi_1 + \varpi_2 + \varpi_3 = 1$, and $\varpi_4 + \varpi_5 + \varpi_6 + \varpi_7 = 1$. Let $\underline{c} \in \mathbb{R}_{>0}$ be a constant defined as

$$\underline{c} \coloneqq \frac{\beta}{2\overline{\Gamma} k_{c2}} + \frac{\underline{c}_3}{2}, \tag{67}$$

k_5 be a positive constant defined as $k_5 \coloneqq (\bar{W} K_{c1} \overline{\nabla_s \sigma} L_y)$. and let $\iota \in \mathbb{R}$ be a positive

constant defined as

$$\iota \triangleq \frac{(k_{c1}+k_{c2})^2 \overline{\left\|\hat{\Delta}\right\|}^2}{4k_{c2}\underline{c}\underline{\varpi}_3} + \frac{1}{4}\overline{\|G_\epsilon\|} + \frac{1}{4\left(k_{a1}+k_{a2}\right)\varpi_6}\left(\frac{1}{2}\overline{W}\overline{\|G_\sigma\|} + \frac{1}{2}\overline{\|\nabla_s \epsilon G^T \nabla_s \sigma^T\|}\right) + \frac{1}{4\left(k_{a1}+k_{a2}\right)\varpi_6}\left(k_{a2}\overline{W} + \frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}^2\overline{\|G_\sigma\|}\right)^2. \quad (68)$$

To facilitate the stability analysis, let $V_L : \mathbb{R}^{n+2L+p} \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a continuously differentiable candidate Lyapunov function defined as

$$V_L\left(Z,t\right) \coloneqq V^*(s) + \frac{1}{2}\tilde{W}_c^T\Gamma^{-1}(t)\tilde{W}_c + \frac{1}{2}\tilde{W}_a^T\tilde{W}_a + V_1(\tilde{\theta}), \quad (69)$$

where V^* is the optimal value function, V_1 was introduced in section 3.2 and $Z \triangleq \left[s;\, \tilde{W}_c;\, \tilde{W}_a;\, \tilde{\theta}\right]$. The update law in (62) ensures that the adaptation gain matrix is bounded such that

$$\underline{\Gamma} \leq \|\Gamma(t)\| \leq \overline{\Gamma}, \forall t \in \mathbb{R}_{\geq T}. \quad (70)$$

Using the fact that V^* and V_1 are positive definite, Lemma 4.3 from [156] yield

$$\underline{v_l}\left(\|Z\|\right) \leq V_L\left(Z,t\right) \leq \overline{v_l}\left(\|Z\|\right), \quad (71)$$

for all $t \in \mathbb{R}_{\geq T}$ and for all $Z \in \mathbb{R}^{n+2L+p}$, where $\underline{v_l}, \overline{v_l} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class $\mathcal{K}$ functions. Let $v_l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a function defined as $v_l\left(\|Z\|\right) \coloneqq \frac{\lambda_{\min}\{Q\}\|s\|^2}{2} + \frac{k_{c2}\underline{c}\underline{\varpi}_1}{2}\left\|\tilde{W}_c\right\|^2 + \frac{(k_{a1}+k_{a2})\varpi_4}{2}\left\|\tilde{W}_a\right\|^2 + \frac{\left\|\tilde{\theta}\right\|^2}{2}$.

The sufficient conditions for ultimate boundedness of Z are derived based on the subsequent stability analysis as

$$\left(k_{c2}\underline{c}\underline{\varpi}_2 - \frac{k_5 r\epsilon}{2}\right)(k_{a1}+k_{a2})\varpi_5 \geq \left(k_{a1} + \frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}\overline{\|G_\sigma\|}\right), \quad (72)$$

$$\left(k_{a1}+k_{a2}\right)\varpi_7 \geq \frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}\overline{\|G_\sigma\|}, \quad (73)$$

$$\lambda_{\min}\{Y_f(T)\} \geq \frac{k_5 r}{2\epsilon} + 1, \quad (74)$$

$$v_l^{-1}(\iota) < \overline{v_l}^{-1}(\underline{v_l}(r)). \quad (75)$$

The bound on the function F and the NN function approximation errors depend on the underlying compact set; hence, ι is a function of r. Even though, in general, ι increases with increasing r, the sufficient condition in (75) can be satisfied provided the points for BE extrapolation are selected such that the constant $\underline{c}$, introduced in (67) is large enough and that the basis for value function approximation are selected such that $\overline{\|\epsilon\|}$ and $\overline{\|\nabla\epsilon\|}$ are small enough.

The orbital derivative of (69) along the trajectories of (40) and (62) - (64) is given by

$$\dot{V}_L = \nabla_s V^* F + \nabla_s V^* G\hat{u} + \tilde{W}_c^T \Gamma^{-1} \dot{\tilde{W}}_c + \frac{1}{2}\tilde{W}_c^T \dot{\Gamma}^{-1} \tilde{W}_c + \tilde{W}_a^T \dot{\tilde{W}}_a + \dot{V}_1. \tag{76}$$

Substituting (62) - (64) in (76) yields

$$\begin{aligned}\dot{V}_L \leq \nabla_s V^* \left(F + Gu^*\right) - \nabla_s V^* G u^* + \nabla_s V^* G\hat{u} - \tilde{W}_c^T \Gamma^{-1} \Bigg(- k_{c1}\Gamma \frac{\omega}{\rho}\hat{\delta} - \frac{1}{N}\Gamma \sum_{k=1}^{N} \frac{k_{c2}\omega_i}{\rho_k}\hat{\delta}_k \Bigg) \\ - \frac{1}{2}\tilde{W}_c^T \Gamma^{-1} \Bigg(\beta\Gamma - k_{c1}(\Gamma \frac{\omega\omega^T}{\rho^2}\Gamma) - \frac{k_{c2}}{N}\Gamma \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2}\Gamma \Bigg)\Gamma^{-1}\tilde{W}_c \\ - \tilde{W}_a^T \Bigg(- k_{a1}(\hat{W}_a - \hat{W}_c) - k_{a2}\hat{W}_a + \big((\frac{k_{c1}\omega}{4\rho}\hat{W}_a^T G_\sigma + \sum_{k=1}^{N} \frac{k_{c2}\omega_k}{4N\rho_k}\hat{W}_a^T G_{\sigma k})^T \hat{W}_c\big) \Bigg) + \dot{V}_1. \end{aligned} \tag{77}$$

Using the inequality $\frac{1}{\rho^2} \leq \frac{1}{\rho}$,

$$\begin{aligned}\dot{V}_L \leq -s^T Q s - \frac{\beta}{2}\tilde{W}_c^T \Gamma^{-1}\tilde{W}_c - \frac{1}{2N}\tilde{W}_c^T \left(\sum_{k=1}^{N} \frac{k_{c2}\omega_k \omega_k^T}{\rho_k} \right) \tilde{W}_c - \left(k_{a1} + k_{a2}\right)\tilde{W}_a^T \tilde{W}_a \\ + \left(\frac{1}{2}W^T G_\sigma + \frac{1}{2}\nabla_s \epsilon G^T \nabla_s \sigma^T + k_{a2} W^T - \frac{1}{4}k_{c1} W^T \frac{\omega}{\rho} W^T G_\sigma - \frac{1}{4}\frac{1}{N} W^T \sum_{k=1}^{N} \frac{k_{c2}\omega_k}{\rho_k} W^T G_{\sigma k} \right) \tilde{W}_a \\ + \tilde{W}_c^T \left(k_{c1}\frac{\omega}{\rho}\Delta + \frac{1}{N}\sum_{k=1}^{N} \frac{k_{c2}\omega_k}{\rho_k}\Delta_k \right) + k_{a1}\tilde{W}_a^T \tilde{W}_c + \frac{1}{4}k_{c1}\tilde{W}_c^T \frac{\omega}{\rho} W^T G_\sigma \tilde{W}_a \\ + \frac{1}{4}\frac{1}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{k_{c2}\omega_k}{\rho_k} W^T G_{\sigma k}\tilde{W}_a + \frac{1}{4}k_{c1} W^T \frac{\omega}{\rho}\tilde{W}_a^T G_\sigma \tilde{W}_a \\ + \frac{1}{4}\frac{1}{N} W^T \sum_{k=1}^{N} \frac{k_{c2}\omega_k}{\rho_k}\tilde{W}_a^T G_{\sigma k}\tilde{W}_a + \frac{1}{4}G_\epsilon + \dot{V}_1 - \tilde{W}_c^T k_{c1}\frac{\omega}{\rho} W^T \nabla_s \sigma y \tilde{\theta} \\ - \frac{1}{N}\tilde{W}_c^T k_{c2} \sum_{k=1}^{N} \frac{\omega_k}{\rho_k} W^T \nabla_s \sigma_k y_k \tilde{\theta}. \end{aligned} \tag{78}$$

So,

$$\dot{V}_L \leq -s^T Q s - \frac{1}{4} W^T G_\sigma W + \frac{1}{2} W^T G_\sigma \tilde{W}_a + \frac{1}{4} G_\epsilon + \frac{1}{2} \tilde{W}_a^T \nabla_s \sigma G_R \nabla_s \epsilon^T$$
$$- \tilde{W}_c^T \Gamma^{-1} \Bigg(- k_{c1} \Gamma \frac{\omega}{\rho} (-\omega^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_\sigma \tilde{W}_a - W^T \nabla_s \sigma y \tilde{\theta} + \frac{1}{2} W^T \nabla_s \sigma G_R \nabla_s \epsilon^T + \frac{1}{4} G_\epsilon - \nabla_s \epsilon F) \Bigg)$$
$$+ \tilde{W}_c^T \Gamma^{-1} \Bigg(\frac{1}{N} \Gamma \sum_{k=1}^{N} \frac{k_{c2} \omega_k}{\rho_k} \big(- \omega_k^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_{\sigma k} \tilde{W}_a - (W^T \nabla_s \sigma_k y_k \tilde{\theta}) + \Delta_k \big) \Bigg)$$
$$- \frac{\beta}{2} \tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{1}{2} k_{c1} \tilde{W}_c^T \frac{\omega \omega^T}{\rho^2} \tilde{W}_c$$
$$+ \frac{1}{2} k_{c2} \tilde{W}_c^T \frac{1}{N} \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \tilde{W}_c + k_{a1} \tilde{W}_a^T \tilde{W}_c - (k_{a1} + k_{a2}) \tilde{W}_a^T \tilde{W}_a + k_{a2} \tilde{W}_a^T W$$
$$- \tilde{W}_a^T \Bigg(\Bigg(\frac{k_{c1} \omega}{4 \rho} \hat{W}_a^T G_\sigma + \sum_{k=1}^{N} \frac{k_{c2} \omega_k}{4 N \rho_k} \hat{W}_a^T G_{\sigma k} \Bigg)^T \hat{W}_c \Bigg) - \tilde{\theta}^T \beta_1^{-1} \beta_1 Y_f^T Y_f \theta. \quad (79)$$

Using Rayleigh-Ritz theorem,

$$\dot{V}_L \leq -s^T Q s - \frac{1}{4} W^T G_\sigma W + \frac{1}{2} W^T G_\sigma \tilde{W}_a + \frac{1}{4} G_\epsilon + \frac{1}{2} \tilde{W}_a^T \nabla_s \sigma G_R \nabla_s \epsilon^T$$
$$- \tilde{W}_c^T \Gamma^{-1} \Bigg(- k_{c1} \Gamma \frac{\omega}{\rho} (-\omega^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_\sigma \tilde{W}_a - W^T \nabla_s \sigma y \tilde{\theta} + \frac{1}{2} W^T \nabla_s \sigma G_R \nabla_s \epsilon^T + \frac{1}{4} G_\epsilon - \nabla_s \epsilon F) \Bigg)$$
$$+ \tilde{W}_c^T \Gamma^{-1} \Bigg(\frac{1}{N} \Gamma \sum_{k=1}^{N} \frac{k_{c2} \omega_k}{\rho_k} \big(- \omega_k^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_{\sigma k} \tilde{W}_a - (W^T \nabla_s \sigma_k y_k \tilde{\theta}) + \Delta_k \big) \Bigg)$$
$$- \frac{\beta}{2} \tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{1}{2} k_{c1} \tilde{W}_c^T \frac{\omega \omega^T}{\rho^2} \tilde{W}_c + \frac{1}{2} k_{c2} \tilde{W}_c^T \frac{1}{N} \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \tilde{W}_c + k_{a1} \tilde{W}_a^T \tilde{W}_c$$
$$- (k_{a1} + k_{a2}) \tilde{W}_a^T \tilde{W}_a + k_{a2} \tilde{W}_a^T W \tilde{W}_a^T \Bigg(\Bigg(\frac{k_{c1} \omega}{4 \rho} \hat{W}_a^T G_\sigma + \sum_{k=1}^{N} \frac{k_{c2} \omega_k}{4 N \rho_k} \hat{W}_a^T G_{\sigma k} \Bigg)^T \hat{W}_c \Bigg) - \lambda_{\min}\{Y_f\} \|\tilde{\theta}\|^2. \quad (80)$$

Using Cauchy-Schwartz inequality,

$$\dot{V}_L \leq -s^T Q s - k_{c2} \underline{c} \left\| \tilde{W}_c \right\|^2 - (k_{a1} + k_{a2}) \left\| \tilde{W}_a \right\|^2$$
$$+ \left(\frac{1}{2} \overline{W} \overline{\|G_\sigma\|} + \frac{1}{2} \left\| \nabla_s \epsilon G^T \nabla_s \sigma^T \right\| + k_{a2} \overline{W} + \frac{1}{4} (k_{c1} + k_{c2}) \overline{W}^2 \overline{\|G_\sigma\|} \right) \left\| \tilde{W}_a \right\|$$
$$+ \left\| \tilde{W}_c \right\| \left((k_{c1} + k_{c2}) \left\| \hat{\delta} \right\| \right) + \tilde{W}_c^T \left(k_{a1} + \frac{1}{4} k_{c1} \frac{\omega}{\rho} W^T G_\sigma + \frac{1}{4} \frac{1}{N} \sum_{k=1}^{N} \frac{k_{c2} \omega_k}{\rho_k} W^T G_{\sigma k} \right) \tilde{W}_a$$
$$+ \frac{1}{4} (k_{c1} + k_{c2}) \overline{W} \overline{\|G_\sigma\|} \left\| \tilde{W}_a \right\|^2 + \frac{1}{4} \overline{\|G_\epsilon\|}$$
$$+ (k_5 r) \left(\frac{\|\tilde{\theta}\|^2}{2 \epsilon} + \frac{\epsilon \|\tilde{W}_c\|^2}{2} \right) - \lambda_{\min}\{Y_f\} \|\tilde{\theta}\|^2. \quad (81)$$

(81) can be re-expressed as

$$\dot{V}_L \leq -s^T Q s - k_{c2}\underline{c}\left(\varpi_1+\varpi_2+\varpi_3\right)\left\|\tilde{W}_c\right\|^2 - \left(k_{a1}+k_{a2}\right)\left(\varpi_4+\varpi_5+\varpi_6+\varpi_7\right)\left\|\tilde{W}_a\right\|^2$$
$$+\left(\frac{1}{2}\overline{W}\overline{\|G_\sigma\|}+\frac{1}{2}\left\|\nabla_s \epsilon G^T \nabla_s \sigma^T\right\|+k_{a2}\overline{W}+\frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}^2\overline{\|G_\sigma\|}\right)\left\|\tilde{W}_a\right\|$$
$$+\left\|\tilde{W}_c\right\|\left(\left(k_{c1}+k_{c2}\right)\left\|\hat{\delta}\right\|\right)+\left(k_{a1}+\frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}\overline{\|G_\sigma\|}\right)\left\|\tilde{W}_a\right\|\left\|\tilde{W}_c\right\|$$
$$+\frac{1}{4}\left(k_{c1}+k_{c2}\right)\overline{W}\overline{\|G_\sigma\|}\left\|\tilde{W}_a\right\|^2+\frac{1}{4}\overline{\|G_\epsilon\|}-\lambda_{\min}\{Y_f\}\|\tilde{\theta}\|^2+\left(k_5 r\right)\left(\frac{\|\tilde{\theta}\|^2}{2\epsilon}+\frac{\epsilon\|\tilde{W}_c\|^2}{2}\right). \quad (82)$$

Provided the gains are selected based on the sufficient conditions in (72), (73), (74) and (75), the Lyapunov derivative can be upper-bounded as

$$\dot{V}_L \leq -\upsilon_l\left(\|Z\|\right), \quad \forall\|Z\| > \upsilon_l^{-1}\left(\iota\right), \quad (83)$$

for all $t \geq T$ and $\forall Z \in B_r$. Using (71), (75), and (83), Theorem 4.18 in [156] can then be invoked to conclude that Z is uniformly ultimately bounded in the sense that $\limsup_{t\to\infty}\left\|Z\left(t\right)\right\| \leq \underline{\upsilon_l}^{-1}\left(\overline{\upsilon_l}\left(\upsilon_l^{-1}\left(\iota\right)\right)\right)$. Furthermore, the concatenated state trajectories are bounded such that $\left\|Z\left(t\right)\right\| \in B_r$ for all $t \in \mathbb{R}_{\geq T}$. Since the estimates $\hat{W}_a$ approximate the ideal weights W, the policy $\hat{u}$ approximates the optimal policy u^*. ■

Using Lemma 3.1.1, it can be concluded that the optimal feedback policy u^*, applied to the original system in (32), achieves the control objective stated in section (3.1.1).

3.5 Simulation

To demonstrate the performance of the developed method for a nonlinear system with an unknown value function, two simulation results, one for a two-state dynamical system (84), and one for a four-state dynamical system (87) corresponding to a two-link planar robot manipulator, are provided.

3.5.1 Two state dynamical system

The dynamical system is given by

$$\dot{x} = f(x)\theta + g(x)u \tag{84}$$

where

$$f(x) = \begin{bmatrix} x_2 & 0 & 0 & 0 \\ 0 & x_1 & x_2 & x_2(\cos(2x_1)+2)^2 \end{bmatrix}, \tag{85}$$

$\theta = [\theta_1; \theta_2; \theta_3; \theta_4]$ and $g(x) = [0; \cos(2x_1) + 2]$. The BT version of the system can be expressed in the form (40) with $G(s) = [0; G_{2_1}]$ and

$$y(s) = \begin{bmatrix} F_{1_1} & 0 & 0 & 0 \\ 0 & F_{2_2} & F_{2_3} & F_{2_4} \end{bmatrix}, \tag{86}$$

where

$$F_{1_1} = \left(\frac{a_1^2 e^{s_1} - 2a_1 A_1 + A_1^2 e^{-s_1}}{A_1 a_1^2 - a_1 A_1^2} \right) x_2,$$

$$F_{2_2} = \left(\frac{a_2^2 e^{s_2} - 2a_2 A_2 + A_2^2 e^{-s_2}}{A_2 a_2^2 - a_2 A_2^2} \right) x_1,$$

$$F_{2_3} = \left(\frac{a_2^2 e^{s_2} - 2a_2 A_2 + A_2^2 e^{-s_2}}{A_2 a_2^2 - a_2 A_2^2} \right) x_2,$$

$$F_{2_4} = \left(\frac{a_2^2 e^{s_2} - 2a_2 A_2 + A_2^2 e^{-s_2}}{A_2 a_2^2 - a_2 A_2^2} \right) x_2(\cos(2x_1)+2)^2,$$

$$G_{2_1} = \left(\frac{a_2^2 e^{s_2} - 2a_2 A_2 + A_2^2 e^{-s_2}}{A_2 a_2^2 - a_2 A_2^2} \right) \cos(2x_1) + 2.$$

The state $x = [x_1 \;\; x_2]^T$ needs to satisfy the constraints, $x_1 \in (-7, 5)$ and $x_2 \in (-5, 7)$. The objective for the controller is to minimize the infinite horizon cost function in (50), with $Q = \text{diag}(10, 10)$ and $R = 0.1$. The basis functions for value function approximation are selected as $\sigma(s) = [s_1^2; s_1 s_2; s_2^2]$. The initial conditions for the system and the initial guesses for the weights and parameters are selected as $x(0) = [-6.5; 6.5]$, $\hat{\theta}(0) = [0; 0; 0; 0]$, $\Gamma(0) = \text{diag}(1, 1, 1)$, and $\hat{W}_a(0) = \hat{W}_c(0) = \left[1/2; 1/2; 1/2\right]$.

The ideal values of the unknown parameters in the system model are $\theta_1 = 1$, $\theta_2 = -1$, $\theta_3 = -0.5$, $\theta_4 = 0.5$ and the ideal values of the actor and the critic weights are unknown. The simulation uses 100 fixed Bellman error extrapolation points in a 4x4 square around the origin of the $s-$coordinate system.

Results for the two state system

As seen from Fig. 2, the system state x stays within the user-specified safe set while converging to the origin. The results in Fig. 3 indicate that the unknown weights for both the actor and critic NNs converge to similar values. As demonstrated in Fig. 4 the parameter estimation errors also converge to the zero.

Since the ideal actor and critic weights are unknown, the estimates cannot be directly compared against the ideal weights. To gauge the quality of the estimates, the trajectory generated by the controller

$$u(t) = \hat{u}\left(s(t), \hat{W}_c^*\right),$$

where $\hat{W}_c^*$ is the final value of the critic weights obtained in Fig. 3, starting from a specific initial condition, is compared against the trajectory obtained using an *offline* numerical solution computed using the GPOPS II optimization software [157]. The total cost, generated by numerically integrating (50), is used as the metric for comparison. The results in Table (1.) indicate that while the two solution techniques generate slightly different trajectories in the phase space (see Fig. 5) the total cost of the trajectories is similar.

Sensitivity Analysis for the two state system

To study the sensitivity of the developed technique to changes in various tuning parameters, a one-at-a-time sensitivity analysis is performed. The parameters k_{c1}, k_{c2}, k_{a1}, k_{a2}, β, and v are selected for the sensitivity analysis. The costs of the trajecto-

Table 1.: Comparison of costs for a single barrier transformed trajectory of (84), obtained using the optimal feedback controller generated via the developed method, and obtained using pseudospectral numerical optimal control software.

Method	Cost
BT MBRL with FCL	71.8422
GPOPS II [157]	72.9005

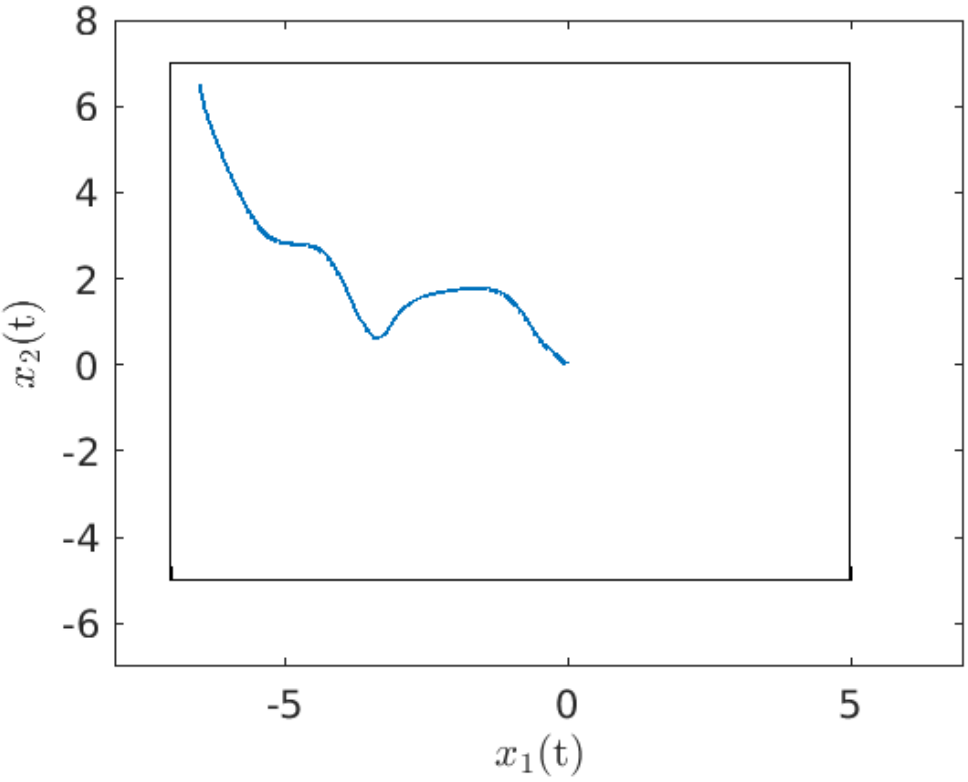

Figure 2: Phase portrait for the two-state dynamical system using MBRL with FCL in the original coordinates. The boxed area represents the user-selected safe set.

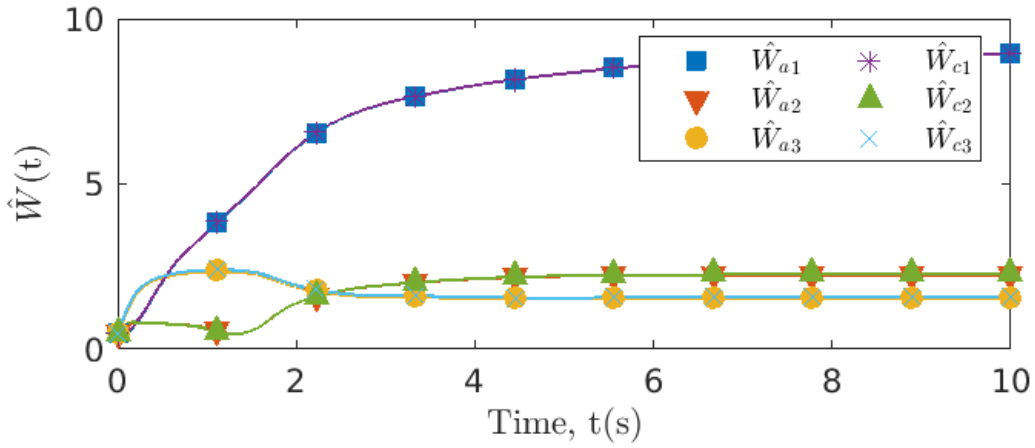

Figure 3: Estimates of the actor and the critic weights under nominal gains for the two-state dynamical system.

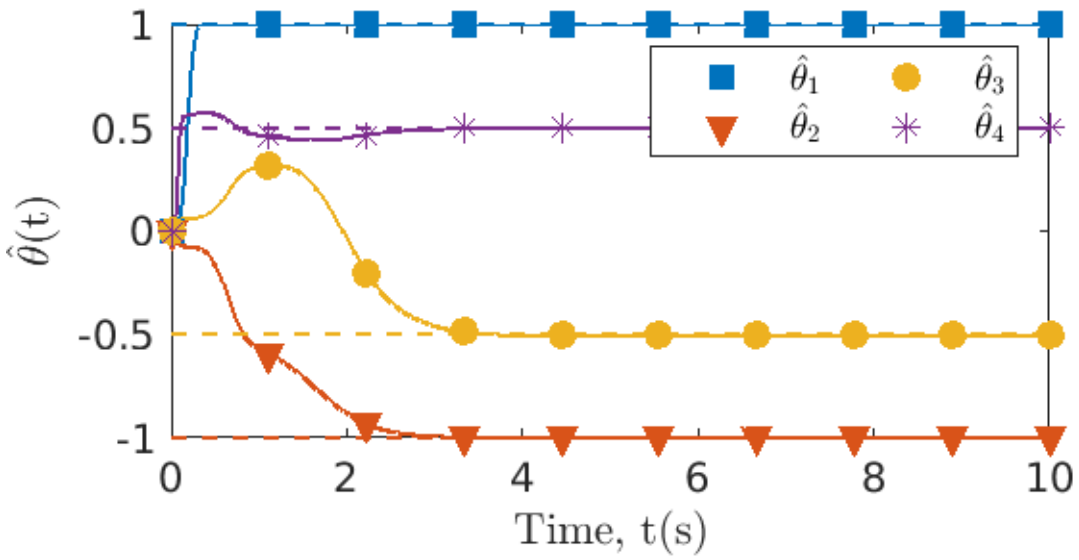

Figure 4: Estimates of the unknown parameters in the system under the nominal gains for the two-state dynamical system. The dash lines in the figure indicates the ideal values of the parameters.

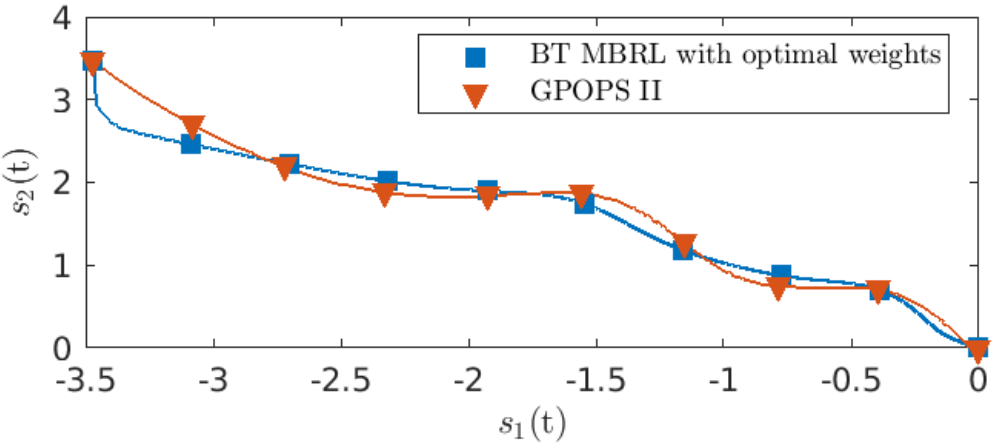

Figure 5: Comparison of the optimal trajectories obtained using GPOPS II and using BT MBRL with FCL and fixed optimal weights for the two-state dynamical system.

ries, under the optimal feedback controller obtained using the developed method, are presented in Table II for 5 different values of each parameter.

The parameters are varied in a neighborhood of the nominal values (selected through trial and error) $k_{c1} = 0.3$, $k_{c2} = 5$, $k_{a1} = 180$, $k_{a2} = 0.0001$, $\beta = .03$, and $\upsilon = 0.5$. The value of β_1 is set to be $\text{diag}(50, 50, 50, 50)$. The results in Table II indicate that the developed method is robust to small changes in the learning gains.

3.5.2 Four state dynamical system

The four-state dynamical system corresponding to a two-link planar robot manipulator is given by

$$\dot{x} = f_1(x) + f_2(x)\theta + g(x)u \tag{87}$$

where

$$f_1(x) = \begin{bmatrix} x_3 \\ x_4 \\ -M^{-1}V_m \begin{bmatrix} x_3 \\ x_4 \end{bmatrix} \end{bmatrix}, \tag{88}$$

Table 2.: Sensitivity Analysis for the two state system

$k_{c_1}=$	0.01	0.05	0.1	0.2	0.3
Cost	72.7174	72.6919	72.5378	72.3019	72.1559
$k_{c_2}=$	2	3	5	10	15
Cost	71.7476	72.3198	72.1559	71.8344	71.7293
$k_{a_1}=$	175	180	250	500	1000
Cost	72.1568	72.1559	72.1384	72.1085	72.0901
$k_{a_2}=$	0.0001	0.0009	0.001	0.005	0.01
Cost	72.1559	72.1559	72.1559	72.1559	72.1559
$\beta=$	0.001	0.005	0.01	0.03	0.04
Cost	72.2141	72.1559	72.1958	72.1559	72.1352
$v=$	0.5	1	10	50	100
Cost	72.1559	72.4054	72.6582	79.1540	81.32

$$f_2(x) = \begin{bmatrix} 0,\ 0,\ 0,\ 0 \\ 0,\ 0,\ 0,\ 0 \\ -[M^{-1}, M^{-1}]D \end{bmatrix}, \quad \theta = \begin{bmatrix} f_{d_1} \\ f_{d_2} \\ f_{s_1} \\ f_{s_1} \end{bmatrix}, \tag{89}$$

$$g(x) = \begin{bmatrix} 0, 0 \\ 0, 0 \\ (M^{-1})^T \end{bmatrix}. \tag{90}$$

where

$$D \coloneqq \operatorname{diag}\left[x_3, x_4, \tanh(x_3), \tanh(x_4)\right], \tag{91}$$

$$M \coloneqq \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \in \mathbb{R}^{2\times 2}, \tag{92}$$

$$V_M \coloneqq \begin{bmatrix} -p_3s_2x_4 & -p_3s_2(x_3 + x_4) \\ p_3s_2x_3 & 0 \end{bmatrix} \in \mathbb{R}^{2\times 2}, \tag{93}$$

with $s_2 = \sin(x_2)$, $c_2 = \cos(x_2)$, $p_1 = 3.473$, $p_2 = 0.196$, $p_3 = 0.242$. The positive constants $f_{d_1}, f_{d_2}, f_{s_1}, f_{s_1} \in \mathbb{R}$ are the unknown parameters. The parameters are selected as $f_{d_1} = 5.3, f_{d_2} = 1.1, f_{s_1} = 8.45, f_{s_1} = 2.35$.

The state $x = [x_1 \quad x_2 \quad x_3 \quad x_4]^T$, that corresponds to angular positions and the angular velocities of the two links needs to satisfy the constraints,
$x_1 \in (-7, 5)$, $x_2 \in (-7, 5)$, $x_3 \in (-5, 7)$ and $x_4 \in (-5, 7)$. The objective for the controller is to minimize the infinite horizon cost function in (50), with $Q = \operatorname{diag}(1, 1, 1, 1)$ and $R = \operatorname{diag}(1, 1)$ while identifying the unknown parameters $\theta \in \mathbb{R}^4$ that correspond to static and dynamic friction coefficients in the two links. The ideal values of the the unknown parameters are $\theta_1 = 5.3$, $\theta_2 = 1.1$, $\theta_3 = 8.45$, and $\theta_4 = 2.35$. The basis functions for value function approximation are selected as $\sigma(s) = [s_1s_3; s_2s_4; s_3s_2; s_4s_1; s_1s_2; s_4s_3; s_1^2; s_2^2; s_3^2; s_4^2]$. The initial conditions for the

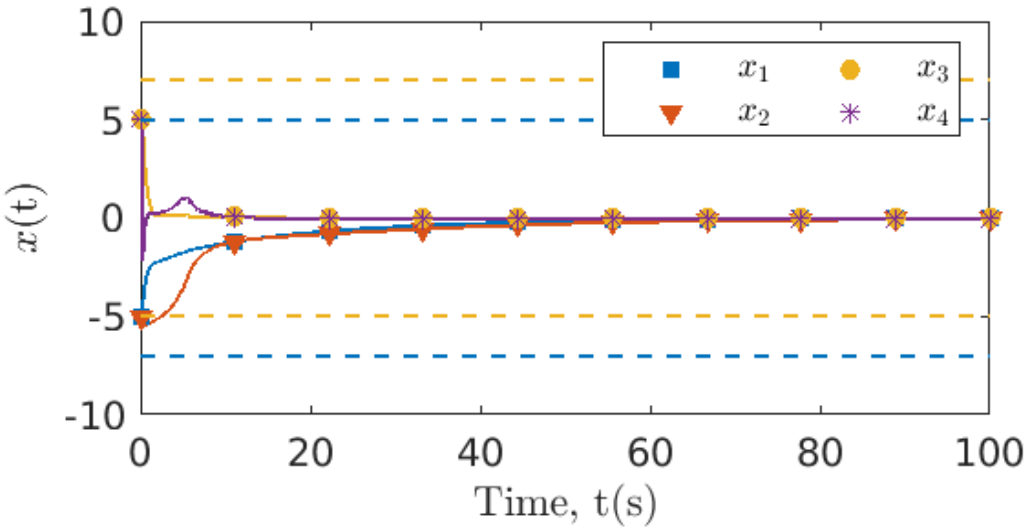

Figure 6: State trajectories for the four-state dynamical system using MBRL with FCL in the original coordinates. The dash lines represent the user-selected safe set.

system and the initial guesses for the weights and parameters are selected as $x(0) = [-5; -5; 5; 5]$, $\hat{\theta}(0) = [5; 5; 5; 5]$, $\Gamma(0) = \text{diag}(10, 10, 10, 10, 10, 10, 10, 10, 10, 10)$, and $\hat{W}_a(0) = \hat{W}_c(0) = [60; 2; 2; 2; 2; 2; 40; 2; 2; 2]$. The ideal values of the actor and the critic weights are unknown. The simulation uses 100 fixed Bellman error extrapolation points in a 4x4 square around the origin of the $s-$coordinate system.

Results for the four state system

As seen from Fig. 6, the system state x stays within the user-specified safe set while converging to the origin. As demonstrated in Fig. 8, the parameter estimations converge to the true values.

A comparison with offline numerical optimal control, similar to the procedure used for the two-state, yields the results in Table (3.) indicate that the two solution techniques generate slightly different trajectories in the state space (see Fig. 9) and the total cost of the trajectories is different. We hypothesize that the difference in costs is due to the basis for value function approximation being unknown.

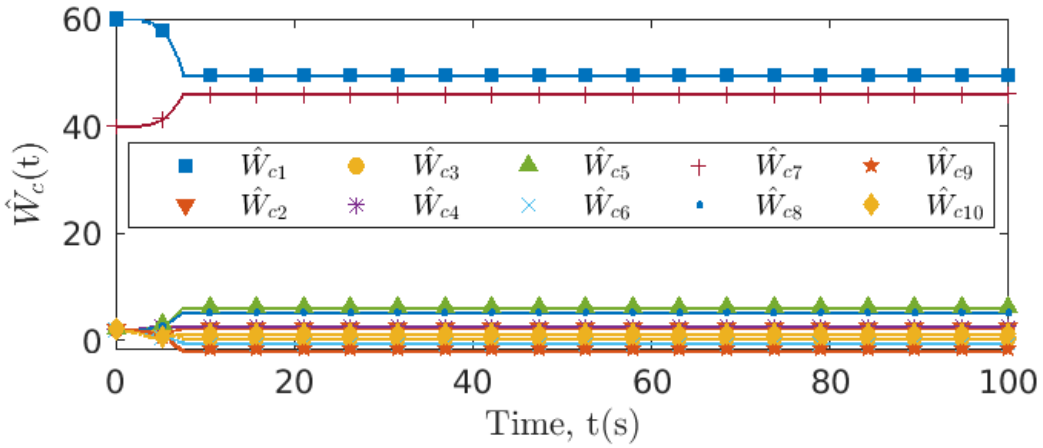

Figure 7: Estimates of the critic weights under nominal gains for the four-state dynamical system.

Table 3.: Costs for a single barrier transformed trajectory of (87), obtained using the developed method, and using pseudospectral numerical optimal control software.

Method	Cost
BT MBRL with FCL	95.1490
GPOPS II	57.8740

In summary, the newly developed method can achieve online optimal feedback control thorough a BT MBRL approach while estimating the value of the unknown parameters in the system dynamics and ensuring safety guarantees in the original coordinates. The following section details a one-at-a-time sensitivity analysis and study the sensitivity of the developed technique to changes in various tuning parameters.

Sensitivity Analysis for the four state system

The parameters k_{c1}, k_{c2}, k_{a1}, k_{a2}, β, and v are selected for the sensitivity analysis. The costs of the trajectories, under the optimal feedback controller obtained using the developed method, are presented in Table II for 5 different values of each parameter. The parameters are varied in a neighborhood of the nominal values (selected through

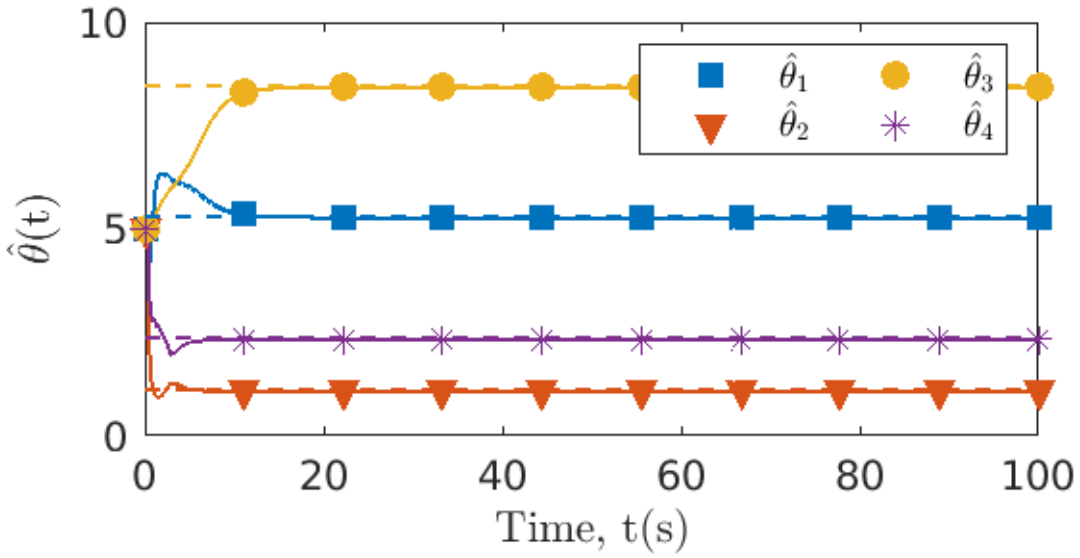

Figure 8: Estimates of the unknown parameters in the system under the nominal gains for the four-state dynamical system. The dash lines in the figure indicates the ideal values of the parameters.

Table 4.: Sensitivity Analysis for the four state system

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

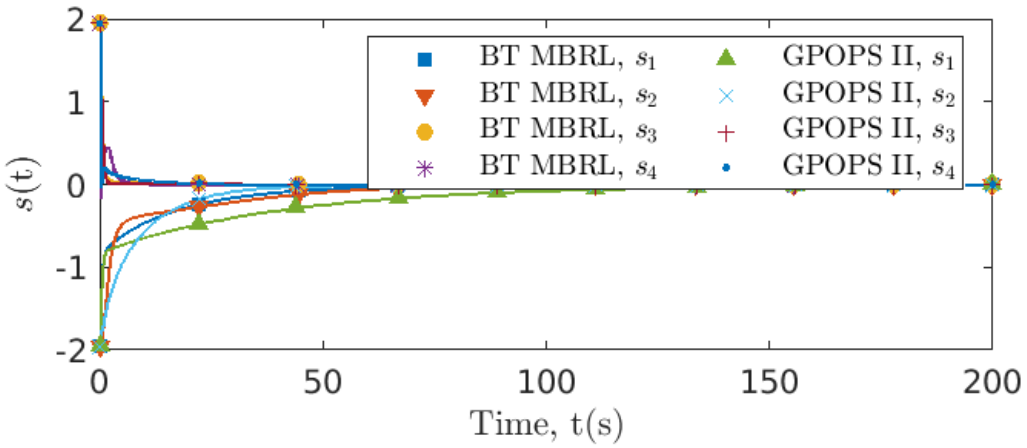

Figure 9: Comparison of the optimal state trajectories obtained using GPOPS II and using BT MBRL with FCL and fixed optimal weights for the four-state dynamical system.

trial and error) $k_{c1} = 0.1$, $k_{c2} = 10$, $k_{a1} = 20$, $k_{a2} = 0.2$, $\beta = 0.8$, and $v = 100$. The value of β_1 is set to be $\text{diag}(100, 100, 100, 100)$. The results in Table (4.) indicate that the developed method is not sensitive to small changes in the learning gains.

Chapter IV

SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING WITH PARTIAL OUTPUT-FEEDBACK

Deployment of unmanned autonomous systems in complex, high-risk tasks provides operational benefits such as accuracy, physical endurance, and so on. Hence, the usage of unmanned autonomous systems has been significantly expanding over the past decades. To realize complex autonomy, techniques that allow autonomous agents to learn to perform tasks, in a provably safe manner, are needed. While recent years have seen prolific progress in the area of safe reinforcement learning [22, 24, 25, 130, 133, 158], most existing techniques require full state feedback. This chapter focuses on the development of a reinforcement learning framework for autonomous systems in continuous time under partial observability, while guaranteeing stability and safety which is a critical, and yet open research question.

4.1 Problem Formulation

We consider the following continuous-time affine nonlinear dynamical system in Brunovsky form

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= f(x) + g(x)u, \end{aligned} \tag{94}$$

where $x_1 \coloneqq [x_{1_1}; \ldots; x_{1_n}] \in \mathbb{R}^n$ and $x_2 \coloneqq [x_{2_1}; \ldots; x_{2_n}] \in \mathbb{R}^n$, $x \coloneqq [x_1; x_2] \in \mathbb{R}^{2n}$ is the system state, $u \in \mathbb{R}^m$ is the control input, and $x_1 \in \mathbb{R}^n$ is the output. The drift dynamics, $f : \mathbb{R}^{2n} \to \mathbb{R}^n$, and control effectiveness, $g : \mathbb{R}^{2n} \to \mathbb{R}^{n \times m}$, are locally

Lipschitz continuous. Let, $\hat{x} \coloneqq [\hat{x}_1; \hat{x}_2]$, $\hat{x}_1 \coloneqq [\hat{x}_{1_1}; \ldots; \hat{x}_{1_n}]$, and $\hat{x}_2 \coloneqq [\hat{x}_{2_1}; \ldots; \hat{x}_{2_n}]$ be the estimates of x, x_1, and x_2 respectively. The notation $[a; b]$ denotes the vector $[a \quad b]^T$.

The objective is to design an adaptive estimator to estimate the state, online, using input-output measurements, and to simultaneously estimate and utilize an optimal controller, u, such that starting from a given feasible initial condition x^0, the trajectories $x(\cdot)$ decay to the origin and satisfy $x_{i_j}(t) \in (a_{i_j}, A_{i_j})$. The notation $(\cdot)_{i_j}$ is used above and in the rest of the manuscript to denote the jth element of the vector $(\cdot)_i$.

Note that the unknown part of the state, x_2 is simply the time derivative of the output, x_1. While the derivative can be computed numerically, state estimators, such as the one designed in the following section, have been shown to be more robust to measurement noise than numerical differentiation. Furthermore, the state estimator designed in the following section allows for rigorous inclusion of state estimation errors in the analysis of the feedback controller.

4.2 State Estimation

In this section, a state estimator inspired by [159] is developed to generate estimates of x. The estimator is given by

$$\dot{\hat{x}} = \begin{bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{bmatrix} = \begin{bmatrix} \hat{x}_2 \\ f(\hat{x}) + g(\hat{x})\, u + \nu_1 \end{bmatrix} \tag{95}$$

where, $\nu_1 = [\nu_{1_1}; \ldots; \nu_{1_n}] \in \mathbb{R}^n$ is a feedback term designed in the following. To design of ν_1 is motivated by the need to establish bounds[1] on state estimation errors in a barrier-transformed coordinate system. To facilitate the design of ν_1, let the state

[1] precisely, (180) in section B.2

estimation errors be defined as

$$\tilde{x}_1 = x_1 - \hat{x}_1,$$

$$\tilde{x}_2 = x_2 - \hat{x}_2. \tag{96}$$

Let the function $b : \mathbb{R} \to \mathbb{R}$, is referred to as barrier function (BF), be defined as

$$b_{(a_{i_j}, A_{i_j})}(y_{i_j}) \coloneqq \log \frac{A_{i_j}(a_{i_j} - y_{i_j})}{a_{i_j}(A_{i_j} - y_{i_j})}, \quad \forall i = 1, 2; \quad \forall j = 1, 2, \ldots, n. \tag{97}$$

Whenever clear from the context, the subscripts a_{i_j} and A_{i_j} of the BF. The feedback component ν_{1_j} is designed as

$$\nu_{1_j} = \frac{(A_{1_j} a_{1_j}^2 - a_{1_j} A_{1_j}^2)}{a_{1_j}^2 e^{b(\hat{x}_{1_j})} - 2a_{1_j} A_{1_j} + A_{1_j}^2 e^{-b(\hat{x}_{1_j})}}$$
$$(\alpha^2 (b(x_{1_j}) - b(\hat{x}_{1_j})) - (k + \alpha + \beta)\, \eta_j), \tag{98}$$

where the signal η_j is added to compensate for the fact that x_{2_j} is not measurable. Based on the subsequent stability analysis, the signal η_j is designed as the output of the dynamic filter

$$\dot{\eta_j} = -\beta_1 \eta_j - k r_j - \alpha \left(\frac{\mathrm{d}}{\mathrm{d}t} \left(b(x_{1_j}) - b(\hat{x}_{1_j}) \right) \right), \tag{99}$$

where $\eta_j\,(T_0) = 0$, α, k, and β are positive constants and the error signal r_j is defined as

$$r_j = \frac{\mathrm{d}}{\mathrm{d}t} \left(b(x_{1_j}) - b(\hat{x}_{1_j}) \right) + \alpha (b(x_{1_j}) - b(\hat{x}_{1_j})) + \eta_j. \tag{100}$$

The signal η_j can be implemented via the integral form,

$$\eta_j(t) = \int_0^t \Big(-(k + \beta_1)\eta_j(\tau) - k\alpha \Big(b(x_{1_j}) - b(\hat{x}_{1_j}) \Big)(\tau) \Big) d\tau - (k + \alpha) \Big(b(x_{1_j})(t) - b(\hat{x}_{1_j})(t) - b(x_{1_j})(0) + b(\hat{x}_{1_j})(0) \Big). \tag{101}$$

While MBRL methods such as those detailed in [44] guarantee stability of the closed-loop with state constraints are typically difficult to establish without extensive trial and error. In the following, a BT is used to guarantee state constraints.

4.3 Barrier Transformation

The inverse of (97) exists on interval (a_{i_j}, A_{i_j}), and is given by

$$b^{-1}_{(a_{i_j},A_{i_j})}(y_{i_j}) = a_{i_j}A_{i_j}\frac{e^{y_{i_j}}-1}{a_{i_j}e^{y_{i_j}}-A_{i_j}}. \tag{102}$$

Consider the BF based state transformation

$$s_{i_j} \coloneqq b_{(a_{i_j},A_{i_j})}(x_{i_j}), \quad x_{i_j} = b^{-1}_{(a_{i_j},A_{i_j})}(s_{i_j}). \tag{103}$$

In the following derivation, whenever clear from the context, the subscripts a_{i_j} and A_{i_j} of the inverse of BF are suppressed for brevity.

To transform the dynamics in (94) using the BT, the time derivative of the transformed state variables can be computed as

$$\dot{s}_1 = H(s), \tag{104}$$

where $H(s) = [H(s_{1_1}, s_{2_1}); \ldots; H(s_{1_n}, s_{2_n})]$, and

$$H(s_{1_j}, s_{2_j}) = \frac{a^2_{1_j}e^{s_{1_j}} - 2a_{1_j}A_{1_j} + A^2_{1_j}e^{-s_{1_j}}}{A_{1_j}a^2_{1_j} - a_{1_j}A^2_{1_j}} b^{-1}(s_{2_j}). \tag{105}$$

Similarly,

$$\dot{s}_2 = F(s) + G(s)u, \tag{106}$$

where $F(s) = [F(s_{1_1}, s_{2_1}); \ldots; F(s_{1_n}, s_{2_n})]$, $G(s) = [G(s_{1_1}, s_{2_1}); \ldots; G(s_{1_n}, s_{2_n})]$,

$$F(s_{1_j}, s_{2_j}) = \left(\frac{a^2_{2_j}e^{s_{2_j}} - 2a_{2_j}A_{2_j} + A^2_{2_j}e^{-s_{2_j}}}{A_{2_j}a^2_{2_j} - a_{2_j}A^2_{2_j}}\right) f([b^{-1}(s_{1_j}), b^{-1}(s_{2_j})]), \tag{107}$$

and

$$G(s_{1_j}, s_{2_j}) = \left(\frac{a^2_{2_j}e^{s_{2_j}} - 2a_{2_j}A_{2_j} + A^2_{2_j}e^{-s_{2_j}}}{A_{2_j}a^2_{2_j} - a_{2_j}A^2_{2_j}}\right) g([b^{-1}(s_{1_j}), b^{-1}(s_{2_j})]). \tag{108}$$

The system (94), in the transformed coordinates, can then be expressed as

$$\dot{s} = [\dot{s}_1; \dot{s}_2] = \begin{bmatrix} H(s) \\ F(s) + G(s)u \end{bmatrix}. \tag{109}$$

As detailed in Lemma 4.3.1 below, design of the BT ensures that the trajectory of (94) and (109) are linked by the BT whenever the initial conditions and the feedback policies are linked by the BT.

Lemma 4.3.1 *If* $t \mapsto \Phi\big(t, b(x^0), \zeta\big)$ *is a Carathéodory solution to* (109)*, starting from the initial condition* $b(x^0)$*, under the feedback policy* $(s,t) \mapsto \zeta(s,t)$*, and if* $t \mapsto \Lambda(t, x^0, \zeta)$ *is a solution to* (94)*, starting from the initial condition* x^0*, under the controller* $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$*, then* $\Lambda(t, x^0, \zeta) = b^{-1}\big(\Phi(t, b(x^0), \zeta)\big)$ *for almost all* $t \in \mathbb{R}_{\geq 0}$*.*

Proof. See Lemma 4.3.1 in Appendix B. ■

To transform the state estimator using the BT, let

$$\hat{s}_{i_j} \coloneqq b(\hat{x}_{i_j}), \quad \text{and} \quad \tilde{s}_{i_j} \coloneqq s_{i_j} - \hat{s}_{i_j}. \tag{110}$$

The state estimator can then be expressed in transformed coordinates as

$$\dot{\hat{s}} = \begin{bmatrix} \dot{\hat{s}}_1 \\ \dot{\hat{s}}_2 \end{bmatrix} = \begin{bmatrix} H(\hat{s}) \\ F(\hat{s}) + G(\hat{s})u + \nu_2(\tilde{s}_1, \eta) \end{bmatrix}, \tag{111}$$

where, $\nu_2 = [\nu_{2_1}; \ldots; \nu_{2_n}]$, $\eta = [\eta_1; \ldots; \eta_n]$, and

$$\nu_{2_j} = \left(\frac{a_{2_j}^2 e^{\hat{s}_{2_j}} - 2a_{2_j}A_{2_j} + A_{2_j}^2 e^{-\hat{s}_{2_j}}}{A_{2_j}a_{2_j}^2 - a_{2_j}A_{2_j}^2} \right) \nu_{1_j}\left([b^{-1}(\tilde{s}_{1_j}), \eta_j\right). \tag{112}$$

As detailed in Lemma 4.3.2 below, the design of the BT ensures that the trajectories of (95), (96), (97), (98), (99), (100) and (111), (112) linked by the BT whenever the underlying state trajectories $x(\cdot)$ and $s(\cdot)$ and the initial conditions $\hat{x}^0$ and $\hat{s}^0$ are linked by the BT.

Lemma 4.3.2 *If* $t \mapsto \Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)$ *is a Carathéodory solution to* (111)*, starting from the initial condition* $b(\hat{x}^0)$ *along the trajectory* $t \mapsto b(x_1(t))$*, and if* $t \mapsto \xi(t; x_1(\cdot), \hat{x}^0)$ *is a solution to* (95)*, starting from the initial condition* $\hat{x}^0$ *along the trajectory* $x_1(\cdot)$*, then* $\xi(t; x_1(\cdot), \hat{x}^0) = b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ *for all* $t \in \mathbb{R}_{\geq 0}$*.*

Proof. See Lemma 4.3.2 in Appendix B. ■

The following section develops a bound on a Lyapunov-like function of the state estimation errors to be utilized in the subsequent stability analysis.

4.4 Optimal Control Formulation

Lemma 4.3.1 implies that if a feedback controller that practically stabilizes the transformed system in (109) is designed, then the same feedback controller, applied to the original system by inverting the BT also achieves the control objective stated in Section 4.1. In the following, a controller that practically stabilizes (109) is designed as an estimate of a controller that minimizes the infinite horizon cost[2]

$$J(u(\cdot)) \coloneqq \int_0^\infty c(\phi(\tau, s^0, u(\cdot)), u(\tau))d\tau, \tag{113}$$

over the set $\mathcal{U}$ of piecewise continuous functions $t \mapsto u(t)$, subject to (109), where $\phi(\tau, s^0, u(\cdot))$ denotes the trajectory of (109), evaluated at time τ, starting from the state s^0, and under the controller $u(\cdot)$. In (113), $c(s, u) \coloneqq Q'(s) + u^T R u$ where $Q'(s) \coloneqq s^T Q s$, $Q'(s) : \mathbb{R}^{2n} \mapsto \mathbb{R}^n$, $Q \in \mathbb{R}^{n \times n}$ and $R \in \mathbb{R}^{m \times m}$ are symmetric positive definite (PD) matrices.

Assumption 4.4.1 *One of the following is true:*

1. Q' *is PD.*

2. Q' *is PD, and* $s_1 \mapsto Q'(s)$ *is PD for all nonzero* $s_2 \in \mathbb{R}^n$.

3. Q' *is PD,* $s_2 \mapsto Q'(s)$ *is PD for all nonzero* $s_1 \in \mathbb{R}^n$ *and* $F(s) \neq 0$ *whenever* $s_1 \neq 0$.

[2]For applications with bounded control inputs, a non-quadratic penalty function similar to [153, Eq. 17] can be incorporated in (113).

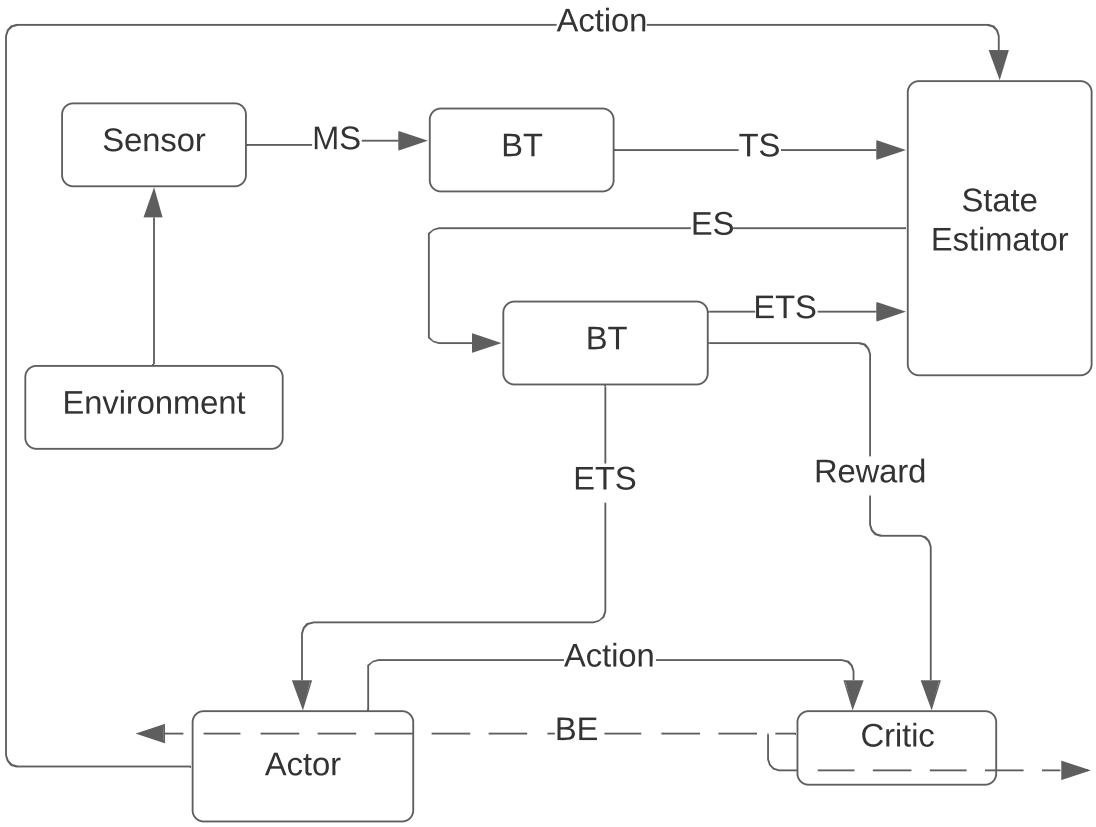

Figure 10: Developed BT MBRL framework. Simulation-based BT-actor-critic-estimator architecture. The critic utilizes Estimated transformed states, actions, and the corresponding Estimated transformed state-derivatives to learn the value function. In the figure, BT: Barrier Transformation; MS: Measured State; TS: Transformed State; ES: Estimated State; ETS: Estimated Transformed State; BE: Bellman Error.

Assuming that an optimal controller exists, let the optimal value function, denoted by $V^* : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{R}$, be defined as

$$V^*(s) := \min_{u(\cdot)\in\mathcal{U}_{[t,\infty)}} \int_t^\infty c(\phi(\tau, s, u_{[t,\tau)}(\cdot)), u(\cdot))d\tau, \tag{114}$$

where u_I and $\mathcal{U}_I$ are obtained by restricting the domains of u and functions in $\mathcal{U}_I$ to the interval $I \subseteq \mathbb{R}$, respectively. Assuming that the optimal value function is continuously differentiable, it can be shown to be the unique PD solution of the Hamilton-Jacobi-Bellman (HJB) equation

$$\min_{u\in\mathbb{R}^q} \Big(V_{s_1}\big(H(s)\big) + V_{s_2}\big(F(s) + G(s)u\big) + s^T Q s + u^T R u \Big) = 0, \tag{115}$$

where $\nabla_{(\cdot)} \coloneqq \frac{\partial}{\partial(\cdot)}$, and $V_{(\cdot)} \coloneqq \nabla_{(\cdot)} V$. Furthermore, the optimal controller is given by the feedback policy $u(t) = u^*(\phi(t, s, u_{[0,t)}))$ where $u^* : \mathbb{R}^n \to \mathbb{R}^m$ defined as

$$u^*(s) := -\frac{1}{2} R^{-1} G(s)^T (\nabla_{s_2} V^*(s))^T. \tag{116}$$

4.4.1 Value function approximation

Since computation of analytical solutions of the HJB equation is generally infeasible, especially for systems with uncertainty, parametric approximation methods are used to approximate the value function V^* and the optimal policy u^*. The optimal value function is expressed as

$$V^*(s) = W^T \sigma(s) + \epsilon(s), \tag{117}$$

where $W \in \mathbb{R}^L$ is an unknown vector of bounded weights, $\sigma : \mathbb{R}^{2n} \to \mathbb{R}^L$ is a vector of continuously differentiable nonlinear activation functions such that $\sigma(0) = 0$ and $\nabla_s \sigma(0) = 0$, $L \in \mathbb{N}$ is the number of basis functions, and $\epsilon : \mathbb{R}^{2n} \to \mathbb{R}$ is the reconstruction error. Exploiting the universal function approximation property of single layer neural networks, it can be concluded that given any compact set[3] $\overline{B}(0, \chi) \subset \mathbb{R}^{2n}$

[3]Note that at this stage, the existence of a compact forward-invariant set that contains trajectories of (109) is not being assumed. The existence of such a set is established in section 4.7, theorem 4.7.1.

and a positive constant $\bar{\epsilon} \in \mathbb{R}$, there exists a number of basis functions $L \in \mathbb{N}$, and known positive constants $\bar{W}$ and $\overline{\sigma}$ such that $\|W\| \leq \bar{W}$, $\sup_{s \in \overline{B}(0,\chi)} \left\|\epsilon\left(s\right)\right\| \leq \bar{\epsilon}$, $\sup_{s \in \overline{B}(0,\chi)} \left\|\nabla_s \epsilon\left(s\right)\right\| \leq \bar{\epsilon}$, $\sup_{s \in \overline{B}(0,\chi)} \left\|\sigma\left(s\right)\right\| \leq \overline{\sigma}$, and $\sup_{s \in \overline{B}(0,\chi)} \left\|\nabla_s \sigma\left(s\right)\right\| \leq \overline{\sigma}$ [154].

Using (115), a representation of the optimal controller using the same basis as the optimal value function is derived as

$$u^*\left(s\right) = -\frac{1}{2} R^{-1} G^T\left(s\right) \left(\nabla_{s_2} \sigma^T\left(s\right) W + \nabla_{s_2} \epsilon^T\left(s\right)\right). \tag{118}$$

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

$$\hat{V}\left(\hat{s}, \hat{W}_c\right) \coloneqq \hat{W}_c^T \sigma\left(\hat{s}\right), \tag{119}$$

$$\hat{u}\left(\hat{s}, \hat{W}_a\right) \coloneqq -\frac{1}{2} R^{-1} G^T\left(\hat{s}\right) \nabla_{\hat{s}_2} \sigma^T\left(\hat{s}\right) \hat{W}_a, \tag{120}$$

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

4.5 Errors bounds for the state estimator

To develop error bounds for the estimation errors, consider the time-derivative of (104) as

$$\ddot{s}_1 = F_2(s) + F_3(s) + G_1(s) u, \tag{121}$$

where $F_2(s_1, s_2) = [F_2(s_{1_1}, s_{2_1}); \ldots; F_2(s_{1_n}, s_{2_n})]$,

$F_3(s_1, s_2) = [F_3(s_{1_1}, s_{2_1}); \ldots; F_3(s_{1_n}, s_{2_n})]$, $G_1(s_1, s_2) = [G_1(s_{1_1}, s_{2_1}); \ldots; G_1(s_{1_n}, s_{2_n})]$,

$$F_2(s_{1_j}, s_{2_j}) = \left(\frac{a_{1_j}^2 e^{s_{1_j}} - A_{1_j}^2 e^{-s_{1_j}}}{A_{1_j} a_{1_j}^2 - a_{1_j} A_{1_j}^2}\right) b^{-1}(s_{2_j}), \tag{122}$$

$$F_3(s_{1_j}, s_{2_j}) = \left(\frac{a_{1_j}^2 e^{s_{1_j}} - 2a_{1_j} A_{1_j} + A_{1_j}^2 e^{-s_{1_j}}}{A_{1_j} a_{1_j}^2 - a_{1_j} A_{1_j}^2}\right) f([b^{-1}(s_{1_j}), b^{-1}(s_{2_j})]), \tag{123}$$

and

$$G_1(s_{1_j}, s_{2_j}) = \left(\frac{a_{1_j}^2 e^{s_{1_j}} - 2a_{1_j}A_{1_j} + A_{1_j}^2 e^{-s_{1_j}}}{A_{1_j}a_{1_j}^2 - a_{1_j}A_{1_j}^2} \right) g([b^{-1}(s_{1_j}), b^{-1}(s_{2_j})]). \tag{124}$$

Similarly, time-derivative of the first state of (111) yields

$$\ddot{\hat{s}}_1 = F_2(\hat{s}) + F_3(\hat{s}) + G_1(\hat{s})u + \nu_3, \tag{125}$$

where $\nu_3 = [\nu_{3_1}; \ldots; \nu_{3_n}]$ and

$$\nu_{3_j} = \left(\frac{a_{1_j}^2 e^{\hat{s}_{1_j}} - 2a_{1_j}A_{1_j} + A_{1_j}^2 e^{-\hat{s}_{1_j}}}{A_{1_j}a_{1_j}^2 - a_{1_j}A_{1_j}^2} \right) \nu_{1_j}([b^{-1}(\tilde{s}_{1_j}), \eta_j). \tag{126}$$

We can rewrite (126) as

$$\nu_{3_j} = (\alpha^2(b(x_{1_j}) - b(\hat{x}_{1_j})) - (k + \alpha + \beta)\,\eta_j) = (\alpha^2 \tilde{s}_{1_j} - (k + \alpha + \beta)\,\eta_j), \tag{127}$$

and (99) as

$$\dot{\eta}_j = -\beta_1 \eta_j - k r_j - \alpha \dot{\tilde{s}}_{1_j}. \tag{128}$$

Using the fact that $\eta = [\eta_1; \ldots; \eta_n]$ which yields

$$\dot{\eta} = -\beta_1 \eta - k r - \alpha(\tilde{H}(s, \hat{s})), \tag{129}$$

where $\tilde{H}(s, \hat{s}) \coloneqq H(s) - H(\hat{s}) = \dot{\tilde{s}}_1$. Furthermore, (100) can be expressed as

$$r = \dot{\tilde{s}}_1 + \alpha \tilde{s}_1 + \eta, \tag{130}$$

where $r = [r_1; \ldots; r_n]$, which yields

$$\dot{\tilde{s}}_1 = r - \alpha \tilde{s}_1 - \eta, \tag{131}$$

The time-derivative of the filtered error signal (130) is given by

$$\dot{r} = \ddot{\tilde{s}}_1 + \alpha \dot{\tilde{s}}_1 + \dot{\eta} = \ddot{s}_1 - \ddot{\hat{s}}_1 + \alpha \dot{\tilde{s}}_1 + \dot{\eta}, \tag{132}$$

which yields

$$\dot{r} = F_2(s) + F_3(s) + G_1(s)\hat{u}(\hat{s}, \hat{W}_a) - F_2(\hat{s}) - F_3(\hat{s}) \\ - G_1(\hat{s})\hat{u}(\hat{s}, \hat{W}_a) - \alpha^2 \tilde{s}_1 + (k + \alpha + \beta_1)\eta \\ + \alpha \dot{\tilde{s}}_1 - \beta_1 \eta - kr - \alpha \dot{\tilde{s}}_1,$$

and can be expressed as

$$\dot{r} = \tilde{F}_2(s, \hat{s}) + \tilde{F}_3(s, \hat{s}) + \tilde{G}_1(s, \hat{s})\hat{u}(\hat{s}, \hat{W}_a) - \alpha^2 \tilde{s}_1 - kr + k\eta + \alpha\eta, \tag{133}$$

where $\tilde{F}_2(s, \hat{s}) \coloneqq F_2(s) - F_2(\hat{s})$, $\tilde{F}_3(s, \hat{s}) \coloneqq F_3(s) - F_3(\hat{s})$, $\tilde{G}_1(s, \hat{s}) \coloneqq G_1(s) - G_1(\hat{s})$. The following lemma 4.5.1 develops a bound on a Lyapunov-like function of the state estimation errors $\tilde{s}_1$, r, and η. The bound is utilized in the subsequent stability analysis in section 4.8.

Lemma 4.5.1 *Let $V_{se} : \mathbb{R}^{3n} \to \mathbb{R}_{\geq 0}$ be a continuously differentiable candidate Lyapunov function defined as $V_{se}(Z_1) \coloneqq \frac{\alpha^2}{2}\tilde{s}_1^T \tilde{s}_1 + \frac{1}{2} r^T r + \frac{1}{2}\eta^T \eta$, where $Z_1 \coloneqq [\tilde{s}_1^T, r^T, \eta^T]$. Provided s, $\hat{s} \in \overline{B}(0, \chi)$ for some $\chi > 0$, the orbital derivative of V_{se} along the trajectories of $\dot{\tilde{s}}_1$, $\dot{r}$, and $\dot{\eta}$, defined as $\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \coloneqq \frac{\partial V_{se}(Z_1, s, \tilde{s}, \tilde{W}_a)}{\partial \tilde{s}_1}(H(s) - H(\hat{s})) + \frac{\partial V_{se}(Z_1, s, \tilde{s}, \tilde{W}_a)}{\partial r}\dot{r} + \frac{\partial V_{se}(Z_1, s, \tilde{s}, \tilde{W}_a)}{\partial \eta}\dot{\eta}$, can be bounded as $\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq -\alpha^3 \|\tilde{s}_1\|^2 - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + \varpi_1 \left(1 + \varpi_4 + \varpi_4 \alpha \|\right) \|r\| \|\tilde{s}_1\| + \varpi_1 \varpi_4 \|r\| \|\eta\| + \varpi_2 \|r\| \|\tilde{W}_a\| + \varpi_3 \|r\|$.*

Proof. See Lemma 4.5.1 in Appendix B. ■

4.6 Model-based Reinforcement Learning

4.6.1 Bellman Error

Substituting (119) and (120) into (115) results in a residual term, $\hat{\delta} : \mathbb{R}^{2n} \times \mathbb{R}^{L} \times \mathbb{R}^{L} \to \mathbb{R}$, which is referred to as Bellman Error (BE), defined as

$$\hat{\delta}(\hat{s}, \hat{W}_c, \hat{W}_a) \coloneqq \hat{V}_{\hat{s}_1}(\hat{s}, \hat{W}_c)\left(H(\hat{s})\right) + \hat{V}_{\hat{s}_2}(\hat{s}, \hat{W}_c)\left(F(\hat{s}) + G(\hat{s})\hat{u}(\hat{s}, \hat{W}_a)\right) + \hat{u}(\hat{s}, \hat{W}_a)^T R \hat{u}(\hat{s}, \hat{W}_a) + \hat{s}^T Q \hat{s}. \quad (134)$$

Traditionally, online RL methods require a persistence of excitation (PE) condition to be able learn the approximate control policy [148, 149, 155]. Guaranteeing PE a priori and verifying PE online are both typically impossible. However, using virtual excitation facilitated by the model, stability and convergence of online RL can established under a PE-like condition that, while impossible to guarantee a priori, can be verified online (by monitoring the minimum eigenvalue of a matrix in the subsequent Assumption 4.8.1) [43]. Using the system model, the BE can be evaluated at any arbitrary point in the state space. Virtual excitation can then be implemented by selecting a set of states $\left\{ s_k \mid k = 1, \cdots, N \right\}$ and evaluating the BE at this set of states to yield

$$\hat{\delta}_k(s_k, \hat{W}_c, \hat{W}_a) \coloneqq \hat{V}_{s_{k_1}}(s_k, \hat{W}_c)\left(H(s_k)\right) + \hat{V}_{s_{k_2}}(s_k, \hat{W}_c)\left(F(s_k) + G(s_k)\hat{u}(s_k, \hat{W}_a)\right) + \hat{u}(s_k, \hat{W}_a)^T R \hat{u}(s_k, \hat{W}_a) + s_k^T Q s_k. \quad (135)$$

Defining the actor and critic weight estimation errors as $\tilde{W}_c \coloneqq W - \hat{W}_c$ and $\tilde{W}_a \coloneqq W - \hat{W}_a$ and substituting the estimates (117) and (118) into (115), and subtracting from (134) yields the analytical BE that can be expressed in terms of the weight

estimation errors as[4]

$$\hat{\delta} = -\omega^T \tilde{W}_c + \frac{1}{4}\tilde{W}_a^T G_\sigma \tilde{W}_a + \Delta, \tag{136}$$

where $\Delta \coloneqq \frac{1}{2}W^T \nabla_{\hat{s}_2}\sigma G_R \nabla_{\hat{s}_2}\epsilon^T + \frac{1}{4}G_\epsilon - (\nabla_{\hat{s}_1}\epsilon H + \nabla_{\hat{s}_2}\epsilon F)$, $G_R \coloneqq GR^{-1}G^T \in \mathbb{R}^{n\times n}$, $G_\epsilon \coloneqq \nabla_{\hat{s}_2}\epsilon G_R \nabla_{\hat{s}_2}\epsilon^T \in \mathbb{R}$, $G_\sigma \coloneqq \nabla_{\hat{s}_2}\sigma G R^{-1} G^T \nabla_{\hat{s}_2}\sigma^T \in \mathbb{R}^{L\times L}$, and $\omega \coloneqq \nabla_{\hat{s}_1}\sigma H + \nabla_{\hat{s}_2}\sigma \left(F + G\hat{u}(\hat{s}, \hat{W}_a)\right) \in \mathbb{R}^L$.

Similarly, (135) implies that

$$\hat{\delta}_k = -\omega_k^T \tilde{W}_c + \frac{1}{4}\tilde{W}_a^T G_{\sigma_k} \tilde{W}_a + \Delta_k, \tag{137}$$

where, $F_k \coloneqq F(s_k)$, $G_k \coloneqq F(s_k)$, $F_{k_1} \coloneqq H(s_k)$, $\epsilon_k \coloneqq \epsilon(s_k)$, $\sigma_k \coloneqq \sigma(s_k)$, $\Delta_k \coloneqq \frac{1}{2}W^T \nabla_{s_{k_2}}\sigma_k G_{R_k} \nabla_{s_{k_2}}\epsilon_k^T + \frac{1}{4}G_{\epsilon_k} - \left(\nabla_{s_{k_1}}\epsilon_k F_{k_1} + \nabla_{s_{k_2}}\epsilon_k F_k\right)$, $G_{\epsilon_k} \coloneqq \nabla_{s_{k_2}}\epsilon_k G_{R_k} \nabla_{s_{k_2}}\epsilon_k^T$, $\omega_k \coloneqq \nabla_{s_{k_1}}\sigma_k F_{k_1} + \nabla_{s_{k_2}}\sigma_k \left(F + G_k\hat{u}(s_k, \hat{W}_a)\right) \in \mathbb{R}^L$, $G_{\sigma_k} \coloneqq \nabla_{s_{k_2}}\sigma_k G_k R^{-1} G_k^T \nabla_{s_{k_2}}\sigma_k^T \in \mathbb{R}^{L\times L}$ and $G_{R_k} \coloneqq G_k R^{-1} G_k^T \in \mathbb{R}^{n\times n}$.

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

4.6.2 Update laws for Actor and Critic weights

Using the instantaneous BE $\hat{\delta}$ from (134) and extrapolated BEs $\hat{\delta}_k$ from (135), the weights are updated according to

$$\dot{\hat{W}}_c = -\frac{k_c}{N}\Gamma \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\hat{\delta}_k, \tag{138}$$

$$\dot{\Gamma} = \beta\Gamma - \frac{k_c}{N}\Gamma \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2}\Gamma, \tag{139}$$

$$\dot{\hat{W}}_a = -k_{a_1}\left(\hat{W}_a - \hat{W}_c\right) - k_{a_2}\hat{W}_a + \sum_{k=1}^{N} \frac{k_{c_2} G_{\sigma_k}^T \hat{W}_a \omega_k^T}{4N\rho_k}\hat{W}_c, \tag{140}$$

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

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

$$u(t) = \hat{u}\left(\hat{s}(t), \hat{W}_a(t)\right), \quad t \geq 0. \tag{141}$$

The Lyapunov function needed to analyze the closed loop system defined by (95), (98), (101), (109), and (138), (139), (140) is constructed using stability properties of (109) under the optimal feedback (116). To that end, the following section analyzes the optimal closed-loop system.

4.7 Stability Under Optimal state Feedback

The following theorem establishes global asymptotic stability of the closed-loop system under optimal state feedback.

Theorem 4.7.1 *If the optimal state feedback controller* (116) *that minimizes the cost function in* (113) *exists and if the corresponding optimal value function is continuously differentiable and radially unbounded, then the origin of closed-loop system*

$$\begin{aligned} \dot{s}_1 &= H(s), \\ \dot{s}_2 &= F(s) + G(s)u^*(s) \end{aligned} \tag{142}$$

is globally asymptotically stable.

Proof. Under the hypothesis of Theorem 4.7.1, the optimal value function is a unique solution of the Hamilton-Jacobi-Bellman equation [160]

$$V_{s_1}^*(s)\,H(s_1, s_2) + V_{s_2}^*(s)\left(F(s) + G(s)\,u^*(s)\right) + c\left(s, u^*(s)\right) = 0, \tag{143}$$

with

$$u^*(s) := -\frac{1}{2} R^{-1} G(s)^T (\nabla_{s_2} V^*(s))^T, \tag{144}$$

Since the solutions of (142) are continuous and the function V^* is positive semidefinite by definition, if $V^*\left(\begin{bmatrix} s_1 \\ s_2 \end{bmatrix}\right) = 0$ for some $s \neq 0$, it can be concluded that $Q\left(\phi\left(t, s, u^*(\cdot)\right)\right) = 0, \forall t \geq 0$, and $u^*\left(\phi\left(t, s, u^*(\cdot)\right)\right) = 0, \forall t \geq 0$. If Assumption 4.4.1-(a) holds then $\phi\left(t, s, u^*(\cdot)\right) = 0, \forall t \geq 0$, which contradicts $s \neq 0$. If Assumption 4.4.1-(b) holds, then $s_1\left(t, s, u^*(\cdot)\right) = 0, \forall t \geq 0$. As a result, $\phi\left(t, s, u^*(\cdot)\right) = 0, \forall t \geq 0$, which contradicts $s \neq 0$. If Assumption 4.4.1-(c) holds, then $s_2\left(t, s, u^*(\cdot)\right) = 0, \forall t \geq 0$. As a result, $s_1\left(t, s, u^*(\cdot)\right) = c_2, \forall t \geq 0$ for some constant $c_2 \in \mathbb{R}^n$. Since $F(s) \neq 0$ if $s_1 \neq 0$, it can be concluded that $c_2 = 0$, which contradicts $s \neq 0$. Hence, $V^*(s)$ cannot be zero for a nonzero s. Furthermore, since $F(0) = 0$, the zero controller is clearly the optimal controller starting from $s = 0$. That is, $V^*(0) = 0$, and as a result, $V^* : \mathbb{R}^{2n} \to \mathbb{R}$ is PD.

Using V^* as a candidate Lyapunov function and using the HJB equation in (143), it can be concluded that

$$V^*_{s_1}(s) H(s) + V^*_{s_2}(s) \left(F(s) + G(s) u^*(s)\right) \leq -Q(s), \tag{145}$$

$\forall s \in \mathbb{R}^{2n}$. If Assumption 4.4.1-(a) holds, then the proof is complete using Lyapunov's direct method. If Assumption 4.4.1-(b) holds, then using the fact that if the output is identically zero then so is the state, the invariance principle [156, Corollary 4.2] can be invoked to complete the proof. If Assumption 4.4.1-(c) holds, then finiteness of the value function everywhere implies that the origin is the only equilibrium point of the closed-loop system. As a result, the invariance principle can be invoked to complete the proof. ■

Using Theorem 4.7.1 and the converse Lyapunov theorem for asymptotic stability [156, Theorem 4.17], the existence of a radially unbounded PD function $\mathcal{V} : \mathbb{R}^{2n} \to \mathbb{R}$

and a PD function $W : \mathbb{R}^{2n} \to \mathbb{R}$ is guaranteed such that

$$\mathcal{V}_{s_1}(s) F(s) + \mathcal{V}_{s_2}(s) \left(F(s) + G(s) u^*(s) \right) \leq -W(s), \tag{146}$$

$\forall s \in \mathbb{R}^{2n}$. The functions $\mathcal{V}$ and W are utilized in the following section to analyze the stability of the output feedback approximate optimal controller.

4.8 Stability Analysis

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

Assumption 4.8.1 *There exists a constant $\underline{c}_1 > 0$ such that the set of points $\left\{ s_k \in \mathbb{R}^n \mid k = 1, \ldots, N \right\}$ satisfies*

$$\underline{c}_1 I_L \leq \inf_{t \in \mathbb{R}_{\geq T}} \left(\frac{1}{N} \sum_{k=1}^{N} \frac{\omega_k(t) \omega_k^T(t)}{\rho_k^2(t)} \right). \tag{147}$$

Since ω_k is a function of the weight estimates $\hat{s}$ and $\hat{W}_a$, Assumption 4.8.1 cannot be guaranteed a priori. However, unlike the PE condition, Assumption 4.8.1 can be verified online. Furthermore, since $\lambda_{\min} \left(\sum_{k=1}^{N} \frac{\omega_k(t) \omega_k^T(t)}{\rho_k^2(t)} \right)$ is non-decreasing in the number of samples, N, Assumption 4.8.1 can be met, heuristically, by increasing the number of samples. It is established in [155, Lemma 1] that under Assumption 4.8.1 and provided $\lambda_{\min} \left\{ \Gamma_0^{-1} \right\} > 0$, the update law in (139) ensures that the least squares gain matrix satisfies

$$\underline{\Gamma} I_L \leq \Gamma(t) \leq \overline{\Gamma} I_L, \tag{148}$$

$\forall t \in \mathbb{R}_{\geq 0}$ and for some $\overline{\Gamma}, \underline{\Gamma} > 0$. Using (137), the orbital derivative of the PD function $\mathcal{V}$ introduced in (146), along the trajectories of (109), under the controller $u = \hat{u} \left(\hat{s}, \hat{W}_a \right)$ be defined as

$$\dot{\mathcal{V}} \left(s, \tilde{s}, \tilde{W}_a \right) \coloneqq \mathcal{V}_{s_1}(s) H(s) + \mathcal{V}_{s_2}(s) \left(F(s) + G(s) \hat{u} \left(\hat{s}, \hat{W}_a \right) \right), \tag{149}$$

adding and subtracting $\mathcal{V}_{s_2}(s)\left(G(s)u^*(s)\right)$,

$$\dot{\mathcal{V}}\left(s,\tilde{s},\tilde{W}_a\right) = \mathcal{V}_{s_1}(s)H(s) + \mathcal{V}_{s_2}(s)\left(F(s)+G(s)u^*(s)\right) - \mathcal{V}_{s_2}(s)\left(G(s)\left(u^*(s)-\hat{u}(s,\hat{W}_a)\right)\right). \tag{150}$$

Using (146), the fact that G is bounded, the basis functions σ are bounded, and that the value function approximation error ϵ and its derivative with respect to $s, \hat{s}$ are bounded on compact sets, the time-derivative can be bounded as

$$\dot{\mathcal{V}}\left(s,\tilde{s},\tilde{W}_a\right) \leq -W(s) + \iota_1\overline{\epsilon} + \iota_2\left\|\tilde{s}\right\|\left\|\tilde{W}_a\right\| + \iota_3\left\|\tilde{W}_a\right\| + \iota_4\left\|\tilde{s}\right\|, \tag{151}$$

for all $\hat{W}_a \in \mathbb{R}^L$, for all $s \in \overline{B}(0,\chi)$, and for all $\hat{s} \in \overline{B}(0,\chi)$, where $\chi \subset \mathbb{R}^{2n}$ is a compact set, $\iota_1, \cdots, \iota_4$ are positive constants, and $\tilde{s} \coloneqq s - \hat{s}$.

Let $\Theta\left(\tilde{W}_c,\tilde{W}_a,t\right) \coloneqq \frac{1}{2}\tilde{W}_c^T\Gamma^{-1}(t)\tilde{W}_c + \frac{1}{2}\tilde{W}_a^T\tilde{W}_a$. The orbital derivative of Θ along the trajectories of (138) - (140) is defined as

$$\dot{\Theta}\left(\tilde{W}_c,\tilde{W}_a,t\right) = \tilde{W}_c^T\Gamma^{-1}\dot{\tilde{W}}_c - \frac{1}{2}\tilde{W}_c^T\Gamma^{-1}\dot{\Gamma}\Gamma^{-1}\tilde{W}_c + \tilde{W}_a^T\dot{\tilde{W}}_a, \tag{152}$$

where $\dot{\tilde{W}}_c = -\dot{\hat{W}}_c$, and $\dot{\tilde{W}}_a = -\dot{\hat{W}}_a$.

Provided the extrapolation states are selected such that $s_k \in \overline{B}(0,\chi)$, $\forall k = 1,\ldots,N$, the orbital derivative in (152) can be bounded [5] as

$$\begin{aligned}\dot{\Theta}\left(\tilde{W}_c,\tilde{W}_a,t\right) \leq\ & -k_c\underline{c}\left\|\tilde{W}_c\right\|^2 - (k_{a1}+k_{a2})\left\|\tilde{W}_a\right\|^2 \\ & + k_c\iota_8\overline{\epsilon}\left\|\tilde{W}_c\right\| + k_c\iota_5\left\|\tilde{W}_a\right\|^2 + (k_c\iota_6+k_{a1})\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| + \left(k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\|,\end{aligned} \tag{153}$$

for all $t \geq 0$, where $\iota_5, \ldots, \iota_8$ are positive constants that are independent of the learning gains, $\overline{W}$ denotes an upper bound on the norm of the ideal weights W, and $\underline{c_3} = \min_{t\geq 0}\lambda_{\min}\left\{\left(\frac{\beta}{2k_c}\Gamma^{-1}(t) + \frac{1}{2N}\sum_{k=1}^{N}\frac{\omega_k\omega_k^T}{\rho_k}\right)\right\}$.

[5] The full derivation is shown in Appendix B.1

Assumption 4.8.1 and (148) guarantee that $\underline{c_3} > 0$. From (182) we get,

$$\dot{V}_{se}\left(Z_1, s, \tilde{s}, \tilde{W}_a\right) \leq -\alpha^3\|\tilde{s}_1\|^2 - (k - \varpi_1\varpi_4)\|r\|^2$$
$$-(\beta_1 - \alpha)\|\eta\|^2 + \varpi_1\left(1 + \varpi_4 + \varpi_4\alpha\right)\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \varpi_2\|r\|\|\tilde{W}_a\| + \varpi_3\|r\|, \tag{154}$$

for all $\hat{W}_a \in \mathbb{R}^L$, for all $s \in \overline{B}(0, \chi)$, and for all $\hat{s} \in \overline{B}(0, \chi)$, where ϖ_2, ϖ_3 are positive constants that are independent of the learning gains and ϖ_1, ϖ_4 are the Lipschitz constants on $\overline{B}(0, \chi)$ for F, and h, respectively.

The candidate Lyapunov function for the overall system is then defined as

$$V_L\left(Z, t\right) \coloneqq \mathcal{V}\left(s\right) + \Theta\left(\tilde{W}_c, \tilde{W}_a, t\right) + V_{se}\left(Z_1\right), \tag{155}$$

where $Z \coloneqq \begin{bmatrix} s^T & \tilde{s}_1^T & r^T & \eta^T & \tilde{W}_c^T & \tilde{W}_a^T \end{bmatrix}^T$. The orbital derivative of the candidate Lyapunov function along the trajectories of (95), (100),(101),(109), (138), (139), (140), under the controller (141), is defined as

$$\dot{V}_L(Z, t) = \dot{\mathcal{V}}\left(s, \tilde{s}, \tilde{W}_a\right) + \dot{V}_{se}\left(Z_1, s, \tilde{s}, \tilde{W}_a\right) + \dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right). \tag{156}$$

Let $\mathcal{C} \subset \mathbb{R}^{5n}$ be a compact set defined as $\mathcal{C} \coloneqq \{(s, \tilde{s}_1, \eta, r) \in \mathbb{R}^{5n} | \|s\| + \|\tilde{s}_1\|(1 + \varpi_4(1+\alpha)) + \varpi_4(\|r\|\| + \|\eta\|) \leq \chi\}$. Using (180), whenever, $(s, \tilde{s}_1, \eta, r) \in \mathcal{C}$, it can be concluded that $s, \hat{s} \in \overline{B}(0, \chi)$. As a result, (151), (153), and (154)
imply that whenever $Z \in \mathcal{C} \times \mathbb{R}^{2L}$, the orbital derivative can be bounded[6] as

$$\begin{aligned}
\dot{V}_L\left(Z, t\right) \leq & -W\left(s\right) - k_c\underline{c_3}\left\|\tilde{W}_c\right\|^2 - \left(k_{a1} + k_{a2} - k_c\iota_5\right)\left\|\tilde{W}_a\right\|^2 - \alpha^3\left\|\tilde{s}_1\right\|^2 \\
& - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + \left(k_c\iota_6 + k_{a1}\right)\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| \\
& + \iota_2(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\|\left\|\tilde{W}_a\right\| + \left(\iota_2\varpi_4 + \varpi_2\right)\|r\|\left\|\tilde{W}_a\right\| + \iota_2\varpi_4\|\eta\|\left\|\tilde{W}_a\right\| \\
& + (1 + \varpi_4 + \varpi_4\alpha)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \iota_4\varpi_4\|\eta\| + (\varpi_3 + \iota_4\varpi_4)\|r\| \\
& + \left(\iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\| + k_c\iota_8\overline{\epsilon}\left\|\tilde{W}_c\right\| + \iota_4(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| + \iota_1\overline{\epsilon},
\end{aligned}$$

[6]The full derivation is shown in Appendix B.2

which yields

$$\dot{V}_L(Z,t) \leq -W(s) - z^T \left(\frac{M + M^T}{2} \right) z + Pz + \iota_1 \bar{\epsilon},$$

where $z \coloneqq \left[\left\| \tilde{W}_c \right\| \quad \left\| \tilde{W}_a \right\| \quad \|\tilde{s}_1\| \quad \|r\| \quad \|\eta\| \right]^T$,

$$P = \left[k_c \iota_8 \bar{\epsilon} \quad \left(k_c \iota_7 + \iota_3 + k_{a2} \overline{W} \right) \quad \iota_4 (1 + \varpi_4 + \varpi_4 \alpha) \quad (\varpi_3 + \iota_4 \varpi_4) \quad \iota_4 \varpi_4 \right],$$

and

$$M = \begin{bmatrix} [k_c \underline{c}_3 & -(k_c \iota_6 + k_{a1}) & 0 & 0 & 0 \\ 0 & (k_{a1} + k_{a2} - k_c \iota_5) & -\iota_2 (1 + \varpi_4 + \varpi_4 \alpha) & -(\iota_2 \varpi_4 + \varpi_2) & -\iota_2 \varpi_4 \\ 0 & 0 & \alpha^3 & -\varpi_1 (1 + \varpi_4 + \varpi_4 \alpha) & 0 \\ 0 & 0 & 0 & (k - \varpi_1 \varpi_4) & -\varpi_1 \varpi_4 \\ 0 & 0 & 0 & 0 & (\beta_1 - \alpha)]. \end{bmatrix},$$

Provided the matrix $M + M^T$ is PD,

$$\dot{V}_L(Z,t) \leq -W(s) - \underline{M} \|z\|^2 + \bar{P} \|z\| + \iota_1 \bar{\epsilon},$$

where $\underline{M} \coloneqq \lambda_{\min} \left\{ \frac{M + M^T}{2} \right\}$. Letting $\underline{M} =: \underline{M}_1 + \underline{M}_2$ and letting $\mathcal{W} : \mathbb{R}^{5n+2L} \to \mathbb{R}$ be defined as $\mathcal{W}(Z) = -W(s) - \underline{M}_1 \|z\|^2$, the time derivative of (155) bounded as

$$\dot{V}_L(Z,t) \leq -\mathcal{W}(Z), \tag{157}$$

$\forall \|z\| > \frac{1}{2} \left(\frac{\overline{P}}{\underline{M}_2} + \sqrt{\frac{\overline{P}^2}{\underline{M}_2^2} + \frac{\iota_1^2 \bar{\epsilon}^2}{\underline{M}_2^2}} \right) = \mu, Z \in \overline{B}(0, \bar{\chi})$, for all $t \geq 0$, and some $\bar{\chi}$ such that $\bar{B}(0, \bar{\chi}) \subseteq \mathcal{C} \times \mathbb{R}^{2L}$.

Using the bound in (148) and the fact that the converse Lyapunov function is guaranteed to be time-independent, radially unbounded, and PD, Lemma 4.3 can be invoked to conclude that

$$\underline{v}\left(\|Z\| \right) \leq V_L(Z,t) \leq \overline{v}\left(\|Z\| \right), \tag{158}$$

for all $t \in \mathbb{R}_{\geq 0}$ and for all $Z \in \mathbb{R}^{5n+2L}$, where $\underline{v}, \overline{v} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class $\mathcal{K}$ functions.

Provided the learning gains, the domain radii χ and $\bar{\chi}$, and the basis functions for function approximation are selected such that $M + M^T$ is PD and $\mu < \overline{v}^{-1}\left(\underline{v}\left(0, \bar{\chi}\right)\right)$, Theorem 4.18 in [156] can be invoked to conclude that Z is uniformly ultimately bounded. Since the estimates W_a approximate the ideal weights W, the policy $\hat{u}$ approximates the optimal policy u^*.

4.9 Simulation

To demonstrate the performance of the developed method for a nonlinear system with an unknown value function, two simulations, one for a two-state dynamical system and one for a four-state dynamical system corresponding to a two-link planar robot manipulator, are provided.

4.9.1 Two state dynamical system

The dynamical system is given by

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = f(x) + g(x)u, \tag{159}$$

where

$$f(x) = -x_1 - \frac{1}{2}x_2\left(1 - \left(\cos\left(2x_1\right) + 2\right)^2\right), \tag{160}$$

$$g(x) = \cos\left(2x_1\right) + 2. \tag{161}$$

Noted that x_1 is the measureable output. Using our estimator, we have the following estimated dynamics

$$\dot{\hat{x}}_1 = \hat{x}_2, \quad \dot{\hat{x}}_2 = f(\hat{x}) + g(\hat{x})u + \nu_1, \tag{162}$$

The state, $x = [x_1 \quad x_2]^T$, and the estimated states $\hat{x} = [\hat{x}_1 \quad \hat{x}_2]^T$ needs to satisfy the constraints, $x_1, \hat{x}_1 \in (a_1, A_1)$ and $x_2, \hat{x}_2 \in (a_2, A_2)$ where a_1 = -7, A_1 = 5, a_2 = -5, A_2 = 7. The objective for the controller is to minimize the infinite horizon cost

Table 5.: Comparison of costs for a single trajectory of barrier transformed (159), obtained using the optimal feedback controller generated via the developed method, and obtained using pseudospectral numerical optimal control software.

Method	Cost
BT MBRL with state estimator	97.25
GPOPS II [157]	86.37

function in (113), with $Q = \text{diag}(10, 10)$ and $R = 0.1$. The basis functions for value function approximation are selected as $\sigma(\hat{s}) = [\hat{s}_1^2; \hat{s}_1\hat{s}_2; \hat{s}_2^2]$. The initial conditions for the state, the estimated state, and the initial guesses for the weights are selected as $x(0) = [-6.5; 6.5], \hat{x}(0) = [-6; 6]$, $\Gamma(0) = \text{diag}(1, 1, 1)$, and $\hat{W}_a(0) = \hat{W}_c(0) = \left[1/2; 1/2; 1/2\right]$ respectively. The ideal values of the actor and the critic weights for the barrier-transformed optimal control problem are unknown. The simulation uses 100 fixed Bellman error extrapolation points in a 4x4 square around the origin of the $s-$coordinate system.

Results for the two state system

Fig.11 indicates that the system state, x, stays within the user-specified safe set while converging to the origin. As seen from Fig. 13, the state estimation errors also converge to the zero. The results in Fig. 12 shows that the unknown weights for both the actor and critic NNs converge to similar values.

As the ideal actor and critic weights are unknown, the estimates cannot be directly compared against the ideal weights. To gauge the quality of the estimates, the trajectory generated by the controller

$$u(t) = \hat{u}\left(\hat{s}(t), \hat{W}_c^*\right),$$

where $\hat{W}_c^*$ is the final value of the critic weights obtained in Fig. 12, starting from

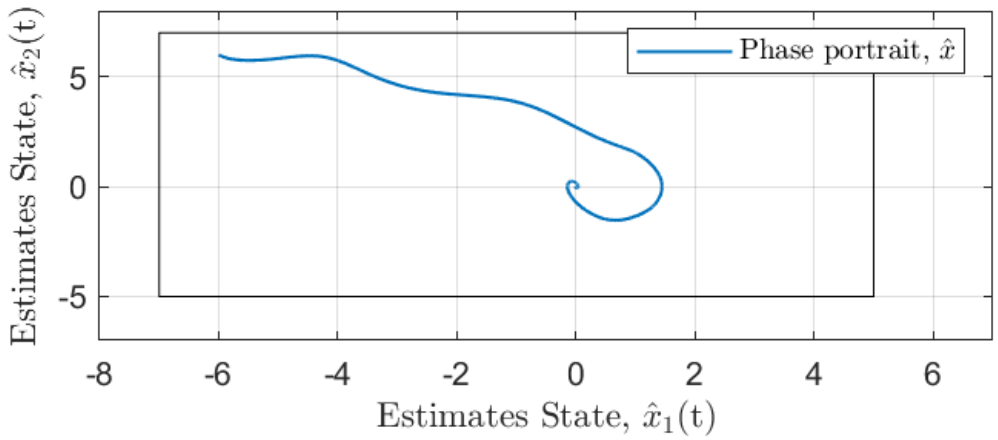

Figure 11: Phase portrait for the two-state dynamical system using MBRL with state estimator in the original coordinates. The boxed area represents the user-selected safe set.

a specific initial condition, and is compared against the trajectory obtained using an *offline* numerical solution computed using the GPOPS II optimization software [157]. The total cost, generated by numerically integrating (113), is used as the metric for comparison. The results in Table 5. indicate that while the two solution techniques generate slightly different trajectories in the phase space (see Fig. 14).

Sensitivity Analysis for the two state system

To study the sensitivity of the developed technique to changes in various tuning gains, a one-at-a-time sensitivity analysis is performed. The gains k, α, β_1, k_c, k_{a1}, k_{a2}, β, and v are selected for the sensitivity analysis. The costs of the trajectories, under the optimal feedback controller obtained using the developed method, are presented in Table 6. for 5 different values of each gain. The gains are varied in a neighborhood of the nominal values (selected through trial and error) $k = 0.0001$, $\alpha = 0.0001$, $\beta_1 = 10$, $k_c = 0.1$, $k_{a1} = 100$, $k_{a2} = 0.1$, $\beta = 5$, and $v = 5$.

The results in Table 6. indicate that the developed method is robust to small changes in the learning gains.

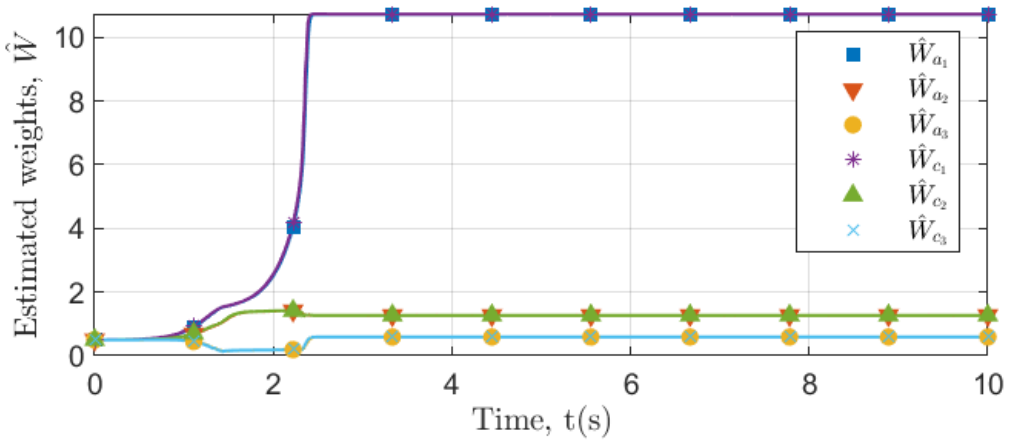

Figure 12: Estimates of the actor and the critic weights under nominal gains for the two-state dynamical system.

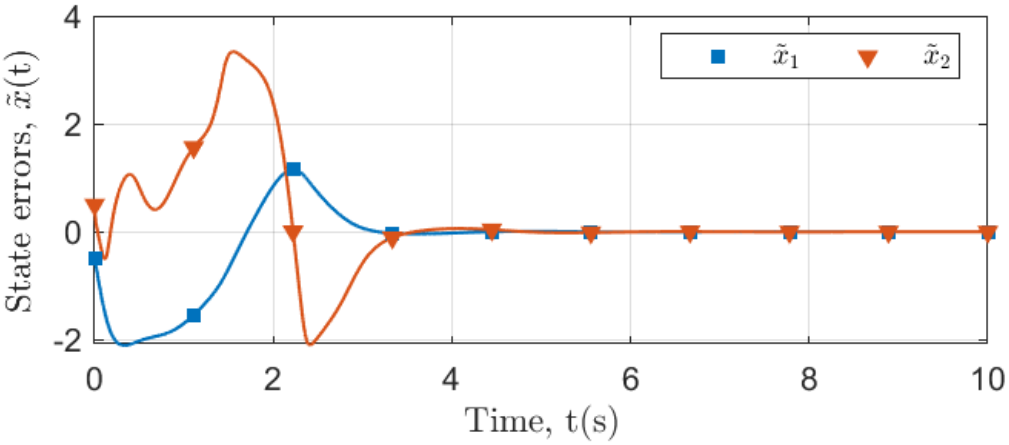

Figure 13: Estimation errors between the original states and the estimated states under nominal gains for the two-state dynamical system.

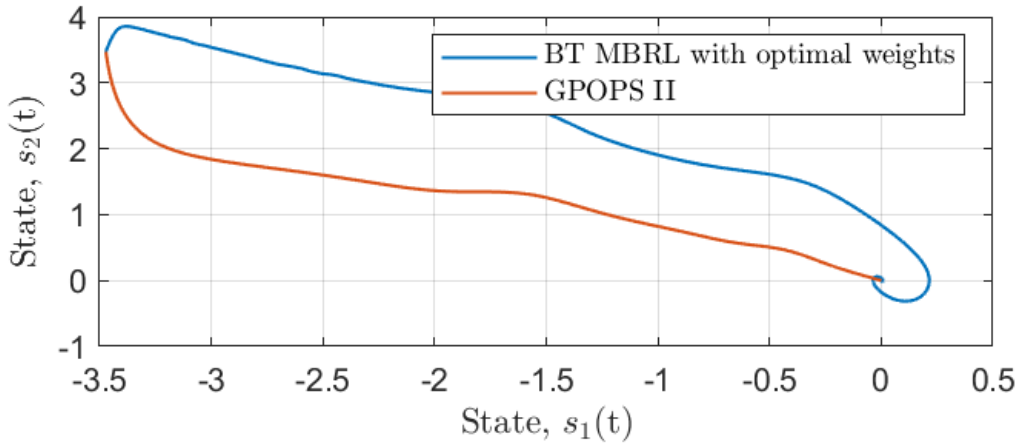

Figure 14: Comparison of the optimal trajectories obtained using GPOPS II and using BT MBRL with fixed optimal weights for the two-state dynamical system.

Table 6.: Sensitivity Analysis for the two state system. The gains are varied in a neighborhood of the nominal values (selected through trial and error) $k = 0.0001$, $\alpha = 0.0001$, $\beta_1 = 10$, $k_c = 0.1$, $k_{a1} = 100$, $k_{a2} = 0.1$, $\beta = 5$, $v = 5$, and NF indicates not feasible.

k_c=	0.001	0.01	0.1	1	10
Cost	97.25	97.25	97.25	97.26	97.38
k_{a_1}=	30	50	100	200	500
Cost	97.26	97.25	97.25	97.25	97.25
k_{a_2}=	0.01	0.05	0.1	0.5	1
Cost	97.25	97.25	97.25	97.25	97.26
β=	1	2	5	10	30
Cost	NF	97.25	97.25	97.25	97.25
v=	0.1	1	5	10	30
Cost	99.06	97.36	97.25	97.25	97.36

4.9.2 Four state dynamical system

The four-state dynamical system corresponding to a two-link planar robot manipulator is given by

$$\dot{x}_1 = x_2, \quad \dot{x}_2 = f(x) + g(x)u, \tag{163}$$

where

$$x_1 = \begin{bmatrix} x_{1_1} \\ x_{1_2} \end{bmatrix}, \quad x_2 = \begin{bmatrix} x_{2_1} \\ x_{2_2}, \end{bmatrix},$$

$$f(x) = -M^{-1}\left(V_M \begin{bmatrix} x_{2_1} \\ x_{2_2} \end{bmatrix} + \begin{bmatrix} f_{d_1} x_{2_1} + f_{s_1} \tanh(x_{2_1}) \\ f_{d_2} x_{2_2} + f_{s_2} \tanh(x_{2_2}) \end{bmatrix} \right),$$

$$g(x) = (M^{-1})^T, \qquad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

$$D \coloneqq \operatorname{diag}\left[x_{2_1}, x_{2_2}, \tanh(x_{2_1}), \tanh(x_{2_2}) \right],$$

$$M \coloneqq \begin{bmatrix} p_1 + 2p_3 c_2 & p_2 + p_3 c_2 \\ p_2 + p_3 c_2 & p_2 \end{bmatrix} \in \mathbb{R}^{2\times 2},$$

$$V_M \coloneqq \begin{bmatrix} -p_3 s_2 x_{2_2} & -p_3 s_2 (x_{2_1} + x_{2_2}) \\ p_3 s_2 x_{2_1} & 0 \end{bmatrix} \in \mathbb{R}^{2\times 2},$$

with $s_2 = \sin(x_{1_2})$, $c_2 = \cos(x_{1_2})$, $p_1 = 3.473$, $p_2 = 0.196$, $p_3 = 0.242$. The parameters are selected as $f_{d_1} = 5.3, f_{d_2} = 1.1, f_{s_1} = 8.45, f_{s_1} = 2.35$.

Noted that x_1 is the measureable output. Using our estimator, we have the following estimated dynamics

$$\dot{\hat{x}}_1 = \hat{x}_2, \quad \dot{\hat{x}}_2 = f(\hat{x}) + g(\hat{x})\hat{u} + \nu_1, \tag{164}$$

Table 7.: Costs for a single barrier transformed trajectory of (163), obtained using the developed method, and using pseudospectral numerical optimal control software.

Method	Cost
BT MBRL with state estimator	11.226
GPOPS II	6.858

The state $x = [x_{1_1} \quad x_{1_2} \quad x_{2_1} \quad x_{2_2}]^T$ corresponds to angular positions and the angular velocities of the two links; $\hat{x} = [\hat{x}_{1_1} \quad \hat{x}_{1_2} \quad \hat{x}_{2_1} \quad \hat{x}_{2_2}]^T$ corresponds to the estimated angular positions and the estimated angular velocities of the two links. Now, x, $\hat{x}$ need to satisfy the constraints, $x_{1_1}, \hat{x}_{1_1} \in (-1, 1)$; $x_{1_2}, \hat{x}_{1_2} \in (-1, 1)$; $x_{2_1}, \hat{x}_{2_1} \in (-2, 2)$; $x_{2_2}, \hat{x}_{2_2} \in (-2, 2)$. The objective for the controller is to minimize the infinite horizon cost function in (113), with $Q = \text{diag}(10, 10, 1, 1)$ and $R = \text{diag}(1, 1)$. The basis functions for value function approximation are selected as $\sigma(\hat{s}) = [\hat{s}_{1_1}\hat{s}_{2_1}; \hat{s}_{1_2}\hat{s}_{2_2}; \hat{s}_{2_1}\hat{s}_{1_2}; \hat{s}_{2_2}\hat{s}_{1_1}; \hat{s}_{1_1}\hat{s}_{1_2}; \hat{s}_{2_2}\hat{s}_{2_1}; \hat{s}_{1_1}^2; \hat{s}_{1_2}^2; \hat{s}_{2_1}^2; \hat{s}_{2_2}^2]$.

The initial conditions for our state, our estimated states, and the initial guesses for the weights are selected as $x(0) = [-0.5; -0.5; 1; 1]$, $\hat{x}(0) = [-0.5; -0.5; 1.1; 1.1]$, $\Gamma(0) = \text{diag}(10, 10, 10, 10, 10, 10, 10, 10, 10, 10)$, and $\hat{W}_a(0) = [5; 15; 0; 0; 10; 2; 15; 5; 2; 2]$, $\hat{W}_c(0) = [15; 15; 0; 0; 10; 2; 15; 5; 2; 2]$. The ideal values of the actor and the critic weights are unknown. The simulation uses 625 fixed Bellman error extrapolation points in a 4x4 square around the origin of the $s-$coordinate system.

Results for the four state system

As seen from Fig. 15, the system estimated state x stays within the user-specified safe set while converging to the origin. As demonstrated in Fig. 17 the state estimations converge to the true values.

A comparison with offline numerical optimal control, similar to the procedure used for the two-state, yields the results in Table 7. indicate that the two solution tech-

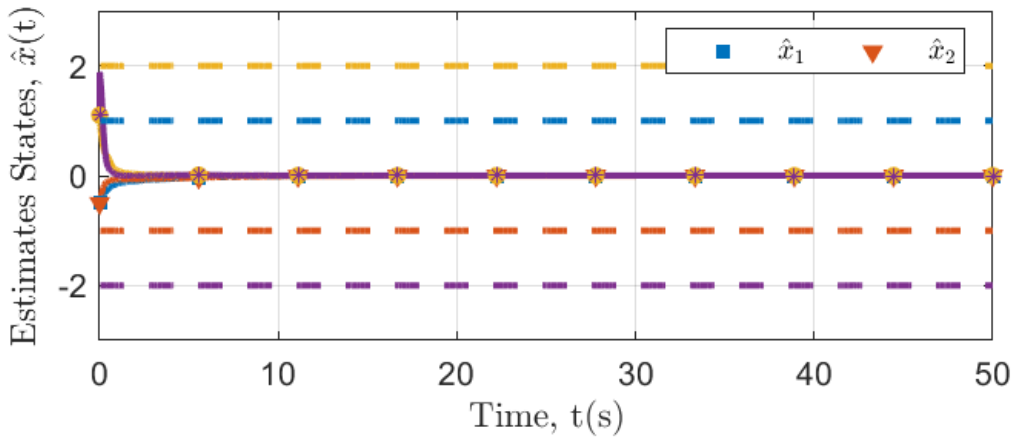

Figure 15: Estimated State trajectories for the four-state dynamical system using MBRL with state estimator in the original coordinates. The dash lines represent the user-selected safe set.

niques generate slightly different trajectories in the state space (see Fig. 18) and the total cost of the trajectories is different. We hypothesize that the difference in costs is due to the basis for value function approximation being unknown.

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

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

Sensitivity Analysis for the four state system

The gains k_c, k_{a1}, k_{a2}, β, and v are selected for the sensitivity analysis. The costs of the trajectories, under the optimal feedback controller obtained using the developed method, are presented in Table 8. for 5 different values of each gain.

The gains are varied in a neighborhood of the nominal values (selected through trial and error) $k_c = 1000$, $k_{a1} = 100$, $k_{a2} = 1$, $\beta = 0.001$, $v = 500$, $k = 0.001$, $\alpha = 1$,

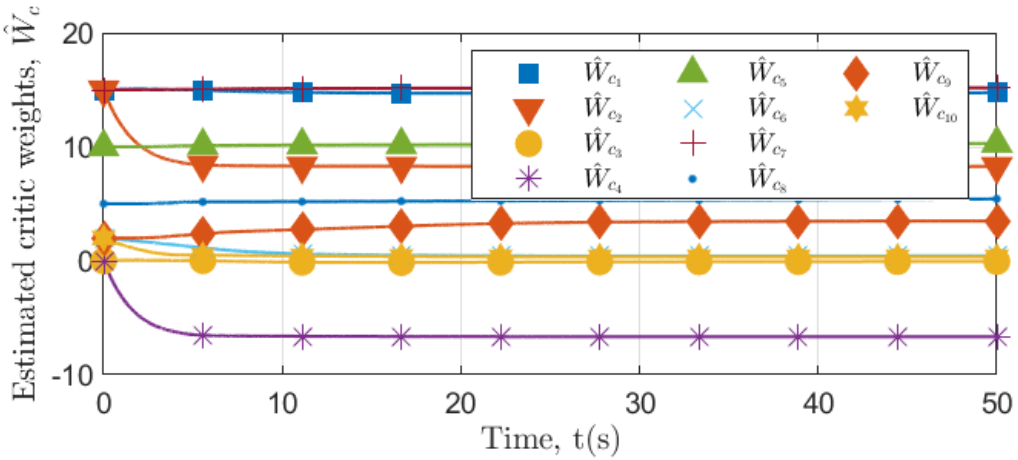

Figure 16: Estimates of the critic weights under nominal gains for the four-state dynamical system.

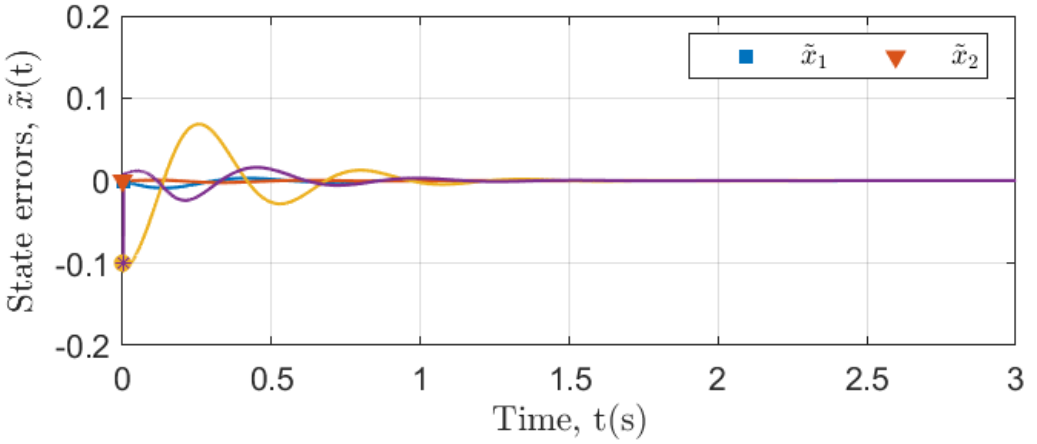

Figure 17: Estimation errors between the original states and the estimated states under nominal gains for the four-state dynamical system.

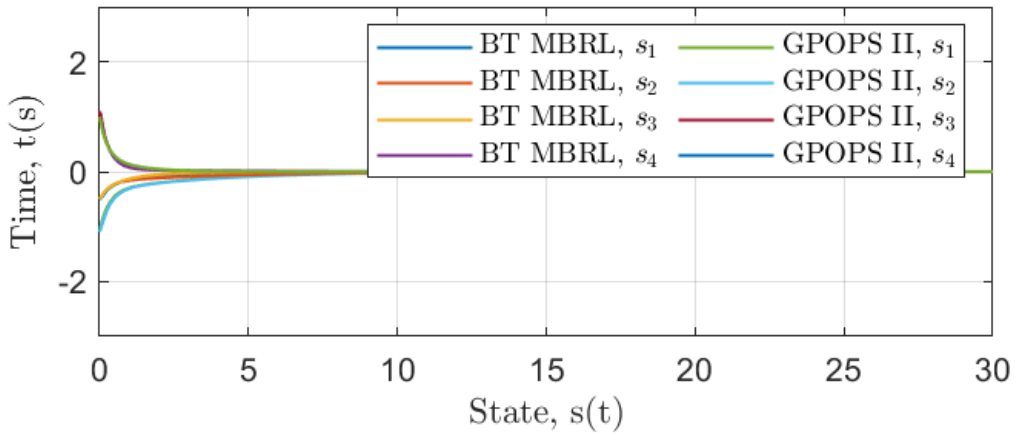

Figure 18: Comparison of the optimal state trajectories obtained using GPOPS II and using BT MBRL with fixed optimal weights for the four-state dynamical system.

Table 8.: Sensitivity Analysis for the four state system. The gains are varied in a neighborhood of the nominal values (selected through trial and error) $k = 0.001$, $\alpha = 1$, $\beta_1 = 100$, $k_c = 1000$, $k_{a1} = 100$, $k_{a2} = 1$, $\beta = 0.001$, $v = 500$; WNC and NF indicate weights not converging and not feasible, respectively.

k_c=	100	500	1000	2000	5000
Cost	11.7277	9.94	11.226	WNC	WNC
k_{a_1}=	10	50	100	250	500
Cost	13.01	11.546	11.226	11.226	11.94
k_{a_2}=	0.01	0.1	1	10	100
Cost	11.326	11.306	11.226	11.42	520.06
β=	0.00001	0.0001	0.001	0.01	0.1
Cost	11.234	11.229	11.226	WNC	WNC
v=	1	50	500	1000	5000
Cost	NF	WNC	11.226	12.026	14.7279

and $\beta_1 = 100$. The results in Table 8. indicate that the developed method is not sensitive to small changes in the learning gains.

Chapter V

CONCLUSION AND FUTURE WORK

5.1 Summary

This thesis focuses on addressing the two key issues: (a) safety, (b) online learning and optimization.

The method to address safety in this thesis, barrier transformation (BT), is an effective method to address the safety issue for a dynamical system in real time as this method reduces the computational cost significantly by avoiding the state constraints. While RL is a powerful technique for optimization and online learning, it is often difficult to use RL to synthesis controllers safely due to the trial and error learning approach that is fundamental to RL. Hence, RL typically requires a large number of iterations due to sample inefficiency. Sample efficiency in RL can be improved using model-based reinforcement learning (MBRL); however, Methods based on MBRL are vulnerable to failure as a result of inaccuracies in models with parametric uncertainties and/or partially observable models. To address this issue, two model-based reinforcement learning (MBRL) techniques for the safety-aware systems have been developed in this thesis.

5.2 Results

Chapter III addresses the optimal controller synthesis issue for the safety-aware systems with parametric uncertainties. This chapter presents a novel online MBRL based controller which uses BFs, BE extrapolation and a novel FCL method. A

known BF transformation is applied to a constrained optimal control problem to generate an unconstrained optimal control problem in the transformed coordinates. The system dynamics, if linear in the parameters in the original coordinates, are shown to be also linearly parameterized in the transformed coordinates. MBRL is used to solve the problem online in the transformed coordinates in conjunction with the novel FCL to learn the unknown model parameters. Regulation of the system states to a neighborhood of the origin and convergence of the estimated policy to a neighborhood of the optimal policy is determined using a Lyapunov-based stability analysis. Simulations are used to demonstrate the applicability of the developed approaches, and to demonstrate their usefulness, comparative simulations are shown whenever alternative techniques are available.

Chapter IV addresses the optimal controller synthesis issue for the safety-aware partially observable systems. This chapter presents a novel framework for approximate optimal control of a class of safety aware nonlinear systems. The framework consists of a novel safe state estimator, and a novel online MBRL based controller. A BT has been applied to a constrained optimal control problem to generate an unconstrained optimal control problem in the transformed coordinates. MBRL is used to solve the problem online in the transformed coordinates in conjunction with the novel state estimator to estimate the transformed states. In the developed method, the cost function is selected to be quadratic in the transformed coordinates. Regulation of the system states to a neighborhood of the origin and convergence of the estimated policy to a neighborhood of the optimal policy is determined using a Lyapunov-based stability analysis. Furthermore, state estimator-based BT MBRL controller is guaranteed to keep the state of the original system within the safety bounds. Simulations are used to demonstrate the applicability of the developed approach, and to demonstrate their usefulness, comparative simulations are shown whenever alternative techniques are available.

5.3 Limitations and future work

Limitations and possible extensions of the ideas presented in this thesis revolve around the same two key issues: (a) safety, and (b) online learning and optimization.

The barrier function used in the BT to address safety is not time varying, a more generic, and adaptive barrier function constructed, perhaps, using sensor data is a subject for future research. The BT method used to address safety uses a box-based barrier, a different barrier approach can be another interesting subject for future research.

For optimal learning, parametric approximation techniques are used to approximate the value functions in this thesis. Parametric approximation of the value function requires selection of appropriate basis functions which may be hard to find for the real-world systems. Developing techniques to systematically determine a set of basis functions for real-world systems is a subject for research.

The barrier transformation method to ensure safety relies on the dynamics of the system. While chapter III addresses parametric uncertainties, the established methods could result a potential safety violation due to the non-parametric uncertainties. To be specific, since the safety relies on the inverting barrier function to recover the original dynamics, Lemma 3.1.1, Lemma 4.3.1, and Lemma 4.3.2 which link between the original dynamics and the transformed dynamics may break down due to the non-parametric uncertainties/unmodeled dynamics; resulting a potential safety violation or/and instability. Future studies can focus on developing a more rigorous theoretical case and/or a more robust approach for ensuring safety.

The approaches developed in this thesis guarantee local stability over a small compact set which causes the difficulty of determining correct gains to stabilize the the states of the system.

A more direct extension of this thesis involves developing techniques to solve the model uncertainty issue for the safety aware partially observable systems with

parametric uncertainties, which can be achieved by merging the techniques developed in this thesis.

Besides, in the developed method, the cost function is selected to be quadratic in the transformed coordinates. We have optimized our cost function in the transformed coordinate. However, a physically meaningful cost function is more likely to be available in the original coordinates. Hence, techniques to transform cost functions from the original coordinates to the barrier coordinates ensure that optimization in barrier coordinates also corresponds to optimization in the original coordinates is another topic for future research.

Bibliography

[1] C. Darwin, *On the Origin of Species by Means of Natural Selection*. London: Murray, 1859, or the Preservation of Favored Races in the Struggle for Life.

[2] I. P. Pavlov, W. H. Gantt, and G. V. Folbort, *Lectures on conditioned reflexes*. Liverwright Publishing Corporation, 1928.

[3] V. Duchaine and C. Gosselin, "Safe, stable and intuitive control for physical human-robot interaction," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 3383–3388.

[4] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[5] S. M. LaValle and J. J. Kuffner Jr, "Randomized kinodynamic planning," *Int. J. Robot. Res.*, vol. 20, no. 5, pp. 378–400, 2001.

[6] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 846–894, 2011.

[7] L. Janson, E. Schmerling, A. Clark, and M. Pavone, "Fast marching tree: A fast marching sampling-based method for optimal motion planning in many dimensions," *Int. J. Robot. Res.*, vol. 34, no. 7, pp. 883–921, 2015.

[8] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Transactions on control systems technology*, vol. 15, no. 3, pp. 566–580, 2007.

[9] P. Falcone, F. Borrelli, J. Asgari, H. Tseng, and D. Hrovat, "Low complexity mpc schemes for integrated vehicle dynamics control problems," in *9th international symposium on advanced vehicle control (AVEC)*, 2008.

[10] T. M. Howard and A. Kelly, "Optimal rough terrain trajectory generation for wheeled mobile robots," *The International Journal of Robotics Research*, vol. 26, no. 2, pp. 141–166, 2007.

[11] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[12] M. A. Patterson and A. V. Rao, "Gpops-ii: A matlab software for solving multiple-phase optimal control problems using hp-adaptive gaussian quadrature collocation methods and sparse nonlinear programming," *ACM Transactions on Mathematical Software (TOMS)*, vol. 41, no. 1, pp. 1–37, 2014.

[13] A. D. Wilson, J. A. Schultz, A. R. Ansari, and T. D. Murphey, "Real-time trajectory synthesis for information maximization using sequential action control and least-squares estimation," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4935–4940.

[14] A. R. Ansari and T. D. Murphey, "Sequential action control: Closed-form optimal control for nonlinear and nonsmooth systems," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1196–1214, 2016.

[15] J. Wurts, J. L. Stein, and T. Ersal, "Collision imminent steering using nonlinear model predictive control," in *2018 Annual American Control Conference (ACC)*. IEEE, 2018, pp. 4772–4777.

[16] J. Ding, E. Li, H. Huang, and C. J. Tomlin, "Reachability-based synthesis of feedback policies for motion planning under bounded disturbances," in *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2011, pp. 2160–2165.

[17] A. Majumdar, R. Vasudevan, M. M. Tobenkin, and R. Tedrake, "Convex optimization of nonlinear feedback controllers via occupation measures," *Int. J. Robot. Res.*, vol. 33, no. 9, pp. 1209–1230, 2014.

[18] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[19] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *The International Journal of Robotics Research*, vol. 36, no. 8, pp. 947–982, 2017.

[20] M. Althoff, "An introduction to cora 2015," in *Proc. of the Workshop on Applied Verification for Continuous and Hybrid Systems*, 2015.

[21] S. Kousik, S. Vaskov, F. Bu, M. Johnson-Roberson, and R. Vasudevan, "Bridging the gap between safety and real-time performance in receding-horizon trajectory design for mobile robots," *The International Journal of Robotics Research*, vol. 39, no. 12, pp. 1419–1469, 2020.

[22] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.

[23] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*, 2019, pp. 3420–3431.

[24] M. H. Cohen and C. Belta, "Approximate optimal control for safety-critical systems with control barrier functions," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 2062–2067.

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

[26] O. von Stryk and R. Bulirsch, "Direct and indirect methods for trajectory optimization," *Ann. Oper. Res.*, vol. 37, no. 1, pp. 357–373, 1992.

[27] J. T. Betts, "Survey of numerical methods for trajectory optimization," *J. Guid. Control Dynam.*, vol. 21, no. 2, pp. 193–207, 1998.

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

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

[30] P. J. Werbos, "A menu of designs for reinforcement learning over time," *Neural Netw. for Control*, pp. 67–95, 1990.

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

[32] R. E. Bellman, *Dynamic programming*. Mineola, NY: Dover Publications, Inc., 2003.

[33] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, 3rd ed. Belmont, MA: Athena Scientific, 2007, vol. 2.

[34] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control*, 3rd ed. Hoboken, NJ: Wiley, 2012.

[35] E. G. Al'Brekht, "On the optimal stabilization of nonlinear systems," *J. Appl. Math. Mech.*, vol. 25, no. 5, pp. 1254–1266, 1961.

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

[37] P. J. 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. Sorge, Eds. Nostrand, New York, 1992, vol. 15, pp. 493–525.

[38] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-dynamic programming*. Athena Scientific, 1996.

[39] J. N. Tsitsiklis and B. Van Roy, "An analysis of temporal-difference learning with function approximation," *IEEE Trans. Autom. Control*, vol. 42, no. 5, pp. 674–690, 1997.

[40] J. N. Tsitsiklis and B. V. Roy, "Average cost temporal-difference learning," *Automatica*, vol. 35, no. 11, pp. 1799–1808, 1999.

[41] J. N. Tsitsiklis, "On the convergence of optimistic policy iteration," *J. Mach. Learn. Res.*, vol. 3, pp. 59–72, 2003.

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

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

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

[45] M. Elbanhawi and M. Simic, "Sampling-based robot motion planning: A review," *IEEE Access*, vol. 2, pp. 56–77, 2014.

[46] Y. Kuwata, J. Teo, G. Fiore, S. Karaman, E. Frazzoli, and J. P. How, "Real-time motion planning with applications to autonomous urban driving," *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1105–1118, 2009.

[47] B. Luders, M. Kothari, and J. How, "Chance constrained rrt for probabilistic robustness to environmental uncertainty," in *AIAA guidance, navigation, and control conference*, 2010, p. 8160.

[48] J. V. Frasch, A. Gray, M. Zanon, H. J. Ferreau, S. Sager, F. Borrelli, and M. Diehl, "An auto-generated nonlinear mpc algorithm for real-time obstacle avoidance of ground vehicles," in *2013 European Control Conference (ECC)*. IEEE, 2013, pp. 4136–4141.

[49] A. Liniger and J. Lygeros, "Real-time control for autonomous racing based on viability theory," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 2, pp. 464–478, 2017.

[50] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin, "Fastrack: A modular framework for fast and guaranteed safe motion planning," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE, 2017, pp. 1517–1522.

[51] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Trans. Autom. Control*, vol. 50, no. 7, pp. 947–957, Jul. 2005.

[52] D. Fridovich-Keil, J. F. Fisac, and C. J. Tomlin, "Safely probabilistically complete real-time planning and exploration in unknown environments," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7470–7476.

[53] M. Nagumo, "Über die lage der integralkurven gewöhnlicher differentialgleichungen," in *Proc. Phys-Math. Soc.*, 1942, pp. 272–559.

[54] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[55] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Hybrid Systems: Computation and Control*, R. Alur and G. J. Pappas, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 477–492.

[56] S. Prajna, "Barrier certificates for nonlinear model validation," *Automatica*, vol. 42, no. 1, pp. 117–126, 2006.

[57] S. Prajna, A. Jadbabaie, and G. J. Pappas, "A framework for worst-case and stochastic safety verification using barrier certificates," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1415–1428, Aug. 2007.

[58] J. Nocedal and S. Wright, *Numerical Optimization*, ser. Springer Series in Operations Research and Financial Engineering. Springer New York, 2000.

[59] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier Lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

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

[61] J.-P. Aubin, A. M. Bayen, and P. Saint-Pierre, *Viability theory*, 2nd ed. Springer-Verlag Berlin Heidelberg, 2011.

[62] P. Wieland and F. Allgöwer, "Constructive safety using control barrier functions," *IFAC Proc. Vol.*, vol. 40, no. 12, pp. 462–467, 2007.

[63] Y. Chen, M. Ahmadi, and A. D. Ames, "Optimal safe controller synthesis: A density function approach," in *2020 American Control Conference (ACC)*, 2020, pp. 5407–5412.

[64] G. Yang, C. Belta, and R. Tron, "Self-triggered control for safety critical systems using control barrier functions," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 4454–4459.

[65] W. Xiao, C. A. Belta, and C. G. Cassandras, "Feasibility-guided learning for constrained optimal control problems," in *2020 59th IEEE Conference on Decision and Control (CDC)*, 2020, pp. 1896–1901.

[66] M. Jankovic, "Robust control barrier functions for constrained stabilization of nonlinear systems," *Automatica*, vol. 96, pp. 359–367, 2018.

[67] D. Kirk, *Optimal control theory: an introduction*. Mineola, NY: Dover, 2004.

[68] F. Fahroo and I. M. Ross, "Pseudospectral methods for infinite-horizon nonlinear optimal control problems," *J. Guid. Control Dynam.*, vol. 31, no. 4, pp. 927–936, 2008.

[69] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation," *J. Guid. Control Dynam.*, vol. 10, no. 4, pp. 338–342, 1987.

[70] G. T. Huntington, "Advancement and analysis of a Gauss pseudospectral transcription for optimal control," Ph.D. dissertation, Department of Aeronautics and Astronautics, MIT, May 2007.

[71] A. V. Rao, "A survey of numerical methods for optimal control," *Adv. Astronaut. Sci.*, vol. 135, no. 1, pp. 497–528, 2009.

[72] C. Darby, "Hp-pseudospectral method for solving continuous-time nonlinear optimal control problems," Ph.D. dissertation, Department of Mechanical and Aerospace Engineering, University of Florida, Apr. 2011.

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

[74] R. A. Freeman and P. V. Kokotovic, *Robust nonlinear control design: state-space and Lyapunov techniques*. Boston, MA: Birkhäuser, 1996.

[75] J. L. Fausz, V.-S. Chellaboina, and W. M. Haddad, "Inverse optimal adaptive control for nonlinear uncertain systems with exogenous disturbances," in *Proc. IEEE Conf. Decis. Control*, Dec. 1997, pp. 2654–2659.

[76] P. Y. Li, "Control of smart exercise machines - Part II: self-optimizing control," *IEEE/ASME Trans. Mechatron.*, vol. 2, no. 4, pp. 237–347, 1997.

[77] W. Luo, Y.-C. Chu, and K.-V. Ling, "Inverse optimal adaptive control for attitude tracking of spacecraft," *IEEE Trans. Autom. Control*, vol. 50, no. 11, pp. 1639–1654, Nov. 2005.

[78] K. Dupree, M. Johnson, P. M. Patre, and W. E. Dixon, "Inverse optimal control of a nonlinear Euler-Lagrange system, Part II: output feedback," in *Proc. IEEE Conf. Decis. Control*, Shanghai, China, Jan. 2009, pp. 327–332.

[79] K. Dupree, W. E. Dixon, G. Hu, and C.-H. Liang, "Lyapunov-based control of a robot and mass-spring system undergoing an impact collision," in *Proc. Am. Control Conf.*, Minneapolis, Minnesota, Jun. 2006, pp. 3241–3246.

[80] M. Johnson, G. Hu, K. Dupree, and W. E. Dixon, "Inverse optimal homography-based visual servo control via an uncalibrated camera," in *Proc. IEEE Conf. Decis. Control*, Shanghai, China, Jan. 2009, pp. 2408–2413.

[81] Q. Wang, N. Sharma, M. Johnson, C. M. Gregory, and W. E. Dixon, "Adaptive inverse optimal neuromuscular electrical stimulation," *IEEE Trans. Cybern.*, vol. 43, pp. 1710–1718, 2013.

[82] D. L. Lukes, "Optimal regulation of nonlinear dynamical systems," *SIAM J. Control*, vol. 7, no. 1, pp. 75–100, 1969.

[83] Y. Nishikawa, N. Sannomiya, and H. Itakura, "A method for suboptimal design of nonlinear feedback systems," *Automatica*, vol. 7, no. 6, pp. 703–712, 1971.

[84] W. L. Garrard and J. M. Jordan, "Design of nonlinear automatic flight control systems," *Automatica*, vol. 13, no. 5, pp. 497–505, 1977.

[85] I. C. Dolcetta, "On a discrete approximation of the Hamilton-Jacobi equation of dynamic programming," *Appl. Math. Optim.*, vol. 10, no. 1, pp. 367–377, 1983.

[86] M. Falcone and R. Ferretti, "Discrete time high-order schemes for viscosity solutions of Hamilton-Jacobi-Bellman equations," *Numer. Math.*, vol. 67, no. 3, pp. 315–344, 1994.

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

[88] R. Gonzalez and E. Rofman, "On deterministic control problems: an approximation procedure for the optimal cost ii. the nonstationary case," *SIAM J. Control Optim.*, vol. 23, no. 2, pp. 267–285, 1985.

[89] M. Falcone, "A numerical approach to the infinite horizon problem of deterministic control theory," *Appl. Math. Optim.*, vol. 15, no. 1, pp. 1–13, 1987.

[90] H. J. Kushner, "Numerical methods for stochastic control problems in continuous time," *SIAM J. Control Optim.*, vol. 28, no. 5, pp. 999–1048, 1990.

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

[92] R. W. Beard and T. W. McLain, "A practical algorithm for designing nonlinear H-infinity control laws," in *Proc. Am. Control Conf.*, vol. 6, 1998, pp. 3742–3743.

[93] J. J. Murray, C. J. Cox, G. 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.

[94] M. Abu-Khalaf and F. L. 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.

[95] K. Hornik, M. Stinchcombe, and H. White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Netw.*, vol. 3, no. 5, pp. 551–560, 1990.

[96] B. Widrow, N. K. 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.

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

[98] I. H. Witten, "An adaptive optimal controller for discrete-time Markov environments," *Inf. Control*, vol. 34, no. 4, pp. 286–295, 1977.

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

[100] I. Grondman, L. Buşoniu, G. A. D. Lopes, and R. Babuška, "A survey of actor-critic reinforcement learning: standard and natural policy gradients," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 42, no. 6, pp. 1291–1307, 2012.

[101] D. Fuselli, F. De Angelis, M. Boaro, S. Squartini, Q. Wei, D. Liu, and F. Piazza, "Action dependent heuristic dynamic programming for home energy resource scheduling," *Int. J. Electr. Power Energy Syst.*, vol. 48, pp. 148–160, 2013.

[102] W. T. Miller, R. Sutton, and P. Werbos, *Neural networks for control.* MIT Press, 1990.

[103] P. J. Werbos, "Advanced forecasting methods for global crisis warning and models of intelligence," *Gen. Syst. Yearb.*, vol. 22, pp. 25–38, 1977.

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

[105] L. Yang, R. Enns, Y.-T. Wang, and J. Si, "Direct neural dynamic programming," in *Stability and Control of Dynamical Systems with Applications*. Springer, 2003, pp. 193–214.

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

[107] G. 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.

[108] S. Ferrari and R. F. Stengel, "An adaptive critic global controller," in *Proc. Am. Control Conf.*, vol. 4, 2002, pp. 2665–2670.

[109] D. Han and S. N. 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.

[110] 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.

[111] 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.

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

[113] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, 2012.

[114] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive dynamic programming for control: algorithms and stability*, ser. Communications and Control Engineering. London: Springer-Verlag, 2013.

[115] Q. Wei and D. Liu, "Optimal tracking control scheme for discrete-time nonlinear systems with approximation errors," in *Advances in Neural Networks - ISNN 2013*, ser. Lecture Notes in Computer Science, C. Guo, Z.-G. Hou, and Z. Zeng, Eds. Springer Berlin Heidelberg, 2013, vol. 7952, pp. 1–10.

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

[117] X. Yang, D. Liu, Q. Wei, and D. Wang, "Direct adaptive control for a class of discrete-time unknown nonaffine nonlinear systems using neural networks," *Int. J. Robust Nonlinear Control*, vol. 25, no. 12, pp. 1844–1861, Apr. 2015.

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

[119] D. Vrabie, O. Pastravanu, M. Abu-Khalaf, and F. L. Lewis, "Adaptive optimal control for continuous-time linear systems based on policy iteration," *Automatica*, vol. 45, no. 2, pp. 477–484, 2009.

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

[121] K. G. Vamvoudakis and F. L. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[122] K. S. Narendra and A. M. Annaswamy, *Stable adaptive systems*. Prentice-Hall, Inc., 1989.

[123] S. S. Sastry and M. Bodson, *Adaptive control: stability, convergence, and robustness*. Upper Saddle River, NJ: Prentice-Hall, 1989.

[124] P. Ioannou and J. Sun, *Robust adaptive control*. Prentice Hall, 1996.

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

[126] D. Vrabie, M. Abu-Khalaf, F. L. 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.

[127] G. Chowdhary, "Concurrent learning for convergence in adaptive control without persistency of excitation," Ph.D. dissertation, Georgia Institute of Technology, Dec. 2010.

[128] P. Walters, R. Kamalapurkar, and W. E. Dixon, "Approximate optimal online continuous-time path-planner with static obstacle avoidance," in *Proc. IEEE Conf. Decis. Control*, Osaka, Japan, Dec. 2015, pp. 650–655.

[129] P. Deptula, H. Chen, R. A. Licitra, J. A. Rosenfeld, and W. E. Dixon, "Approximate optimal motion planning to avoid unknown moving avoidance regions," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 414–430, 2020.

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

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

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

[133] M. L. Greene, P. Deptula, S. Nivison, and W. E. Dixon, "Sparse learning-based approximate dynamic programming with barrier constraints," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 743–748, 2020.

[134] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *Int. J. Adapt. Control Signal Process.*, vol. 27, no. 4, pp. 280–301, 2013.

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

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

[137] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of Markov decision processes," *Math. Oper. Res.*, vol. 12, no. 3, pp. 441–450, 1987.

[138] O. Madani, S. Hanks, and A. Condon, "On the undecidability of probabilistic planning and related stochastic optimization problems," *Artif. Intell.*, vol. 147, no. 1-2, pp. 5–34, 2003.

[139] H. Modares, F. L. Lewis, and Z.-P. Jiang, "Optimal output-feedback control of unknown continuous-time linear systems using off-policy reinforcement learning," *IEEE Trans. Cybern.*, vol. 46, no. 11, pp. 2401–2410, Sep. 2016.

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

[141] D. Liberzon, *Switching in systems and control*. Birkhäuser, 2003.

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

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

[144] A. Al-Tamimi, F. L. Lewis, and M. Abu-Khalaf, "Adaptive critic designs for discrete-time zero-sum games with application to H-infinity control," *IEEE Trans. Syst. Man Cybern. Part B Cybern.*, vol. 37, pp. 240–247, 2007.

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

[146] K. G. Vamvoudakis and F. L. Lewis, "Policy iteration algorithm for distributed networks and graphical games," in *Proc. IEEE Conf. Decis. Control*, 2011, pp. 128–135.

[147] K. Vamvoudakis, F. L. Lewis, M. Johnson, and W. E. Dixon, "Online learning algorithm for Stackelberg games in problems with hierarchy," in *Proc. IEEE Conf. Decis. Control*, Maui, HI, Dec. 2012, pp. 1883–1889.

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

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

[150] H. Modares, F. L. Lewis, and M.-B. Naghibi-Sistani, "Integral reinforcement learning and experience replay for adaptive optimal control of partially-unknown constrained-input continuous-time systems," *Automatica*, vol. 50, no. 1, pp. 193–202, 2014.

[151] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, vol. 50, no. 7, pp. 1780–1792, 2014.

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

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

[154] 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.

[155] R. Kamalapurkar, J. A. Rosenfeld, and W. E. Dixon, "Efficient model-based reinforcement learning for approximate online optimal control," *Automatica*, vol. 74, pp. 247–258, Dec. 2016.

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

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

[158] Y. Chen and A. D. Ames, "Duality between density function and value function with applications in constrained optimal control and markov decision process," arXiv:1902.09583, 2019.

[159] H. T. Dinh, R. Kamalapurkar, S. Bhasin, and W. E. Dixon, "Dynamic neural network-based robust observers for uncertain nonlinear systems," *Neural Netw.*, vol. 60, pp. 44–52, Dec. 2014.

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

Appendix

Proofs

A Chapter III

Lemma 3.1.1 If $t \mapsto \Phi\big(t, b(x^0), \zeta\big)$ is a Carathéodory solution to (40), starting from the initial condition $b(x^0)$, under the feedback policy $(s,t) \mapsto \zeta(s,t)$, and if $t \mapsto \Lambda(t, x^0, \zeta)$ is a solution to (32), starting from the initial condition x^0, under the controller $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$, then $\Lambda(t, x^0, \zeta) = b^{-1}\big(\Phi(t, b(x^0), \zeta)\big)$ for all $t \in \mathbb{R}_{\geq 0}$.

Proof. Since $t \mapsto \Phi\big(t; b(x^0), \zeta\big)$ is a Carathéodory solution to $\dot{s} = y(s)\theta + G(s)u$, it is differentiable at almost all t. Since b^{-1} smooth, $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is also differentiable at almost all t. When $b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is differentiable,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)}{\mathrm{d}s} \frac{\mathrm{d}\Phi\big(t, b(x^0), \zeta\big)}{\mathrm{d}t}.$$

So,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)}{\mathrm{d}s} \Big(y\big(\Phi\big(t; b(x^0), \zeta\big)\big)\theta + G\big(\Phi\big(t; b(x^0), \zeta\big)\big)\zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)\Big).$$

By the construction of y and G, for almost all $t \in \mathbb{R}_{\geq 0}$,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = f\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)\theta + g\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)\zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big),$$

clearly $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is a Carathéodory solution to (32), starting from the $b^{-1}\big(b(x^0)\big) = x^0$ under the controller $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$. Finally, continuity of $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ and $t \mapsto \Lambda(t, x^0, \zeta)$ implies that $b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \Lambda(t, x^0, \zeta)$ for all $t \in \mathbb{R}_{\geq 0}$. ■

Lemma 3.2.1 If $\|Y_f\|$ is non-decreasing in time then (46) admits Carathéodory solutions.

Proof. Since $\|Y_f(0)\| = 0$, given any piecewise continuous control signal $t \mapsto u(t)$ and initial conditions s^0 and θ^0, the Cauchy problem $\dot{z} = h_1(z, u)$, $z(0) = z^0 = [s^0; 0; 0; 0; 0; \theta^0]$ admits a unique Carathéodory solution $t \mapsto z_1(0, z^0)$ over $[0, t^*)$, with $t^* = \min(t_1, t_2)$, where $t_1 = \inf\{t \in \mathbb{R}_{\geq 0} \mid \|Y_{f1}(t, z^0\| = \overline{Y_f}\}$ and $t_2 = \inf\{t \in \mathbb{R}_{\geq 0} | \lim_{\tau \mapsto t} \|z_1(\tau, z^0)\| = \infty\}$, where Y_{f1} denotes the Y_f component of z_1.

Given any $(t^{'}, z^{'}) \in \mathbb{R}_{\geq 0} \times \mathbb{R}^{2n+2p+p^2+np}$, the Cauchy problem $\dot{z} = h_2(z, u), z(t^{'}) = z^{'}$, also admits a unique Carathéodory solution $t \mapsto z_2(t; t^{'}, z^{'})$ over $[t^{'}, t^{**})$ where $t^{**} = \min\Big(\infty, \big(\inf\{t \in \mathbb{R}_{\geq t^{'}} | \lim_{\tau \mapsto t} \|z_2(\tau, b^{'}, z^{'})\| = \infty\}\big)\Big)$.

If $t^* = t_2$ then $t \mapsto z_1(t, z^0)$ is also a unique Carathéodory solution to the Cauchy problem $\dot{z} = h(z, u)$, $z(0) = z^0$. If not, then

$$t \mapsto z^*(t, z^0) = \begin{cases} z_1(t, z^0), & t < t_1 \\ z_2\big(t, t_1, \lim_{\tau \uparrow t_1} z_1(\tau, z^0)\big), & t \geq t_1 \end{cases}, \tag{165}$$

is a unique Carathéodory solution to the Cauchy problem $\dot{z} = h(z, u)$, $z(0) = z^0$. ■

B Chapter IV

Lemma 4.3.1 If $t \mapsto \Phi\big(t, b(x^0), \zeta\big)$ is a Carathéodory solution to (109), starting from the initial condition $b(x^0)$, under the feedback policy $(s, t) \mapsto \zeta(s, t)$, and if $t \mapsto \Lambda(t, x^0, \zeta)$ is a solution to (94), starting from the initial condition x^0, under the controller $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$, then $\Lambda(t, x^0, \zeta) = b^{-1}\big(\Phi(t, b(x^0), \zeta)\big)$ for all

$t \in \mathbb{R}_{\geq 0}$.

Proof. Since $t \mapsto \Phi\big(t; b(x^0), \zeta\big)$ is a Carathéodory solution to $\dot{s} = \begin{bmatrix} H(s) \\ F(s) + G(s)u \end{bmatrix}$, it is differentiable at almost all t. Since b^{-1} smooth, $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is also differentiable at almost all t. When $b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is differentiable,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)}{\mathrm{d}s} \frac{\mathrm{d}\Phi\big(t; b(x^0), \zeta\big)}{\mathrm{d}t}.$$

So,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big)}{\mathrm{d}s} \begin{bmatrix} H\big(\Phi\big(t; b(x^0), \zeta\big)\big) \\ F\big(\Phi\big(t; b(x^0), \zeta\big)\big) + G\big(\Phi\big(t; b(x^0), \zeta\big)\big)\zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big) \end{bmatrix}.$$

By the construction of H, F, and G, for almost all $t \in \mathbb{R}_{\geq 0}$,

$$\frac{\mathrm{d}}{\mathrm{d}t} b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \begin{bmatrix} b^{-1}\big(\Phi_2\big(t; b(x_2^0), \zeta\big)\big) \\ f\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big) + g\big(b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)\big) \\ \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big) \end{bmatrix}.$$

Clearly $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ is a Carathéodory solution to (94), starting from the $b^{-1}\big(b(x^0)\big) = x^0$ under the controller $u(t) = \zeta\big(\Phi\big(t; b(x^0), \zeta\big), t\big)$. Finally, continuity of $t \mapsto b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big)$ and $t \mapsto \Lambda(t, x^0, \zeta)$ implies that $b^{-1}\big(\Phi\big(t; b(x^0), \zeta\big)\big) = \Lambda(t, x^0, \zeta)$ for all $t \in \mathbb{R}_{\geq 0}$. ■

Lemma 4.3.2 If $t \mapsto \Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)$ is a Carathéodory solution to (111), starting from the initial condition $b(\hat{x}^0)$ along the trajectory $t \mapsto b(x_1(t))$, and if $t \mapsto \xi(t; x_1(\cdot), \hat{x}^0)$ is a solution to (95), starting from the initial condition $\hat{x}^0$ along the trajectory $x_1(\cdot)$, then $\xi(t; x_1(\cdot), \hat{x}^0) = b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ for all $t \in \mathbb{R}_{\geq 0}$.

Proof. Since $t \mapsto \Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)$ is a Carathéodory solution to $\dot{\hat{s}} = \begin{bmatrix} H(\hat{s}) \\ F(\hat{s}) + G(\hat{s})u + \nu_2(\tilde{s_1}, \eta) \end{bmatrix}$, it is differentiable at almost all t. Since b^{-1} smooth,

$t \mapsto b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ is also differentiable at almost all t.

When $b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ is differentiable,

$$\frac{\mathrm{d}}{\mathrm{d}t}\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big)}{\mathrm{d}s}\frac{\mathrm{d}s}{\mathrm{d}t}.$$

So,

$$\frac{\mathrm{d}}{\mathrm{d}t}\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big) = \frac{\mathrm{d}\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big)}{\mathrm{d}s} \begin{bmatrix} H\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big) \\ F\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big) + G\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)u(t) \\ +\nu_2\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big), t\big) \end{bmatrix}.$$

By the construction of H, F, ν_2 and G, for almost all $t \in \mathbb{R}_{\geq 0}$,

$$\frac{\mathrm{d}}{\mathrm{d}t}\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big) = \begin{bmatrix} b^{-1}\big(\Psi_2\big(t; b(x_1(\cdot)), b(\hat{x}_2^0)\big)\big) \\ f\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big) \\ +g\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)\big)u(t) \\ +\nu_1\big(b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big), t\big)\big) \end{bmatrix}.$$

Clearly $t \mapsto b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ is a Carathéodory solution to (95), starting from the initial condition $b^{-1}\big(b(\hat{x}^0)\big) = \hat{x}^0$ along the trajectory $t \mapsto b(x_1(t))$. Finally, continuity of $t \mapsto b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big)$ and $t \mapsto \xi(t; x_1(\cdot), \hat{x}^0)$ implies that $b^{-1}\big(\Psi\big(t; b(x_1(\cdot)), b(\hat{x}^0)\big)\big) = \xi(t; x_1(\cdot), \hat{x}^0)$ for all $t \in \mathbb{R}_{\geq 0}$. ■

Lemma 4.5.1 Let $V_{se} : \mathbb{R}^{3n} \to \mathbb{R}_{\geq 0}$ be a continuously differentiable candidate Lyapunov function defined as $V_{se}(Z_1) \coloneqq \frac{\alpha^2}{2}\tilde{s}_1^T\tilde{s}_1 + \frac{1}{2}r^Tr + \frac{1}{2}\eta^T\eta$, where $Z_1 \coloneqq [\tilde{s}_1^T, r^T, \eta^T]$. Provided s, $\hat{s} \in \overline{B}(0, \chi)$ for some $\chi > 0$, the orbital derivative of V_{se} along the trajectories of $\dot{\tilde{s}}_1$, $\dot{r}$, and $\dot{\eta}$, defined as $\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \coloneqq \frac{\mathrm{d}V_{se}(Z_1,s,\tilde{s},\tilde{W}_a)}{\mathrm{d}\tilde{s}_1}(H(s) - H(\hat{s})) + \frac{\mathrm{d}V_{se}(Z_1,s,\tilde{s},\tilde{W}_a)}{\mathrm{d}r}\dot{r} + \frac{\mathrm{d}V_{se}(Z_1,s,\tilde{s},\tilde{W}_a)}{\mathrm{d}\eta}\dot{\eta}$, can be bounded as $\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq -\alpha^3\|\tilde{s}_1\|^2 - (k -$

$\varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + \varpi_1\left(1 + \varpi_4 + \varpi_4\alpha\|\right)\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\|$
$+ \varpi_2\|r\|\|\tilde{W}_a\| + \varpi_3\|r\|.$

Proof. Using the fact that V_{se} is PD and Lemma 4.3 from [156] yields

$$\underline{v_l}(\|Z_1\|) \leq V_{se}(Z_1) \leq \overline{v_l}(\|Z_1\|) \tag{166}$$

for all $t \in \mathbb{R}_{\geq t_0}$ and for all $Z_1 \in \mathbb{R}^{3n}$, where $\underline{v_l}, \overline{v_l} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class κ functions. Let $v_l : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ be a class κ function such that $v_l(\|Z_1\|) = \frac{1}{2}(\|\tilde{s}_1\|^2 + \|r\|^2 + \|\eta\|^2)$. Using (104), first state of (111), (128), and (133), the orbital derivative can be expressed as

$$\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) = \alpha^2 {\tilde{s_1}}^T \dot{\tilde{s}}_1 + r^T\dot{r} + \eta^T\dot{\eta}. \tag{167}$$

(131) and (133) yields,

$$\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) = \alpha^2 \tilde{s}_1^T(r - \alpha\tilde{s_1} - \eta) + r^T\dot{r} + \eta^T(-\beta_1\eta - kr - \alpha\dot{\tilde{s}}_1), \tag{168}$$

using (130),

$$\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) = -\alpha^3 {\tilde{s_1}}^T\tilde{s_1} - kr^Tr - (\beta_1 - \alpha)\eta^T\eta + (r^T\tilde{F}_2(s, \hat{s}) + r^T\tilde{F}_3(s, \hat{s}) + r^T\tilde{G}_1(s, \hat{s})\hat{u}). \tag{169}$$

Rewriting (169) as

$$\begin{aligned}\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq & -\alpha^3 {\tilde{s_1}}^T\tilde{s_1} - kr^Tr - (\beta_1 - \alpha)\eta^T\eta + r^T\tilde{F}_2(s, \hat{s}) + r^T\tilde{F}_3(s, \hat{s}) \\ & -r^T\tilde{G}_1(s, \hat{s})\,\hat{u}\left(s, \tilde{W}_a\right) + r^T\tilde{G}_1(s, \hat{s})\,\tilde{u}\left(s, \hat{s}, \tilde{W}_a\right) - r^T\tilde{G}_1(s, \hat{s})\,\tilde{u}(s, \hat{s}, W) \\ & + r^T\tilde{G}_1(s, \hat{s})\,\hat{u}(s, W),\end{aligned} \tag{170}$$

which yields

$$\begin{aligned}\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq & -\alpha^3 {\tilde{s_1}}^T\tilde{s_1} - kr^Tr - (\beta_1 - \alpha)\eta^T\eta + r^T\tilde{F}_2(s, \hat{s}) + r^T\tilde{F}_3(s, \hat{s}) \\ & -r^T\tilde{G}_1(s, \hat{s})\,\hat{u}\left(s, \tilde{W}_a\right) + r^T\tilde{G}_1(s, \hat{s})\,\hat{u}\left(s, \tilde{W}_a\right) - r^T\tilde{G}_1(s, \hat{s})\,\hat{u}\left(\hat{s}, \tilde{W}_a\right) \\ & -r^T\tilde{G}_1(s, \hat{s})\,\hat{u}(s, W) + r^T\tilde{G}_1(s, \hat{s})\,\hat{u}(\hat{s}, W) + r^T\tilde{G}_1(s, \hat{s})\,\hat{u}(s, W).\end{aligned} \tag{171}$$

Simplifying (171) yields

$$\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq -\alpha^3 \tilde{s_1}^T \tilde{s_1} - k r^T r - (\beta_1 - \alpha)\eta^T \eta + r^T \tilde{F}_2(s, \hat{s}) + r^T \tilde{F}_3(s, \hat{s}) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}\left(s, \tilde{W}_a\right) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}\left(s, \tilde{W}_a\right) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}\left(\hat{s}, \tilde{W}_a\right) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}(s, W) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}(\hat{s}, W) + r^T \tilde{G}_1(s, \hat{s})\, \hat{u}(s, W). \quad (172)$$

Using the Cauchy-Schwarz inequality and the fact that F_2, F_3, and G are Lipschitz continuous on $\overline{B}(0, \chi)$.

$$\dot{V}_{se}(Z_1, s, \tilde{s}, \tilde{W}_a) \leq -\alpha^3 \tilde{s_1}^T \tilde{s_1} - k r^T r - (\beta_1 - \alpha)\eta^T \eta + \varpi_1 \|r\| \|\tilde{s}\| + \varpi_2 \|r\| \|\tilde{W}_a\| + \varpi_3 \|r\|, \quad (173)$$

Provided $s, \hat{s} \in \overline{B}(0, \chi)$ from (106) and (111),

$$s_2 = b\left(\dot{s}_1\left(\frac{A_1 a_1^2 - a_1 A_1^2}{a_1^2 e^{s_1} - 2a_1 A_1 + A_1^2 e^{-s_1}}\right)\right) = h(s_1, \dot{s}_1),$$

$$\hat{s}_2 = b\left(\dot{\hat{s}}_1\left(\frac{A_1 a_1^2 - a_1 A_1^2}{a_1^2 e^{\hat{s}_1} - 2a_1 A_1 + A_1^2 e^{-\hat{s}_1}}\right)\right) = h(\hat{s}_1, \dot{\hat{s}}_1),$$

and

$$\tilde{s}_2 = s_2 - \hat{s}_2 = h(s_1, \dot{s}_1) - h(\hat{s}_1, \dot{\hat{s}}_1). \quad (174)$$

Provided $\dot{s}_1$ is fixed, Lipschitz continuity of h, we can write,

$$|h(s_1, \dot{s}_1) - h(\hat{s}_1, \dot{s}_1)| \leq \varpi_4 \|(s_1, \dot{s}_1) - (\hat{s}_1, \dot{s}_1)\|, \quad (175)$$

where ϖ_4 is the Lipschitz constant. (175) yields,

$$|h(s_1, \dot{s}_1) - h(\hat{s}_1, \dot{s}_1)| \leq \varpi_4 \|s_1 - \hat{s}_1\| \quad \text{or,} \quad |h(s_1, \dot{s}_1) - h(\hat{s}_1, \dot{s}_1)| \leq \varpi_4 \|\tilde{s}_1\|. \quad (176)$$

Provided s_1 is fixed, Lipschitz continuity of h, we can write,

$$|h(s_1, \dot{s}_1) - h(s_1, \dot{\hat{s}}_1)| \leq \varpi_4 \|(s_1, \dot{s}_1) - (s_1, \dot{\hat{s}}_1)\|, \quad (177)$$

(177) yields,

$$|h(s_1,\dot{s}_1)-h(s_1,\dot{\hat{s}}_1)|\leq \varpi_4\|\dot{s}_1-\dot{\hat{s}}_1\| \quad \text{or,} \quad |h(s_1,\dot{s}_1)-h(s_1,\dot{\hat{s}}_1)|\leq \varpi_4\|\dot{\tilde{s}}_1\|. \tag{178}$$

Provided $s,\hat{s}\in\chi$, Lipschitz continuity of h can be exploited to derive the bound

$$\begin{aligned}|h(s_1,\dot{s}_1)-h(\hat{s}_1,\dot{\hat{s}}_1)| &= |h(s_1,\dot{s}_1)-h(\hat{s}_1,\dot{s}_1)+h(\hat{s}_1,\dot{s}_1)-h(\hat{s}_1,\dot{\hat{s}}_1)|\\ &\leq |h(s_1,\dot{s}_1)-h(\hat{s}_1,\dot{s}_1)|+|h(\hat{s}_1,\dot{s}_1)-h(\hat{s}_1,\dot{\hat{s}}_1)|\leq \varpi_4\|\tilde{s}_1\|+\varpi_4\|\dot{\tilde{s}}_1\|\\ &\leq \varpi_4\|\tilde{s}_1\|+\varpi_4\|r-\alpha\tilde{s}_1-\eta\|. \end{aligned}\tag{179}$$

Using the triangle inequality,

$$\|\tilde{s}\|\leq\|\tilde{s_1}\|+\|\tilde{s_2}\|\leq(1+\varpi_4+\varpi_4\alpha)\|\tilde{s}_1\|+\varpi_4\|r\|+\varpi_4\|\eta\|. \tag{180}$$

Substituting (180) into (173)yields

$$\begin{aligned}\dot{V}_{se}(Z_1,s,\tilde{s},\tilde{W}_a) &\leq -\alpha^3\tilde{s_1}^T\tilde{s_1}-kr^Tr-(\beta_1-\alpha)\eta^T\eta\\ &+\varpi_1\|r\|\Big((1+\varpi_4+\varpi_4\alpha)\|\tilde{s}_1\|+\varpi_4\|r\|+\varpi_4\|\eta\|\Big)+\varpi_2\|r\|\|\tilde{W}_a\|+\varpi_3\|r\|. \end{aligned}\tag{181}$$

(181) can be rearranged as

$$\begin{aligned}\dot{V}_{se}(Z_1,s,\tilde{s},\tilde{W}_a) &\leq -\alpha^3\|\tilde{s}_1\|^2-(k-\varpi_1\varpi_4)\|r\|^2-(\beta_1-\alpha)\|\eta\|^2\\ &+\varpi_1\left(1+\varpi_4+\varpi_4\alpha\|\right)\|r\|\|\tilde{s}_1\|+\varpi_1\varpi_4\|r\|\|\eta\|+\varpi_2\|r\|\|\tilde{W}_a\|+\varpi_3\|r\|. \end{aligned}\tag{182}$$

■

B.1 Full derivative of Weight parameters

Let $\Theta\left(\tilde{W}_c,\tilde{W}_a,t\right):=\frac{1}{2}\tilde{W}_c^T\Gamma^{-1}(t)\,\tilde{W}_c+\frac{1}{2}\tilde{W}_a^T\tilde{W}_a$. The orbital derivative of Θ along the trajectories of (138) - (140) is defined as

$$\dot{\Theta}\left(\tilde{W}_c,\tilde{W}_a,t\right)=\tilde{W}_c^T\Gamma^{-1}\dot{\tilde{W}}_c-\frac{1}{2}\tilde{W}_c^T\Gamma^{-1}\dot{\Gamma}\Gamma^{-1}\tilde{W}_c+\tilde{W}_a^T\dot{\tilde{W}}_a, \tag{183}$$

where $\dot{\tilde{W}}_c = -\dot{\hat{W}}_c$, and $\dot{\tilde{W}}_a = -\dot{\hat{W}}_a$. Substituting (138) - (140) in (183) yields,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\tilde{W}_c^T \Gamma^{-1} \left(-\frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k}{\rho_k} \hat{\delta}_k \right) - \frac{1}{2} \tilde{W}_c^T \Gamma^{-1} \left(\beta \Gamma - \frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \Gamma \right) \Gamma^{-1} \tilde{W}_c - \tilde{W}_a^T \left(-k_{a1} \left(W_a - W_c \right) - k_{a2} W_a + \sum_{k=1}^{N} \frac{k_c G_k^T W_a \omega_k^T}{4 N \rho_k} W_c \right), \quad (184)$$

so,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\tilde{W}_c^T \Gamma^{-1} \left(-\frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k}{\rho_k} \hat{\delta}_k \right) - \frac{1}{2} \tilde{W}_c^T \Gamma^{-1} \left(\beta \Gamma - \frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \Gamma \right) \Gamma^{-1} \tilde{W}_c - \tilde{W}_a^T \left(-k_{a1} \left(W_a - W_c \right) - k_{a2} W_a + \sum_{k=1}^{N} \frac{k_c G_k^T W_a \omega_k^T}{4 N \rho_k} W_c \right), \quad (185)$$

so,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\tilde{W}_c^T \Gamma^{-1} \left(-\frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k}{\rho_k} \hat{\delta}_k \right) - \frac{\beta}{2} \tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{k_c}{2N} \tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \tilde{W}_c - \left(k_{a1} + k_{a2} \right) \tilde{W}_a^T \tilde{W}_a + k_{a1} \tilde{W}_a^T \tilde{W}_c + k_{a2} \tilde{W}_a^T W - \sum_{k=1}^{N} \tilde{W}_a^T \frac{k_c G_k^T W_a \omega_k^T}{4 N \rho_k} W_c, \quad (186)$$

so

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\tilde{W}_c^T \Gamma^{-1} \left(-\frac{k_c}{N} \Gamma \sum_{k=1}^{N} \frac{\omega_k}{\rho_k} \left(-\omega_k^T \tilde{W}_c + \frac{1}{4} \tilde{W}_a^T G_{\sigma_k} \tilde{W}_a + \Delta_k \right) \right) - \frac{\beta}{2} \tilde{W}_c^T \Gamma^{-1} \tilde{W}_c + \frac{k_c}{2N} \tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k \omega_k^T}{\rho_k^2} \tilde{W}_c - \left(k_{a1} + k_{a2} \right) \tilde{W}_a^T \tilde{W}_a + k_{a1} \tilde{W}_a^T \tilde{W}_c + k_{a2} \tilde{W}_a^T W - \sum_{k=1}^{N} \tilde{W}_a^T \frac{k_c G_k^T W_a \omega_k^T}{4 N \rho_k} W_c, \quad (187)$$

so,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\frac{k_c}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\omega_k^T \tilde{W}_c + \frac{k_c}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\frac{1}{4}\tilde{W}_a^T G_{\sigma_k}\tilde{W}_a$$
$$+ \frac{k_c}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\Delta_k - \frac{\beta}{2}\tilde{W}_c^T\Gamma^{-1}\tilde{W}_c + \frac{k_c}{2N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k\omega_k^T}{\rho_k^2}\tilde{W}_c - (k_{a1}+k_{a2})\,\tilde{W}_a^T\tilde{W}_a$$
$$+ k_{a1}\tilde{W}_a^T\tilde{W}_c + k_{a2}\tilde{W}_a^T W - \sum_{k=1}^{N}\tilde{W}_a^T \frac{k_c G_k^T W_a \omega_k^T}{4N\rho_k} W_c, \quad (188)$$

so,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\frac{\beta}{2}\tilde{W}_c^T\Gamma^{-1}\tilde{W}_c - (k_{a1}+k_{a2})\,\tilde{W}_a^T\tilde{W}_a - \frac{k_c}{2N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\omega_k^T\tilde{W}_c$$
$$- \frac{k_c}{2N}\tilde{W}_c^T \sum_{k=1}^{N}\left(\frac{\omega_k\omega_k^T}{\rho_k} - \frac{\omega_k\omega_k^T}{\rho_k^2}\right)\tilde{W}_c + \frac{k_c}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\Delta_k + \frac{k_c}{N}\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{\rho_k}\frac{1}{4}\tilde{W}_a^T G_{\sigma_k}\tilde{W}_a$$
$$+ k_{a1}\tilde{W}_a^T\tilde{W}_c + k_{a2}\tilde{W}_a^T W - \sum_{k=1}^{N}\tilde{W}_a^T \frac{k_c G_k^T W_a \omega_k^T}{4N\rho_k} W_c, \quad (189)$$

so,

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) = -\tilde{W}_c^T\left(\frac{\beta}{2}\Gamma^{-1} + \frac{k_c}{2N}\sum_{k=1}^{N}\frac{\omega_k\omega_k^T}{\rho_k}\right)\tilde{W}_c - (k_{a1}+k_{a2})\,\tilde{W}_a^T\tilde{W}_a$$
$$+ k_c\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{N\rho_k}\Delta_k + k_c\tilde{W}_c^T \sum_{k=1}^{N} \frac{\omega_k}{4N\rho_k}\tilde{W}_a^T G_{\sigma_k}\tilde{W}_a + k_{a1}\tilde{W}_a^T\tilde{W}_c$$
$$+ k_{a2}\tilde{W}_a^T W - k_c \sum_{k=1}^{N}\tilde{W}_a^T \frac{G_k^T W_a \omega_k^T}{4N\rho_k} W_c. \quad (190)$$

Provided the extrapolation states are selected such that $s_k \in \overline{B}(0,\chi)$, $\forall k = 1,\ldots,N$, the orbital derivative in (183) can be bounded as

$$\dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right) \leq -k_c\underline{c}\left\|\tilde{W}_c\right\|^2 - (k_{a1}+k_{a2})\left\|\tilde{W}_a\right\|^2$$
$$+ k_c\iota_8\overline{\epsilon}\left\|\tilde{W}_c\right\| + k_c\iota_5\left\|\tilde{W}_a\right\|^2 + (k_c\iota_6 + k_{a1})\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| + \left(k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\|,$$

for all $t \geq 0$, where $\iota_5,\ldots,\iota_8$ are positive constants that are independent of the learning gains, $\overline{W}$ denotes an upper bound on the norm of the ideal weights W, and $\underline{c_3} = \min_{t\geq 0}\lambda_{\min}\left\{\left(\frac{\beta}{2k_c}\Gamma^{-1}(t) + \frac{1}{2N}\sum_{k=1}^{N}\frac{\omega_k\omega_k^T}{\rho_k}\right)\right\}$. Assumption 4.8.1 and (148) guarantee that $\underline{c_3} > 0$.

B.2 Derivation for candidate Lyapunov function

The candidate Lyapunov function for the overall system is then defined as

$$V_L\left(Z,t\right) \coloneqq \mathcal{V}\left(s\right) + \Theta\left(\tilde{W}_c, \tilde{W}_a, t\right) + V_{se}\left(Z_1\right), \tag{191}$$

where $Z \coloneqq \begin{bmatrix} s^T & \tilde{s}_1^T & r^T & \eta^T & \tilde{W}_c^T & \tilde{W}_a^T \end{bmatrix}^T$. The orbital derivative of the candidate Lyapunov function along the trajectories of (95), (100),(101),(109), (138), (139), (140), under the controller (141), is defined as

$$\dot{V}_L(Z,t) = \dot{\mathcal{V}}\left(s, \tilde{s}, \tilde{W}_a\right) + \dot{V}_{se}\left(Z_1, s, \tilde{s}, \tilde{W}_a\right) + \dot{\Theta}\left(\tilde{W}_c, \tilde{W}_a, t\right). \tag{192}$$

$$\begin{aligned}
\dot{V}_L\left(Z,t\right) \leq & -W\left(s\right) + \iota_1\overline{\epsilon} + \iota_2\left\|\tilde{s}\right\|\left\|\tilde{W}_a\right\| + \iota_3\left\|\tilde{W}_a\right\| + \iota_4\left\|\tilde{s}\right\| \\
& - k_c\underline{c}\left\|\tilde{W}_c\right\|^2 - \left(k_{a1}+k_{a2}\right)\left\|\tilde{W}_a\right\|^2 + k_c\iota_8\overline{\epsilon}\left\|\tilde{W}_c\right\| + k_c\iota_5\left\|\tilde{W}_a\right\|^2 \\
& + \left(k_c\iota_6 + k_{a1}\right)\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| + \left(k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\| \\
& - \alpha^3\|\tilde{s}_1\|^2 - \left(k - \varpi_1\varpi_4\right)\|r\|^2 - \left(\beta_1 - \alpha\right)\|\eta\|^2 + \varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\tilde{s}_1\| \\
& + \varpi_1\varpi_4\alpha\|\|r\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \varpi_2\|r\|\|\tilde{W}_a\| + \varpi_3\|r\|,
\end{aligned} \tag{193}$$

so,

$$\begin{aligned}
\dot{V}_L\left(Z,t\right) \leq & -W\left(s\right) + \iota_1\overline{\epsilon} + \left(\iota_2\left\|\tilde{s}\right\| + \iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\| + \iota_4\left\|\tilde{s}\right\| \\
& - k_c\underline{c}\left\|\tilde{W}_c\right\|^2 - \left(k_{a1}+k_{a2}-k_c\iota_5\right)\left\|\tilde{W}_a\right\|^2 + k_c\iota_8\overline{\epsilon}\left\|\tilde{W}_c\right\| + \left(k_c\iota_6 + k_{a1}\right)\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| \\
& - \alpha^3\left\|\tilde{s}_1\right\|^2 - \left(k - \varpi_1\varpi_4\right)\|r\|^2 - \left(\beta_1 - \alpha\right)\|\eta\|^2 \\
& + \left(1 + \varpi_4 + \varpi_4\alpha\right)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \varpi_2\|r\|\|\tilde{W}_a\| + \varpi_3\|r\|,
\end{aligned} \tag{194}$$

so,

$$\dot{V}_L(Z,t) \leq -W(s) - k_{c\underline{c}} \left\| \tilde{W}_c \right\|^2 - (k_{a1} + k_{a2} - k_c\iota_5) \left\| \tilde{W}_a \right\|^2 - \alpha^3 \|\tilde{s}_1\|^2 - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + \iota_1\overline{\epsilon} + \left(\iota_2 \|\tilde{s}\| + \iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right) \left\| \tilde{W}_a \right\| + \iota_4 \|\tilde{s}\| + k_c\iota_8\overline{\epsilon} \left\| \tilde{W}_c \right\| + (k_c\iota_6 + k_{a1}) \left\| \tilde{W}_c \right\| \left\| \tilde{W}_a \right\| + \left(1 + \varpi_4 + \varpi_4\alpha\right)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \varpi_2\|r\|\|\tilde{W}_a\| + \varpi_3\|r\|, \quad (195)$$

so,

$$\dot{V}_L(Z,t) \leq -W(s) - k_{c\underline{c}} \left\| \tilde{W}_c \right\|^2 - (k_{a1} + k_{a2} - k_c\iota_5) \left\| \tilde{W}_a \right\|^2 - \alpha^3 \|\tilde{s}_1\|^2 - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + \left(\iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right) \left\| \tilde{W}_a \right\| r + \iota_4(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| + (\varpi_3 + \iota_4\varpi_4)\|r\| + \iota_4\varpi_4\|\eta\| + k_c\iota_8\overline{\epsilon} \left\| \tilde{W}_c \right\| + (k_c\iota_6 + k_{a1}) \left\| \tilde{W}_c \right\| \left\| \tilde{W}_a \right\| + (1 + \varpi_4 + \varpi_4\alpha)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \iota_2(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| \left\| \tilde{W}_a \right\| + \left(\iota_2\varpi_4 + \varpi_2\right)\|r\| \left\| \tilde{W}_a \right\| + \iota_2\varpi_4\|\eta\| \left\| \tilde{W}_a \right\| + \iota_1\overline{\epsilon}, \quad (196)$$

so,

$$\dot{V}_L(Z,t) \leq -W(s) - k_{c\underline{c}} \left\| \tilde{W}_c \right\|^2 - (k_{a1} + k_{a2} - k_c\iota_5) \left\| \tilde{W}_a \right\|^2 - \alpha^3 \|\tilde{s}_1\|^2 - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + (k_c\iota_6 + k_{a1}) \left\| \tilde{W}_c \right\| \left\| \tilde{W}_a \right\| + \iota_2(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| \left\| \tilde{W}_a \right\| + \left(\iota_2\varpi_4 + \varpi_2\right)\|r\| \left\| \tilde{W}_a \right\| + \iota_2\varpi_4\|\eta\| \left\| \tilde{W}_a \right\| + (1 + \varpi_4 + \varpi_4\alpha)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \iota_4\varpi_4\|\eta\| + (\varpi_3 + \iota_4\varpi_4)\|r\| + \left(\iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right) \left\| \tilde{W}_a \right\| + k_c\iota_8\overline{\epsilon} \left\| \tilde{W}_c \right\| + \iota_4(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| + \iota_1\overline{\epsilon}, \quad (197)$$

Let $\mathcal{C} \subset \mathbb{R}^{5n}$ be a compact set defined as $\mathcal{C} \coloneqq \{(s, \tilde{s}_1, \eta, r) \in \mathbb{R}^{5n} | \|s\| + \|\tilde{s}_1\|(1 + \varpi_4(1+\alpha)) + \varpi_4(\|r\|\|\| + \|\eta\|) \leq \chi\}$. Using (180), whenever, $(s, \tilde{s}_1, \eta, r) \in \mathcal{C}$, it can be concluded that $s, \hat{s} \in \overline{B}(0, \chi)$. As a result, (151), (153), and (154)
imply that whenever $Z \in \mathcal{C} \times \mathbb{R}^{2L}$, the orbital derivative can be bounded as

$$\begin{aligned}
\dot{V}_L(Z,t) \leq & -W(s) - k_c \underline{c}_3 \left\|\tilde{W}_c\right\|^2 - (k_{a1} + k_{a2} - k_c\iota_5)\left\|\tilde{W}_a\right\|^2 - \alpha^3 \|\tilde{s}_1\|^2 \\
& - (k - \varpi_1\varpi_4)\|r\|^2 - (\beta_1 - \alpha)\|\eta\|^2 + (k_c\iota_6 + k_{a1})\left\|\tilde{W}_c\right\|\left\|\tilde{W}_a\right\| \\
& + \iota_2(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\|\left\|\tilde{W}_a\right\| + \left(\iota_2\varpi_4 + \varpi_2\right)\|r\|\left\|\tilde{W}_a\right\| + \iota_2\varpi_4\|\eta\|\left\|\tilde{W}_a\right\| \\
& + (1 + \varpi_4 + \varpi_4\alpha)\varpi_1\|r\|\|\tilde{s}_1\| + \varpi_1\varpi_4\|r\|\|\eta\| + \iota_4\varpi_4\|\eta\| + (\varpi_3 + \iota_4\varpi_4)\|r\| \\
& + \left(\iota_3 + k_c\iota_7 + k_{a2}\overline{W}\right)\left\|\tilde{W}_a\right\| + k_c\iota_8\bar{\epsilon}\left\|\tilde{W}_c\right\| + \iota_4(1 + \varpi_4 + \varpi_4\alpha)\|\tilde{s}_1\| + \iota_1\bar{\epsilon},
\end{aligned}$$

which yields

$$\dot{V}_L(Z,t) \leq -W(s) - z^T\left(\frac{M + M^T}{2}\right)z + Pz + \iota_1\bar{\epsilon},$$

where $z \coloneqq \begin{bmatrix}\left\|\tilde{W}_c\right\| & \left\|\tilde{W}_a\right\| & \|\tilde{s}_1\| & \|r\| & \|\eta\|\end{bmatrix}^T$,

$$P = \begin{bmatrix}k_c\iota_8\bar{\epsilon} & \left(k_c\iota_7 + \iota_3 + k_{a2}\overline{W}\right) & \iota_4(1 + \varpi_4 + \varpi_4\alpha) & (\varpi_3 + \iota_4\varpi_4) & \iota_4\varpi_4\end{bmatrix},$$

and

$$M = \begin{bmatrix}
[k_c\underline{c}_3 & -(k_c\iota_6 + k_{a1}) & 0 & 0 & 0 \\
0 & (k_{a1} + k_{a2} - k_c\iota_5) & -\iota_2(1 + \varpi_4 + \varpi_4\alpha) & -(\iota_2\varpi_4 + \varpi_2) & -\iota_2\varpi_4 \\
0 & 0 & \alpha^3 & -\varpi_1(1 + \varpi_4 + \varpi_4\alpha) & 0 \\
0 & 0 & 0 & (k - \varpi_1\varpi_4) & -\varpi_1\varpi_4 \\
0 & 0 & 0 & 0 & (\beta_1 - \alpha)].
\end{bmatrix}.$$

Provided the matrix $M + M^T$ is PD,

$$\dot{V}_L(Z,t) \leq -W(s) - \underline{M}\|z\|^2 + \bar{P}\|z\| + \iota_1\bar{\epsilon},$$

where $\underline{M} \coloneqq \lambda_{\min}\left\{\frac{M+M^T}{2}\right\}$. Letting $\underline{M} =: \underline{M}_1 + \underline{M}_2$ and letting $\mathcal{W} : \mathbb{R}^{5n+2L} \to \mathbb{R}$ be defined as $\mathcal{W}(Z) = -W(s) - \underline{M}_1\|z\|^2$, the time derivative of (155) bounded as

$$\dot{V}_L(Z,t) \leq -\mathcal{W}(Z), \tag{198}$$

$\forall \|z\| > \frac{1}{2}\left(\frac{\overline{P}}{\underline{M}_2} + \sqrt{\frac{\overline{P}^2}{\underline{M}_2^2} + \frac{\iota_1^2 \overline{\epsilon}^2}{\underline{M}_2^2}}\right) = \mu, Z \in \overline{B}\left(0, \bar{\chi}\right)$, for all $t \geq 0$, and some $\bar{\chi}$ such that $\bar{B}(0, \bar{\chi}) \subseteq \mathcal{C} \times \mathbb{R}^{2L}$.

Using the bound in (148) and the fact that the converse Lyapunov function is guaranteed to be time-independent, radially unbounded, and PD, Lemma 4.3 can be invoked to conclude that

$$\underline{v}\left(\|Z\|\right) \leq V_L\left(Z, t\right) \leq \overline{v}\left(\|Z\|\right), \tag{199}$$

for all $t \in \mathbb{R}_{\geq 0}$ and for all $Z \in \mathbb{R}^{5n+2L}$, where $\underline{v}, \overline{v} : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ are class $\mathcal{K}$ functions.

Provided the learning gains, the domain radii χ and $\bar{\chi}$, and the basis functions for function approximation are selected such that $M + M^T$ is PD and $\mu < \overline{v}^{-1}\left(\underline{v}\left(0, \bar{\chi}\right)\right)$, Theorem 4.18 in [156] can be invoked to conclude that Z is uniformly ultimately bounded. Since the estimates W_a approximate the ideal weights W, the policy $\hat{u}$ approximates the optimal policy u^*.

VITA

S M Nahid Mahmud

Candidate for the Degree of

Masters of Science

Thesis: SAFETY-AWARE MODEL-BASED REINFORCEMENT LEARNING USING BARRIER TRANSFORMATION

Major Field: Mechanical and Aerospace Engineering

Biographical:

Education:

Completed the requirements for the Masters of Science in Mechanical Engineering at Oklahoma State University, Stillwater, Oklahoma in May, 2021.

Received a Bachelors of Science in Mechanical Engineering at Islamic University of Technology, Gazipur, Bangladesh in December, 2015.

Experience:

Graduate Research Assistant, Systems, Cognition, and Control Laboratory, Oklahoma State University.

Graduate Teaching Assistant, Mechanical and Aerospace Engineering Department, Oklahoma State University.

Adjunct Lecturer, Mechanical Engineering Department, Sonargaon University.

Administrator, Moon Engineering Works.