Intrinsic Closed-Loop Practical Asymptotic Stability in Discrete-Time Reinforcement Learning

Jan de Priester, Ricardo G. Sanfelice

Keywords: Closed-Loop Stability, Input-to-State Stability, Practical Asymptotic Stability, Continuous Control, Lyapunov Functions, Reinforcement Learning.

Summary

Deep reinforcement learning (RL) has achieved impressive empirical success in complex continuous control tasks, many of which are fundamentally set-stabilization problems. To safely deploy learned feedback laws, existing methods often modify RL algorithms to enforce asymptotic stability, for example, by embedding Lyapunov constraints or applying external safety filters. While these approaches provide stability certificates, they alter the core RL objective, which can restrict natural exploration and complicate training.

This paper offers a complementary perspective: while such algorithmic modifications are valuable for training-time safety during exploration, modifying the core RL objective is often unnecessary to guarantee the stability of the deployed policy. We prove that the unmodified discounted RL objective intrinsically guarantees uniform practical asymptotic stability for deterministic continuous control tasks. By adapting the discrete-time input-to-state stability (ISS) framework, we model the suboptimality induced by both the discount factor and the neural network approximation error as bounded disturbances. Our analysis demonstrates that near-optimal parameterized policies inherently steer the closed-loop system into a bounded neighborhood of the target set. The size of this neighborhood vanishes in the limit as the approximation error approaches zero and, subsequently, the discount factor approaches one.

Contribution(s)

1. We prove that the exact optimal policy for the discounted reinforcement learning objective intrinsically guarantees closed-loop uniform practical asymptotic stability (UPAS) in deterministic continuous control tasks.
   Context: Prior work typically enforces stability by modifying the RL algorithm (for example, via Lyapunov constraints (Han et al., 2020; Osinenko et al., 2020)) or applying external safety filters (Berkenkamp et al., 2017). While these modifications are valuable for training-time safety during exploration, our analysis evaluates the unmodified RL objective, demonstrating that modifying the core objective is often unnecessary to guarantee the stability of the deployed policy.
2. We extend this practical stability guarantee to near-optimal parameterized policies, proving that the ultimate steady-state error of the closed-loop system is explicitly bounded by the discount factor and the function approximation error.
   Context: The discrete-time input-to-state stability (ISS) framework (Jiang & Wang, 2001) has been used to analyze the effect of discounting in model predictive control (MPC) (Postoyan et al., 2016). We extend this approach to the deep RL setting.

Intrinsic Closed-Loop Practical Asymptotic Stability in Discrete-Time Reinforcement Learning

Jan de Priester[1], **Ricardo G. Sanfelice**[1]
{jadeprie,ricardo}@ucsc.edu

[1]**Department of Electrical and Computer Engineering, University of California, Santa Cruz**

Abstract

Deep reinforcement learning (RL) has achieved impressive empirical success in complex control tasks, many of which are fundamentally *set-stabilization* problems. To safely deploy learned policies, existing methods often modify RL algorithms to enforce formal stability guarantees. However, these algorithmic modifications alter the core RL objective, which can restrict natural exploration and complicate training. This paper offers a complementary perspective: while such algorithmic modifications are valuable for training-time safety during exploration, modifying the core RL objective is often unnecessary to guarantee the stability of the deployed policy. We prove that the unmodified discounted RL objective intrinsically guarantees closed-loop *practical asymptotic stability* in deterministic continuous control settings. By adapting the discrete-time input-to-state stability (ISS) framework and drawing inspiration from discounted model predictive control (MPC), we model the suboptimality induced by the discount factor and the neural network approximation error as bounded disturbances. We establish that near-optimal parameterized feedback laws steer the closed-loop system into a bounded neighborhood of the target set. The size of this neighborhood vanishes in the limit as the approximation error approaches zero and, subsequently, the discount factor approaches one.

1 Introduction

Deep reinforcement learning (RL) has empirically demonstrated impressive results in complex control tasks, such as dynamic legged locomotion (Hwangbo et al., 2019), acrobatic maneuvers (Peng et al., 2018), high-speed autonomous flight (Kaufmann et al., 2023), and dexterous manipulation (Andrychowicz et al., 2020). Fundamentally, these tasks are *set-stabilization* problems. Rather than driving a system to a single static equilibrium point, these tasks require the agent to converge to and indefinitely remain within a geometric *target set* in the state space. For example, dynamic legged locomotion requires stabilizing a continuous, repeating motion (a limit cycle). Similarly, autonomous drone racing involves converging to a spatial racing line, while dexterous manipulation requires steering an object into a manifold of valid target poses. While RL empirically excels at these tasks, formally guaranteeing closed-loop stability properties native to control theory remains challenging. Providing such formal guarantees is essential for safely deploying agents in real-world, safety-critical applications, such as autonomous driving or human-robot interaction.

To address this, a prominent branch of the literature modifies the RL objective to explicitly enforce formal stability guarantees. Many approaches target strict *asymptotic stability*, which guarantees that the closed-loop system eventually reaches and maintains a target behavior. Such approaches include jointly training neural Lyapunov functions (Richards et al., 2018; Chang et al., 2019; Han et al., 2020; Dai et al., 2021) or applying safety filters (Berkenkamp et al., 2017). Other methods relax this

requirement to practical stability, but still achieve it by embedding Lyapunov decrease constraints into the learning process (Osinenko et al., 2020; Han et al., 2021). While these techniques lead to asymptotic stability-like properties, they fundamentally alter the core RL objective, which can restrict natural exploration and complicate training.

In this work, we offer a complementary perspective: while such algorithmic modifications are valuable for training-time safety during exploration, modifying the core RL objective is often unnecessary to guarantee the stability of the deployed policy. We argue that *practical asymptotic stability*—where a system eventually enters and indefinitely remains within a small, bounded neighborhood of the target—is an intrinsic property of the near-optimal solution to the RL problem. In classical control theory, asymptotic stability is an infinite-horizon property. In contrast, RL optimizes an expected *discounted* return. As established in the discounted model predictive control (MPC) literature, see, e.g., Postoyan et al. (2016), a discount factor effectively truncates the optimization horizon. This yields a myopic agent that prioritizes near-term performance and mathematically settles for being "close enough" to the target. Thus, the native steady-state error observed in RL is not an algorithmic failure, but the direct consequence of the discounted objective.

To formally establish these intrinsic guarantees, we adapt the discrete-time input-to-state stability (ISS) framework in Jiang & Wang (2001) and draw inspiration from discounted MPC. While MPC typically relies on exact online optimization, practical RL uses parameterized neural networks that produce *near-optimal* solutions at best (Kidger & Lyons, 2020). Using ISS, we bridge this gap by treating both the truncated horizon (due to discounting) and the suboptimality of the agent (due to function approximation) as bounded disturbances to the ideal closed-loop system. To build this theoretical foundation, we first establish essential properties of the RL problem, including the continuity of the value functional and the uniform practical asymptotic stability of the exact optimal feedback law. We then extend our analysis to the approximate setting and derive near-optimality bounds for parameterized neural network-based feedback laws. Integrating these results, we prove that deep RL intrinsically provides practical closed-loop stability guarantees.

The remainder of this paper is organized as follows. Section 2 introduces the necessary notation and mathematical preliminaries. Section 3 formulates the ideal discrete-time RL problem and proves that the exact optimal feedback law intrinsically guarantees uniform practical asymptotic stability. Section 4 extends these stability guarantees to the practical approximate RL setting, where parameterized neural networks are used. Finally, Section 5 provides concluding remarks. The proofs will be uploaded elsewhere.

2 Preliminaries and Notation

2.1 Notation

The following notation is used throughout the paper. The n-dimensional Euclidean space is denoted by $\mathbb{R}^n$. The positive real numbers are denoted by $\mathbb{R}_{>0}$, i.e., $\mathbb{R}_{>0} := (0, \infty)$. The nonnegative real numbers are denoted by $\mathbb{R}_{\geq 0}$, i.e., $\mathbb{R}_{\geq 0} := [0, \infty)$. The natural numbers including 0 are denoted by $\mathbb{N}$, i.e., $\mathbb{N} := \{0, 1, 2, \dots\}$. The natural numbers excluding 0 are denoted by $\mathbb{N}_{>0}$, i.e., $\mathbb{N}_{>0} := \{1, 2, \dots\}$. The Euclidean norm of the vector x is denoted by $|x|$. The distance from x to the nonempty set S is denoted by $|x|_S$, which is given by $\inf_{y \in S} |x - y|$. A sequence of vectors $v_0, v_1, \dots$ is denoted by $\{v_k\}_{k \in \mathbb{N}}$.

2.2 Preliminaries

Definition 2.1 (Continuous Functions)**.** *Let $\mathcal{X}$ be a compact subset of $\mathbb{R}^n$ and $\mathcal{U}$ a compact subset of $\mathbb{R}^m$. We denote by $\mathcal{C}(\mathcal{X})$ the space of all real-valued continuous functions on $\mathcal{X}$, endowed with the supremum norm*

$$|v|_\infty := \sup_{x \in \mathcal{X}} |v(x)|.$$

Similarly, $\mathcal{C}(\mathcal{X},\mathcal{U})$ denotes the space of all continuous functions $\kappa\colon \mathcal{X} \to \mathcal{U}$ endowed with the supremum norm

$$|\kappa|_\infty := \sup_{x\in\mathcal{X}} |\kappa(x)|.$$

Equipped with the supremum norm, $\mathcal{C}(\mathcal{X})$ and $\mathcal{C}(\mathcal{X},\mathbb{R}^m)$ are Banach spaces, and $\mathcal{C}(\mathcal{X},\mathcal{U})$ is a complete metric space[1]; see, for example, Theorem 7.15 in Rudin (1976). Moreover, $\mathcal{X}$ and $\mathcal{U}$ are metric spaces endowed with the Euclidean norm, inherited from $\mathbb{R}^n$ and $\mathbb{R}^m$, respectively.

Definition 2.2 (Compactness in Function Space)**.** *A subset $\mathbb{K} \subset \mathcal{C}(\mathcal{X},\mathcal{U})$ is said to be compact with respect to the supremum norm if it is sequentially compact. That is, for every sequence $\{\kappa_n\}_{n\in\mathbb{N}}$ contained in $\mathbb{K}$, there exists a subsequence $\{\kappa_{n_j}\}_{j\in\mathbb{N}}$ and a limit function $\kappa^* \in \mathbb{K}$ such that*

$$\lim_{j\to\infty} |\kappa_{n_j} - \kappa^*|_\infty = 0.$$

Remark. *An example of such a compact function space is the class of feedforward neural networks with a fixed architecture, continuous activation functions, and bounded weights, which are commonly used to parameterize continuous control policies in deep RL (Peng et al., 2018; Hwangbo et al., 2019; Kaufmann et al., 2023).*

Definition 2.3 (Class-$\mathcal{K}$ Functions)**.** *A function $\alpha\colon \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class-$\mathcal{K}$ function, also written $\alpha \in \mathcal{K}$, if α is zero at zero, continuous, and strictly increasing.*

Definition 2.4 (Class-$\mathcal{KL}$ Functions)**.** *A function $\beta\colon \mathbb{R}_{\geq 0}\times\mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is a class-$\mathcal{KL}$ function, also written $\beta \in \mathcal{KL}$, if it is nondecreasing in its first argument, nonincreasing in its second argument, $\lim_{r\searrow 0} \beta(r,s) = 0$ for each $s \in \mathbb{R}_{\geq 0}$, and $\lim_{s\to\infty} \beta(r,s) = 0$ for each $r \in \mathbb{R}_{\geq 0}$.*

3 Discrete-Time Reinforcement Learning

3.1 System Definition

We consider a discrete-time (DT) controlled dynamical system of the form

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

where $x \in \mathcal{X}$ denotes the state variable, with $\mathcal{X}$ being the state space, $x^+ \in \mathcal{X}$ is the next value of the state, $u \in \mathcal{U}$ is the control input, with $\mathcal{U}$ being the action space, and $g\colon \mathcal{X}\times\mathcal{U} \to \mathcal{X}$ is the DT dynamics. A solution to the DT system in (1) is defined as follows.

Definition 3.1 (Solution to the DT System)**.** *Given an initial condition $x_0 \in \mathcal{X}$ and an input sequence $\{u_k\}_{k\in\mathbb{N}}$, a solution to the DT system in (1) is the sequence $\{x_k\}_{k\in\mathbb{N}}$ defined recursively by $x_{k+1} = g(x_k,u_k)$ for each $k \in \mathbb{N}$.*[2]

Let $R\colon \mathcal{X}\times\mathcal{U} \to \mathbb{R}$ be a continuous function capturing the reward and let $\gamma \in (0,1)$ represent the discount factor. The total discounted reward of the DT system in (1) is described by the *reward functional* defined as follows.

Definition 3.2 (Reward Functional)**.** *Fix $\gamma \in (0,1)$. For a given initial condition $x_0 \in \mathcal{X}$ and an input sequence $\mathbf{u} := \{u_k\}_{k\in\mathbb{N}}$ generating a solution $\{x_k\}_{k\in\mathbb{N}}$ of the DT system in (1), the reward functional*[3] *is defined as*

$$J_\gamma(x_0,\mathbf{u}) := \sum_{k=0}^{\infty} \gamma^k R(x_k,u_k).$$

Next, we assign the control input to a feedback law, defined as follows.

[1] The space $\mathcal{C}(\mathcal{X},\mathcal{U})$ is not a Banach space as $\mathcal{U}$ is not a vector space.

[2] By definition, all solutions are *complete*, meaning solutions are defined for all future discrete time steps $k \in \mathbb{N}$.

[3] For each x_0, the mapping $\mathbf{u} \mapsto J_\gamma(x_0,\mathbf{u})$ is called a *functional* because its argument is a function (or sequence) rather than a finite-dimensional vector.

Definition 3.3 (Feedback Law and Closed-Loop System)**.** *A feedback law is a function $\kappa \in \mathcal{C}(\mathcal{X},\mathcal{U})$. The DT controlled dynamical system in (1) subject to a feedback law κ is given by the DT closed-loop system*

$$x^+ = g(x, \kappa(x)). \tag{2}$$

Note that, for each feedback law $\kappa \in \mathcal{C}(\mathcal{X},\mathcal{U})$ and initial condition $x_0 \in \mathcal{X}$, there exists a unique solution $\{x_k^\kappa\}_{k\in\mathbb{N}}$ to the DT closed-loop system in (2) by the fact that the mapping $x \mapsto g(x, \kappa(x))$ is single-valued. Furthermore, given $(x_0, \kappa) \in \mathcal{X} \times \mathcal{C}(\mathcal{X},\mathcal{U})$, the total discounted reward of the DT closed-loop system in (2) is described by the *value functional* defined as follows.

Definition 3.4 (Value Functional under Feedback)**.** *Fix $\gamma \in (0, 1)$. Given an initial condition $x_0 \in \mathcal{X}$ and a feedback law $\kappa \in \mathcal{C}(\mathcal{X},\mathcal{U})$ generating a solution $\{x_k^\kappa\}_{k\in\mathbb{N}}$ to the DT closed-loop system in (2), the value functional $\mathcal{V}_\gamma \colon \mathcal{X} \times \mathcal{C}(\mathcal{X},\mathcal{U}) \to \mathbb{R}$ is defined as*

$$\mathcal{V}_\gamma(x_0, \kappa) := \sum_{k=0}^{\infty} \gamma^k R\big(x_k^\kappa, \kappa(x_k^\kappa)\big). \tag{3}$$

We make the following assumption on the DT dynamics g, the reward function R, and the state and action spaces $\mathcal{X}$ and $\mathcal{U}$.

Assumption 3.1. *The functions $g \colon \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ and $R \colon \mathcal{X} \times \mathcal{U} \to \mathbb{R}$ are continuous, and the sets $\mathcal{X}$ and $\mathcal{U}$ are compact.*

Remark. *In physical applications such as robotics, assuming the state and action spaces $\mathcal{X}$ and $\mathcal{U}$ are compact sets is not restrictive. In particular, workspaces are often bounded, and actuators have physical saturation limits. Furthermore, the DT dynamics g are typically obtained by discretizing continuous-time models. Because the underlying continuous-time dynamics of many physical systems, such as robotic manipulators and wheeled mobile robots, are typically smooth, discretizing their dynamics over a compact domain inherently yields a continuous mapping g. Lastly, a common choice of the reward function R is the negated Euclidean distance to a target set, which is a continuous function.*

3.2 Regularity of the Value Functional

In this section, useful properties of the value functional in (3) are derived. Establishing these properties is necessary to guarantee the existence of an optimal feedback law in Section 3.3 and to relate expected performance to pointwise error bounds when evaluating feedforward neural network-based feedback laws in Section 4.

First, we characterize the regularity of the closed-loop trajectories. Under Assumption 3.1, the evaluation map $(\kappa, x) \mapsto \kappa(x)$ is jointly continuous. Namely, for each $(\kappa, x) \in \mathcal{C}(\mathcal{X},\mathcal{U}) \times \mathcal{X}$ and any $\varepsilon > 0$, there exists a $\delta > 0$ such that for any perturbed feedback law $\tilde{\kappa} \in \mathcal{C}(\mathcal{X},\mathcal{U})$ and perturbed state $\tilde{x} \in \mathcal{X}$ satisfying $|\tilde{\kappa} - \kappa|_\infty < \delta$ and $|\tilde{x} - x| < \delta$, the resulting variation in the control input satisfies $|\tilde{\kappa}(\tilde{x}) - \kappa(x)| < \varepsilon$. Consequently, solutions to the DT closed-loop system in (2) depend continuously on the initial condition and the feedback law.

This continuous dependence on the initial condition and the feedback law, along with the compactness of $\mathcal{X}$ and $\mathcal{U}$, the continuity of the reward function, and the discount factor, is sufficient to guarantee the continuity of the value functional, as we show next.

Theorem 3.2 (Continuity of the Value Functional)**.** *Suppose Assumption 3.1 holds. Then, for each $\gamma \in (0, 1)$, the value functional $\mathcal{V}_\gamma \colon \mathcal{X} \times \mathcal{C}(\mathcal{X},\mathcal{U}) \to \mathbb{R}$ is continuous with respect to the supremum norm.*

Lastly, the following lemma confirms that the value functional satisfies the Bellman equation. This recursive property is used in the subsequent stability analysis to evaluate the difference of the value functional along solutions to the closed-loop system.

Lemma 3.3 (Bellman Equation)**.** *For each $\gamma \in (0,1)$, the value functional $\mathcal{V}_\gamma \colon \mathcal{X} \times \mathcal{C}(\mathcal{X},\mathcal{U}) \to \mathbb{R}$ satisfies the Bellman equation*

$$\mathcal{V}_\gamma(x_0,\kappa) = R(x_0,\kappa(x_0)) + \gamma \mathcal{V}_\gamma(g(x_0,\kappa(x_0)),\kappa), \tag{4}$$

for each $(x_0,\kappa) \in \mathcal{X} \times \mathcal{C}(\mathcal{X},\mathcal{U})$.

3.3 Reinforcement Learning Formulation

In this section, we formulate the RL problem and prove the existence of an optimal feedback law. We restrict the optimization to a specific class of feedback laws, defined as follows.

Definition 3.5 (Admissible Feedback Laws)**.** *A set of feedback laws $\mathbb{K} \subset \mathcal{C}(\mathcal{X},\mathcal{U})$ is called admissible if it is a non-empty subset that is compact with respect to the supremum norm.*

Because the value functional $\mathcal{V}_\gamma$ depends on the initial state $x \in \mathcal{X}$, evaluating the global performance of a feedback law requires aggregating these pointwise values into a single scalar objective. To perform this aggregation, we define an *admissible* initial state distribution to weight the relative importance of each initial state.

Definition 3.6 (Admissible Initial State Distribution)**.** *A function $p_0 \colon \mathcal{X} \to \mathbb{R}_{>0}$ is called an admissible initial state distribution if it is a continuous, strictly positive function that satisfies*

$$\int_{\mathcal{X}} p_0(x)\,\mathrm{d}x = 1.$$

Problem 1 (Reinforcement Learning Problem)**.** *Given $\gamma \in (0,1)$, an admissible initial state distribution $p_0 \colon \mathcal{X} \to \mathbb{R}_{>0}$, and an admissible set of feedback laws $\mathbb{K}$, the RL problem is defined as*

$$\max_{\kappa \in \mathbb{K}} \mathcal{J}_\gamma(\kappa), \tag{5}$$

where $\mathcal{J}_\gamma \colon \mathbb{K} \to \mathbb{R}$ is the expected value functional defined as

$$\mathcal{J}_\gamma(\kappa) := \int_{\mathcal{X}} \mathcal{V}_\gamma(x,\kappa) p_0(x)\,\mathrm{d}x \quad \forall \kappa \in \mathbb{K}. \tag{6}$$

In Problem 1, we aim to find a feedback law by searching over the admissible set of feedback laws to maximize the expected value functional over the state space, weighting the value of each initial state by the initial state distribution.

Remark. *While RL typically maximizes value functionals over stochastic system dynamics and stochastic feedback laws (Sutton & Barto, 2018), the objective functional $\mathcal{J}_\gamma$ in Problem 1 assumes deterministic dynamics and deterministic feedback laws. This deterministic framework aligns with practical continuous control: physical systems are typically deterministic, and the stochastic feedback laws used for exploration during training typically converge to deterministic feedback laws as they approach optimality. Under these conditions, the typical RL objective and $\mathcal{J}_\gamma$ share the same optimal solution. Consequently, analyzing the closed-loop properties of the maximizer of $\mathcal{J}_\gamma$ directly characterizes the behavior of RL in deterministic settings.*

The following lemma shows that the expected value functional in Problem 1 is continuous.

Lemma 3.4 (Continuity of the Expected Value Functional)**.** *Suppose Assumption 3.1 holds. Then, for each $\gamma \in (0,1)$, the expected value functional $\mathcal{J}_\gamma \colon \mathbb{K} \to \mathbb{R}$ is continuous with respect to the supremum norm.*

To conclude this section, we show that there exists an optimal feedback law that solves Problem 1.

Theorem 3.5 (Existence of the Optimal Feedback Law)**.** *Suppose Assumption 3.1 holds. Then, for each $\gamma \in (0,1)$, there exists an optimal feedback law $\kappa_\gamma^* \in \mathbb{K}$ that solves Problem 1; that is,*

$$\kappa_\gamma^* \in \arg\max_{\kappa \in \mathbb{K}} \mathcal{J}_\gamma(\kappa). \tag{7}$$

3.4 Properties of the Optimal Feedback Law

In this section, we study the properties of the optimal feedback law that solves Problem 1. We start by defining the *value function* for a given feedback law (instead of a *functional*) for notational brevity; see Footnote 3.

Definition 3.7 (Value Function of a Fixed Feedback Law)**.** *Fix $\gamma \in (0,1)$. Given a fixed feedback law $\kappa \in \mathcal{C}(\mathcal{X},\mathcal{U})$, its corresponding value function $V_\gamma^\kappa \colon \mathcal{X} \to \mathbb{R}$ is defined as*

$$V_\gamma^\kappa(x) := \mathcal{V}_\gamma(x,\kappa) \quad \forall x \in \mathcal{X}. \tag{8}$$

Next, we define the notion of *proper indicator* (Goebel et al., 2012, Definition 7.9).

Definition 3.8 (Proper Indicator)**.** *Given a compact target set $\mathcal{A} \subset \mathcal{X}$, a continuous function $\omega \colon \mathcal{X} \to \mathbb{R}_{\geq 0}$ is called a proper indicator of $\mathcal{A}$ on $\mathcal{X}$ if it is positive definite with respect to $\mathcal{A}$, that is, $\omega(x) = 0$ if and only if $x \in \mathcal{A}$.*

Remark. *In the context of RL, a proper indicator quantifies how far the agent deviates from the desired behavior (cf. stability of a desired target set). This formulation naturally captures a wide variety of standard continuous control tasks. For example, whether the task involves classical point stabilization, navigating to a broader goal region, or converging to a limit cycle (e.g., a periodic walking gait), the target set represents the desired state or motion, and the proper indicator corresponds to the penalty engineered into the reward function to drive the system toward this target.*

Inspired by Grimm et al. (2005); Postoyan et al. (2016), we make the following assumptions on the reward function and the set of admissible feedback laws.

Assumption 3.6 (Control Properties)**.** *Given a reward function $R \colon \mathcal{X} \times \mathcal{U} \to \mathbb{R}$, a compact target set $\mathcal{A} \subset \mathcal{X}$, and a proper indicator $\omega \colon \mathcal{X} \to \mathbb{R}_{\geq 0}$ of $\mathcal{A}$ on $\mathcal{X}$, let $\overline{R}_\mathcal{A} := \max_{(x,u)\in\mathcal{A}\times\mathcal{U}} R(x,u)$. We assume the following properties hold:*

(i) ***Strict Reward Penalty:*** *There exists $\underline{\alpha} \in \mathcal{K}$ such that*

$$R(x,u) - \overline{R}_\mathcal{A} \leq -\underline{\alpha}(\omega(x)) \quad \forall (x,u) \in \mathcal{X} \times \mathcal{U}.$$

(ii) ***Stabilizing Baseline:*** *There exists $\kappa_s \in \mathbb{K}$ for which there exists a class-$\mathcal{K}$ function $\overline{\alpha}_V^{\kappa_s}$ such that, for each $\gamma \in (0,1)$, its value function $V_\gamma^{\kappa_s} \colon \mathcal{X} \to \mathbb{R}$ satisfies*

$$\frac{\overline{R}_\mathcal{A}}{1-\gamma} - V_\gamma^{\kappa_s}(x) = \sum_{k=0}^{\infty} \gamma^k \left(\overline{R}_\mathcal{A} - R(x_k^{\kappa_s}, \kappa_s(x_k^{\kappa_s}))\right) \leq \overline{\alpha}_V^{\kappa_s}(\omega(x)) \quad \forall x \in \mathcal{X}.$$

The assumptions can be interpreted as follows. Item (i) in Assumption 3.6 ensures that, relative to the maximum reward obtained in the target set, the reward is upper-bounded by a function that is negative definite with respect to the target set and decreases as the distance to the target set increases. For example, the negated Euclidean distance to a target set with a constant baseline offset satisfies this condition. Item (ii) in Assumption 3.6 ensures that the stabilization task is solvable by a continuous feedback law, as illustrated in the following example.

Example 1. *Consider the DT system $x^+ = x + u$ with compact sets $\mathcal{X} = \mathcal{U} = [-1,1]$. Let $\mathcal{A} = \{0\}$, $\omega(x) = |x|$, and $R(x,u) = -(x^2+u^2)$, which gives $\overline{R}_\mathcal{A} = \max_{u\in[-1,1]} R(0,u) = 0$. Suppose $\mathbb{K}$ contains the proportional continuous feedback law $\kappa_s(x) = -0.5x$. Applying κ_s yields the closed-loop DT dynamics $x_{k+1} = 0.5x_k$ and the solution $x_k^{\kappa_s} = 0.5^k x_0$ for each $k \in \mathbb{N}$ and each initial state $x_0 \in \mathcal{X}$. For each $k \in \mathbb{N}$, $-R(x_k^{\kappa_s}, \kappa_s(x_k^{\kappa_s})) = (x_k^{\kappa_s})^2 + (-0.5x_k^{\kappa_s})^2 = 1.25(x_k^{\kappa_s})^2$. Substituting in the solution $x_k^{\kappa_s}$ yields $-R(x_k^{\kappa_s}, \kappa_s(x_k^{\kappa_s})) = 1.25((0.5)^k x_0)^2 = 1.25(0.25)^k x_0^2$. Evaluating at $x_0 = x$, the shifted value function for κ_s is then*

$$\frac{\overline{R}_\mathcal{A}}{1-\gamma} - V_\gamma^{\kappa_s}(x) = \sum_{k=0}^{\infty} \gamma^k (1.25(0.25)^k x^2) = 1.25x^2 \sum_{k=0}^{\infty} (0.25\gamma)^k \leq 1.25x^2 \sum_{k=0}^{\infty} (0.25)^k = \frac{5}{3}x^2,$$

where we use the fact that $\gamma \in (0,1)$ *and properties of the geometric series. By defining the class-*$\mathcal{K}$ *function* $\overline{\alpha}_V^{\kappa_s}(s) := \frac{5}{3}s^2$, *we obtain* $\frac{\overline{R}_{\mathcal{A}}}{1-\gamma} - V_\gamma^{\kappa_s}(x) \leq \overline{\alpha}_V^{\kappa_s}(\omega(x))$ *for each* $x \in \mathcal{X}$ *and each* $\gamma \in (0,1)$.

Next, we make the following assumption on the existence of a pointwise optimal feedback law.

Assumption 3.7 (Ideal Expressivity)**.** *For each* $\gamma \in (0,1)$, *the admissible set of feedback laws* $\mathbb{K}$ *contains a pointwise optimal feedback law* $\kappa_\gamma^\circ \in \mathbb{K}$, *that is,*

$$V_\gamma^{\kappa_\gamma^\circ}(x) \geq V_\gamma^{\kappa}(x) \quad \forall (x,\kappa) \in \mathcal{X} \times \mathbb{K}.$$

To illustrate Assumption 3.7, we revisit Example 1.

Example 2. *Consider the system from Example 1. As the DT dynamics are linear and the cost quadratic, the pointwise optimal feedback law is of the form* $\kappa_\gamma^\circ(x) = -K_\gamma x$, *where* K_γ *is given by the discounted DT algebraic Riccati equation; see, for example, Section 3.2 in Bertsekas (2001). In particular,* κ_γ° *is continuous and* K_γ *is finite. Suppose* $K_\gamma < K$, *where* $K > 0$ *is finite. If we define* $\mathbb{K}$ *as the space of* K*-Lipschitz continuous functions, it naturally follows that* $\kappa_\gamma^\circ \in \mathbb{K}$.

Remark. *Assumption 3.7 implicitly assumes that Problem 1 can be pointwise optimally solved by a continuous feedback law. Consequently, this formulation excludes RL problems where the pointwise optimal feedback law is discontinuous, such as bang-bang control, and RL problems that contain topological obstructions, such as obstacle avoidance tasks (de Priester et al., 2022; 2024). While this is a limitation of our theoretical work, discontinuous feedback laws typically lack robustness to measurement noise and execution delays, which can induce high-frequency chattering that degrades physical actuators. In fact, smooth continuous feedback laws are often preferred over aggressive, discontinuous optimal control in applications.*

The RL objective in Problem 1 evaluates policies by aggregating their return over the state space via an expected value functional. While maximizing an expected value generally guarantees optimality only in an averaged sense, the following lemma establishes a strictly stronger result. It proves that, under Assumption 3.7, the feedback law that maximizes the RL objective is exactly pointwise optimal for every individual state. Furthermore, we leverage this pointwise optimality to derive a uniform upper bound on the value function.

Lemma 3.8 (Pointwise Optimality of the Optimal Feedback Law)**.** *Fix* $\gamma \in (0,1)$. *Suppose Assumption 3.1, item (ii) in Assumption 3.6, and Assumption 3.7 hold. Let* $\kappa_\gamma^* \in \mathbb{K}$ *be an optimal feedback law that solves Problem 1 and* $\kappa_\gamma^\circ \in \mathbb{K}$ *be a pointwise optimal feedback law from Assumption 3.7. Then, its value function* $V_\gamma^{\kappa_\gamma^*} : \mathcal{X} \to \mathbb{R}$ *is pointwise optimal for each* $x \in \mathcal{X}$, *that is,*

$$V_\gamma^{\kappa_\gamma^*}(x) = V_\gamma^{\kappa_\gamma^\circ}(x) \quad \forall x \in \mathcal{X}.$$

Consequently, there exists $\kappa_s \in \mathbb{K}$ *for which there exists a class-*$\mathcal{K}$ *function* $\overline{\alpha}_V^{\kappa_s}$ *such that, for each* $\gamma \in (0,1)$, *the value function* $V_\gamma^{\kappa_\gamma^*} : \mathcal{X} \to \mathbb{R}$ *satisfies*

$$\frac{\overline{R}_{\mathcal{A}}}{1-\gamma} - V_\gamma^{\kappa_\gamma^*}(x) \leq \overline{\alpha}_V^{\kappa_s}(\omega(x)) \quad \forall x \in \mathcal{X}, \tag{9}$$

where $\overline{R}_{\mathcal{A}} := \max_{(x,u)\in\mathcal{A}\times\mathcal{U}} R(x,u)$.

Building upon the uniform bound established in Lemma 3.8, we now present the main theoretical result of this section, which establishes that the DT closed-loop system in (2) subject to the optimal feedback law that solves Problem 1 renders the target set *uniformly practically asymptotically stable* (UPAS). This property guarantees that complete solutions to the closed-loop system approach and indefinitely remain within a bounded *neighborhood* around the target set.

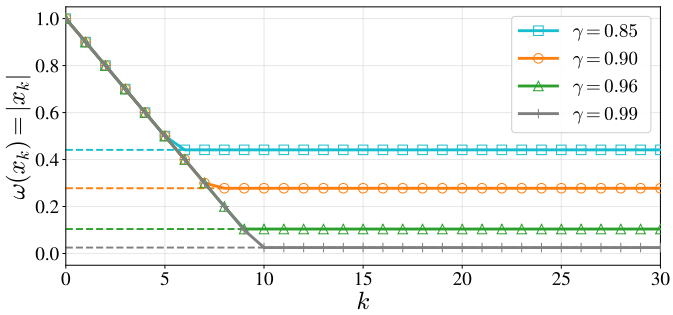


Figure 1: Solution trajectories of the closed-loop system defined in Example 3 generated by the optimal feedback law κ_γ^* for $\gamma \in \{0.85, 0.90, 0.96, 0.99\}$.

Theorem 3.9 (Uniform Practical Asymptotic Stability of the Optimal Feedback Law)**.** *Suppose Assumptions 3.1, 3.6, and 3.7 hold. Let $\omega\colon \mathcal{X} \to \mathbb{R}_{\geq 0}$ be a proper indicator of $\mathcal{A}$ on $\mathcal{X}$, and for each $\gamma \in (0,1)$, let $\kappa_\gamma^* \in \mathbb{K}$ denote an optimal feedback law that solves Problem 1, whose existence is guaranteed by Theorem 3.5. Then, there exist $\beta \in \mathcal{KL}$ and a continuous function $\nu\colon (0,1) \to \mathbb{R}_{\geq 0}$ such that, for each $\gamma \in (0,1)$ and each initial state $x_0^* \in \mathcal{X}$, the solution $\{x_k^*\}_{k\in\mathbb{N}}$ to the DT closed-loop system in (2) generated by κ_γ^* satisfies*

$$\omega(x_k^*) \leq \max\{\beta(\omega(x_0^*), k),\ \nu(\gamma)\} \quad \forall k \in \mathbb{N}, \tag{10}$$

where $\lim_{\gamma\to 1} \nu(\gamma) = 0$.

Importantly, the size of this neighborhood depends on the discount factor γ. In particular, a smaller discount factor corresponds to a myopic agent that can settle for a small steady-state error once it gets "close enough" to the target. As the agent becomes more farsighted, that is, as γ approaches 1, this neighborhood shrinks, and, in the limit, shrinks to the target set.

Example 3. *Consider the DT system $x^+ = x + 0.1u$ on the compact domains $\mathcal{X} = [-1,1]$ and $\mathcal{U} = [-1,1]$. Let the target set be $\mathcal{A} = \{0\}$ with proper indicator $\omega(x) = |x|$. To observe the ultimate bound $\nu(\gamma)$ established in Theorem 3.9, we define the reward function $R(x,u) = -(x^2 + 0.5|u|)$.*

Because the linear control penalty dominates the quadratic state penalty near the origin, the optimal feedback law exhibits a steady-state error. Inside this ultimate bounding region, the optimal action is $u = 0$, as the immediate cost of a corrective step outweighs the discounted benefit of achieving a smaller future state penalty. In particular, the boundary of this steady-state error can be analytically determined as

$$\nu(\gamma) = \frac{5(1-\gamma)}{2\gamma}.$$

The optimal feedback law is given by the continuous function $\kappa_\gamma^*(x) = \text{clip}\Big(-10\max(x-\nu(\gamma),0) + 10\max(-x-\nu(\gamma),0),\ -1,\ 1\Big)$, where $\text{clip}(s,a,b) := \max\{a, \min\{s,b\}\}$ for each $(s,a,b) \in \mathbb{R}^3$. Figure 1 shows that the solution trajectories generated by κ_γ^* to the closed-loop system converge to their analytical boundaries $\nu(\gamma)$ for each $\gamma \in \{0.85, 0.90, 0.96, 0.99\}$. Furthermore, as γ increases, the steady-state error decreases.

As established in Theorem 3.9, the ultimate bound in (10) vanishes as the discount factor γ approaches one. However, in practice, pushing $\gamma \to 1$ often induces training instability. Fortunately, a direct and practical consequence of Theorem 3.9 is that, for any desired level of closeness, there exists a discount factor $\gamma < 1$ that is sufficiently large to achieve this tolerance without requiring the strict limit $\gamma \to 1$. For example, if we want to train a robotic manipulator to reach and stay within a particular distance to the target, we can always find a discount factor whose optimal feedback law guarantees this specification. The following corollary formalizes this property.

Corollary 3.9.1. *Under the conditions of Theorem 3.9, for each $\rho > 0$, there exists $\gamma^* \in (0,1)$ such that, for each $\gamma \in [\gamma^*, 1)$ and each initial state $x_0^* \in \mathcal{X}$, the solution $\{x_k^*\}_{k\in\mathbb{N}}$ to the DT closed-loop system in (2) generated by κ_γ^* satisfies*

$$\omega(x_k^*) \leq \max\{\beta(\omega(x_0^*), k),\ \rho\} \quad \forall k \in \mathbb{N}. \tag{11}$$

4 Approximate Solution

Searching the infinite-dimensional space $\mathbb{K}$ to exactly solve Problem 1 is computationally intractable. Instead, RL algorithms typically restrict the search to a parameterized class of continuous feedback laws, specifically feedforward neural networks (Kidger & Lyons, 2020, Definition 3.1).

Definition 4.1 (Feedforward Neural Networks)**.** *Let $\sigma\colon \mathbb{R} \to \mathbb{R}$ and $n, m, w \in \mathbb{N}_{>0}$. Let $\mathcal{NN}^\sigma_{n,m,w}$ represent the class of functions from $\mathbb{R}^n$ to $\mathbb{R}^m$ described by feedforward neural networks with n neurons in the input layer, m neurons in the output layer, and an arbitrary number of hidden layers, each with w neurons using the activation function σ. Every neuron in the output layer has the identity activation function.*

The following theorem and lemma guarantee that this feedforward neural network architecture is capable of uniformly approximating the value function of any continuous feedback law with arbitrary precision.

Theorem 4.1 (Universal Approximation (Kidger & Lyons, 2020, Theorem 3.2))**.** *Let $\sigma\colon \mathbb{R} \to \mathbb{R}$ be any nonaffine continuous function that is continuously differentiable at at least one point, with a nonzero derivative at that point.*[4] *Let $\mathcal{X} \subset \mathbb{R}^n$ be compact. Then, $\mathcal{NN}^\sigma_{n,m,n+m+2}$ is dense in $\mathcal{C}(\mathcal{X}, \mathbb{R}^m)$ with respect to the supremum norm.*

Lemma 4.2 (Uniform Value Approximation)**.** *Fix $\gamma \in (0,1)$. Suppose Assumption 3.1 holds. Let $n, m \in \mathbb{N}_{>0}$ be the given dimensions of the state space $\mathcal{X}$ and action space $\mathcal{U}$, respectively, and let σ be an activation function satisfying the conditions of Theorem 4.1. Then, for each continuous feedback law $\kappa \in \mathbb{K}$ and tolerance $\delta > 0$, there exist $\ell \in \mathbb{N}_{>0}$ and a parameter vector $\bar{\psi} \in \mathbb{R}^\ell$ such that the parameterized network $\kappa_{\bar{\psi}} \in \mathcal{NN}^\sigma_{n,m,n+m+2} \cap \mathbb{K}$ satisfies*

$$\max_{x\in\mathcal{X}} \left| V_\gamma^\kappa(x) - V_\gamma^{\kappa_{\bar{\psi}}}(x) \right| \leq \delta. \tag{12}$$

While Lemma 4.2 guarantees the existence of a near-optimal network, searching over all possible neural network architectures and parameters is computationally intractable. Instead, RL algorithms typically fix a specific architecture and optimize its finite-dimensional parameter vector, namely, weights and biases, over a bounded domain. We formalize this approximate RL problem as follows.

Problem 2 (Approximate Reinforcement Learning Problem)**.** *Given a discount factor $\gamma \in (0,1)$, state and action space dimensions $n, m \in \mathbb{N}_{>0}$, a feedforward neural network architecture $\mathcal{NN}^\sigma_{n,m,w}$ with given activation function σ and hidden layer width $w \in \mathbb{N}_{>0}$, and a compact parameter set $\Psi \subset \mathbb{R}^\ell$ for some parameter dimension $\ell \in \mathbb{N}_{>0}$, the approximate RL problem is defined as*

$$\psi^* \in \arg\max_{\psi\in\Psi} \mathcal{J}_\gamma(\kappa_\psi) \quad \text{s.t.} \quad \kappa_\psi \in \mathcal{NN}^\sigma_{n,m,w} \cap \mathbb{K}. \tag{13}$$

Remark. *Constraining the search to a compact parameter set Ψ ensures that the maximizers in Problem 2 are well-defined. Furthermore, in RL implementations, network parameters are typically bounded by regularization techniques, such as weight decay and gradient clipping.*

The following lemma guarantees that solving the approximate RL problem preserves near-optimal performance, provided the network architecture is chosen sufficiently large.

[4] This condition is satisfied by almost all common activation functions. It includes smooth functions like $\tanh$ and sigmoid, as well as ReLU (since its derivative is strictly positive for all $x > 0$).

Lemma 4.3 (Near-Optimality of Approximate RL)**.** *Fix $\gamma \in (0,1)$. Suppose Assumption 3.1 holds. Let $n, m \in \mathbb{N}_{>0}$ be the given dimensions of the state space $\mathcal{X}$ and action space $\mathcal{U}$, respectively, and let σ be an activation function satisfying the conditions of Theorem 4.1. Then, for each $\kappa_\gamma^* \in \mathbb{K}$ that solves Problem 1 and each $\delta > 0$, there exist a parameter dimension $\ell \in \mathbb{N}_{>0}$ and a compact set $\Psi \subset \mathbb{R}^\ell$ such that an optimal parameterized feedback law $\kappa_{\psi^*} \in \mathcal{NN}^\sigma_{n,m,n+m+2} \cap \mathbb{K}$ that solves Problem 2 using Ψ satisfies*

$$\mathcal{J}_\gamma(\kappa_{\psi^*}) \in [\mathcal{J}_\gamma(\kappa_\gamma^*) - \delta, \mathcal{J}_\gamma(\kappa_\gamma^*)]. \tag{14}$$

As formalized in Problems 1 and 2, RL algorithms maximize the expected return $\mathcal{J}_\gamma$, which evaluates the value functional integrated over the state space $\mathcal{X}$ for a given admissible state distribution. However, establishing UPAS requires a worst-case bound on the value function that holds pointwise across all individual states. By leveraging the compactness of the admissible set $\mathbb{K}$, the following lemma bridges this gap: it establishes that bounding the expected suboptimality of a feedback law, which is the metric optimized in Problem 2, guarantees a worst-case bound on its pointwise error over $\mathcal{X}$.

Lemma 4.4 (Pointwise Error Bound)**.** *Fix $\gamma \in (0,1)$. Suppose Assumption 3.1, item (ii) in Assumption 3.6, and Assumption 3.7 hold. Let $\kappa_\gamma^* \in \mathbb{K}$ denote an optimal feedback law that solves Problem 1, whose existence is guaranteed by Theorem 3.5. Then, there exists $\alpha_\gamma \in \mathcal{K}$ such that*

$$\max_{x \in \mathcal{X}} \left\{ V_\gamma^{\kappa_\gamma^*}(x) - V_\gamma^{\kappa}(x) \right\} \leq \alpha_\gamma(\mathcal{J}_\gamma(\kappa_\gamma^*) - \mathcal{J}_\gamma(\kappa)) \quad \forall \kappa \in \mathbb{K}. \tag{15}$$

We now extend the stability guarantees in Theorem 3.9 to the approximate RL setting of Problem 2. While computing the exact optimal policy is generally intractable, solving Problem 2 yields a parameterized feedback law with an approximation error δ. Our main theoretical result establishes that this parameterized feedback law still guarantees closed-loop UPAS. The ultimate steady-state error is now governed by the discount factor γ *and* the approximation error δ.

Theorem 4.5 (Uniform Practical Asymptotic Stability of the Optimal Parameterized Feedback Law)**.** *Suppose Assumptions 3.1, 3.6, and 3.7 hold. Let $n, m \in \mathbb{N}_{>0}$ be the given dimensions of the state space $\mathcal{X}$ and action space $\mathcal{U}$, respectively, and let σ be an activation function satisfying the conditions of Theorem 4.1. For each $\gamma \in (0,1)$, let $\kappa_\gamma^* \in \mathbb{K}$ denote an optimal feedback law that solves Problem 1, whose existence is guaranteed by Theorem 3.5. For each $\delta > 0$, let $\kappa_{\psi^*} \in \mathcal{NN}^\sigma_{n,m,n+m+2} \cap \mathbb{K}$ denote an optimal parameterized feedback law, whose existence within a compact set Ψ for width $w = n + m + 2$ is guaranteed by Lemma 4.3, that solves Problem 2. Then, there exist $\beta \in \mathcal{KL}$ and a function $\tilde{\nu} \colon (0,1) \times \mathbb{R}_{>0} \to \mathbb{R}_{\geq 0}$ such that, for each $\gamma \in (0,1)$, $\delta > 0$, and $x_0 \in \mathcal{X}$, the solution $\{x_k\}_{k \in \mathbb{N}}$ to $x^+ = g(x, \kappa_{\psi^*}(x))$ satisfies*

$$\omega(x_k) \leq \max\{\beta(\omega(x_0), k),\ \tilde{\nu}(\gamma, \delta)\} \quad \forall k \in \mathbb{N}, \tag{16}$$

where $\lim_{\gamma \to 1} (\lim_{\delta \to 0} \tilde{\nu}(\gamma, \delta)) = 0$.

Example 4. *Consider the setup from Example 3 with a fixed discount factor $\gamma = 0.90$. To observe the ultimate bound $\tilde{\nu}(\gamma, \delta)$ established in Theorem 4.5, we evaluate parameterized feedback laws κ_{ψ^*} subject to varying approximation errors $\delta > 0$. Testing these varying δ values can be thought of as simulating the RL training process, where the approximation error δ is progressively reduced as the network parameters are optimized.*

As established in Theorem 4.5, the approximation error δ acts as a bounded disturbance to the closed-loop system, strictly enlarging the steady-state error as δ increases. We model the parameterized policy $\kappa_{\psi^*} \in \mathcal{NN}^\sigma_{n,m,w}$ as a piecewise-linear ReLU network given by

$$\kappa_{\psi^*}(x) = \text{clip}\Big(-10\max(w_\delta x - \nu(0.90), 0) + 10\max(-w_\delta x - \nu(0.90), 0),\ -1,\ 1\Big).$$

Here, the approximation error δ is induced by a degraded network weight $w_\delta \in (0,1]$. This weight artificially dilates the agent's perceived distance to the target, thereby expanding the steady-state error boundary from the analytical bound $\nu(0.90)$ to $\tilde{\nu}(\gamma, \delta) := \nu(0.90)/w_\delta$.

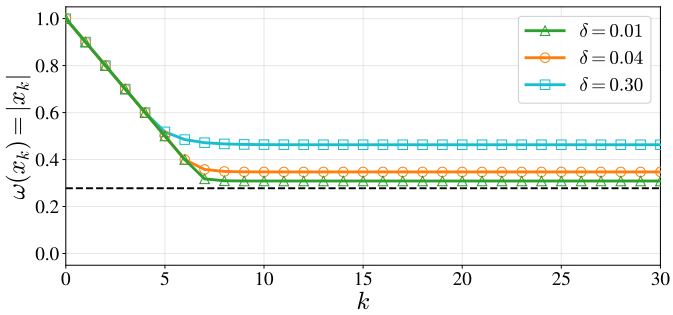


Figure 2: Solution trajectories of the closed-loop system defined in Example 4 generated by the parameterized feedback laws κ_{ψ^*} for $\gamma = 0.90$ and $\delta \in \{0.01, 0.04, 0.30\}$.

By numerically evaluating the value functions over the state space $\mathcal{X}$, we map the improving weights $w_\delta \in \{0.6, 0.8, 0.9\}$ to their maximum pointwise approximation errors $\delta = \max_{x \in \mathcal{X}} |V_\gamma^{\kappa_\gamma^*}(x) - V_\gamma^{\kappa_{\psi^*}}(x)|$, yielding a sequence of shrinking errors $\delta \in \{0.30, 0.04, 0.01\}$, respectively. As shown in Figure 2, the closed-loop trajectories converge to their boundaries $\tilde{\nu}(\gamma, \delta)$. Consistent with the practical stability bound in (16), larger approximation errors δ yield proportionally larger steady-state errors beyond the analytical boundary $\nu(\gamma)$.

5 Conclusion

In this paper, we showed that RL intrinsically guarantees a practical notion of closed-loop stability for continuous control tasks under mild assumptions without requiring algorithmic modifications. By adapting the discrete-time input-to-state stability (ISS) framework, we modeled the suboptimality induced by the discount factor and the neural network approximation error as bounded disturbances. We proved that near-optimal parameterized policies inherently steer the closed-loop system into a bounded neighborhood of the target set. The size of this neighborhood vanishes in the limit as the approximation error approaches zero and, subsequently, the discount factor approaches one. Future work includes accommodating discontinuous optimal feedback laws to handle tasks with topological obstructions, such as obstacle avoidance. Additionally, while our current analysis is restricted to deterministic systems, we believe there are no fundamental hurdles to extending these proofs to stochastic dynamics; by incorporating probabilistic transitions into the framework, we hypothesize that similar ultimate bounds can be derived that hold in expectation.

Acknowledgments

This research has been partially supported by NSF Grant no. CNS-2111688, by AFOSR Grants nos. FA9550-23-1-0313, FA9550-23-1-0678, and FA9550-26-1-B339, by AFRL Grant nos. FA8651-22-1-0017 and FA8651-23-1-0004, by ARO Grant no. W911NF261A048, by DoD Grant no. W911NF-23-1-0158, and by UC Alianza MX Grant no. SRPUCMX24-01.

References

Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Jozefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, et al. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

Felix Berkenkamp, Matteo Turchetta, Angela Schoellig, and Andreas Krause. Safe model-based reinforcement learning with stability guarantees. *Advances in neural information processing systems*, 30, 2017.

Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*, volume 2. Athena Scientific, Belmont, MA, 2nd edition, 2001.

Ya-Chien Chang, Nima Roohi, and Sicun Gao. Neural lyapunov control. *Advances in neural information processing systems*, 32, 2019.

Hongkai Dai, Benoit Landry, Lujie Yang, Marco Pavone, and Russ Tedrake. Lyapunov-stable neural-network control. *arXiv preprint arXiv:2109.14152*, 2021.

Jan de Priester, Ricardo G. Sanfelice, and Nathan van de Wouw. Hysteresis-based rl: Robustifying reinforcement learning-based control policies via hybrid control. In *2022 American Control Conference (ACC)*, pp. 2663–2668, 2022. DOI: 10.23919/ACC53348.2022.9867627.

Jan de Priester, Zachary Bell, Prashant Ganesh, and Ricardo Sanfelice. MultiHyRL: Robust hybrid RL for obstacle avoidance against adversarial attacks on the observation space. *Reinforcement Learning Journal*, 4:2017–2040, 2024.

Rafal Goebel, Ricardo G. Sanfelice, and Andrew R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, Princeton, NJ, 2012.

Gene Grimm, Michael J Messina, Sezai Emre Tuna, and Andrew R Teel. Model predictive control: for want of a local control lyapunov function, all is not lost. *IEEE Transactions on Automatic Control*, 50(5):546–558, 2005.

Minghao Han, Lixian Zhang, Jun Wang, and Wei Pan. Actor-critic reinforcement learning for control with stability guarantee. *IEEE Robotics and Automation Letters*, 5(4):6217–6224, 2020.

Minghao Han, Yuan Tian, Lixian Zhang, Jun Wang, and Wei Pan. Reinforcement learning control of constrained dynamic systems with uniformly ultimate boundedness stability guarantee. *Automatica*, 129:109689, 2021.

Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science robotics*, 4(26):eaau5872, 2019.

Zhong-Ping Jiang and Yuan Wang. Input-to-state stability for discrete-time nonlinear systems. *Automatica*, 37(6):857–869, 2001.

Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620(7976):982–987, 2023.

Patrick Kidger and Terry Lyons. Universal Approximation with Deep Narrow Networks. In Jacob Abernethy and Shivani Agarwal (eds.), *Proceedings of Thirty Third Conference on Learning Theory*, volume 125 of *Proceedings of Machine Learning Research*, pp. 2306–2327. PMLR, 09–12 Jul 2020. URL https://proceedings.mlr.press/v125/kidger20a.html.

Pavel Osinenko, Lukas Beckenbach, Thomas Göhrt, and Stefan Streif. A reinforcement learning method with closed-loop stability guarantee. *IFAC-PapersOnLine*, 53(2):8043–8048, 2020.

Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *ACM Transactions On Graphics (TOG)*, 37(4):1–14, 2018.

Romain Postoyan, Lucian Buşoniu, Dragan Nešić, and Jamal Daafouz. Stability analysis of discrete-time infinite-horizon optimal control with discounted cost. *IEEE Transactions on Automatic Control*, 62(6):2736–2749, 2016.

Spencer M Richards, Felix Berkenkamp, and Andreas Krause. The lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems. In *Conference on robot learning*, pp. 466–476. PMLR, 2018.

Walter Rudin. *Principles of Mathematical Analysis*. McGraw-Hill, New York, 3rd edition, 1976.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.