Title
Robustness in Nonlinear and Learning Based Control

Author
Yin, He

Publication Date
2021

Peer reviewed|Thesis/dissertation

Robustness in Nonlinear and Learning Based Control

by

He Yin

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Mechanical Engineering

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Murat Arcak, Chair
Professor Kameshwar Poolla
Professor Javad Lavaei

Spring 2021

Robustness in Nonlinear and Learning Based Control

Abstract

Robustness in Nonlinear and Learning Based Control

by

He Yin

Doctor of Philosophy in Engineering - Mechanical Engineering

University of California, Berkeley

Professor Murat Arcak, Chair

In this dissertation we propose a control synthesis and analysis framework for nonlinear, and neural network (NN) controlled systems with robustness guarantees. We quantify systems' robustness against external disturbances and perturbations using the following measures: (i) the forward reachable set; (ii) the backward reachable set; (iii) the tracking error bound; (iv) the region of attraction. These measures are all characterized by sublevel sets of storage functions satisfying appropriate dissipation inequalities that account for external disturbances and perturbations. Integral quadratic constraints (IQCs) are used to describe perturbations, allowing for a variety of perturbations including parametric uncertainty, unmodeled dynamics, and nonlinear activation functions in NNs. We formulate sum-of-squares (SOS) constraints and Linear matrix inequality conditions by merging dissipation inequalities with IQCs to compute controllers and their associated robustness measures.

We start off by focusing on the finite time horizon robustness of uncertain nonlinear (polynomial) systems, which are modeled as interconnections of nominal polynomial systems and perturbations. We propose a method of outer-approximating the forward reachable set on finite horizons for the uncertain nonlinear systems with controllers given. Then we move from analysis to control synthesis, and present a method for synthesizing a polynomial control law that steers the system to the target set with the goal of maximizing inner-approximations to the backward reachable set. The approximations to both the forward reachable set and backward reachable set are characterized by time dependent polynomial storage functions, and are computed using SOS programming. IQCs with both hard and soft factorizations are used to describe perturbations.

Furthermore, we address robust trajectory planning and control design for nonlinear systems. A hierarchical trajectory planning and control framework is proposed, where a low-fidelity model is used to plan trajectories satisfying planning constraints, and a high-fidelity model is used for synthesizing tracking controllers guaranteeing the boundedness of the error state between the low- and high-fidelity models. We formulate SOS optimizations for computing the tracking controllers and their associated tracking error bound, with the

goal of minimizing the volume of the tracking error bound. The tracking error bound is then used to redesign the planning constraints to guarantee safety of the system.

Finally, we move to NN controlled systems. We propose two theorems to prove local stability of NN controlled linear time invariant systems, and to compute inner-approximations to the region of attraction. The first theorem merges dissipation inequalities with local sector quadratic constraints (QCs) to bound the nonlinear activation functions in the NN. The second theorem includes IQCs to allow for perturbations, and further refine the description of activation functions by capturing their slope information. Then we move from analysis to control synthesis. Loop transformation is used to derive stability and safety conditions that are jointly convex in the Lyapunov function, weights of the NN controller, and the Lagrange multipliers for including QCs. These convex conditions are incorporated in the imitation learning process, which trades off between imitation learning accuracy, and size of the region of attraction inner-approximations, to learn robust NN controllers. We propose an alternating direction method of multipliers based algorithm to solve the constrained imitation learning problem.

To my fiancée Le Ma

and

to my parents Xiang Yin and Li Zhang

Contents

List of Figures

List of Tables

Acknowledgments

I would like to take this opportunity to thank the following people.

My greatest gratitude goes to my advisors, Andy Packard and Murat Arcak, and my collaborator Pete Seiler. I first knew Andy when I was taking ME 132 (fundamental control class) and ME 190L (loopshaping class) with him in Fall 2015 during my undergraduate study. It was very impressive to see such a well-known professor being so passionate about teaching, and patient with students. After joining his research group, it didn't take long for me to realize that he is incredibly smart, and at the same time, unbelievably modest. He is a role model for me, now and always. Murat has always been a great advisor, and a close friend. He gave me a lot of help and support especially when I was transitioning from the class taking mode to the research mode, and he has provided me many interesting ideas and opportunities. Thanks to him for providing me with a seat in an office with a coffee machine and a big window. Pete Seiler has an exceptional mind, and he is always willing to help. Thanks to him for long and valuable discussions on robust control and his support and guidance.

I thank Kameshwar Poolla for being a great leader and supporter for the members of the BCCI lab. I would like to thank him and Javad Lavaei for serving on my dissertation and qualifying exam committees. Thanks to Johnathan Shewchuk, and Koushil Sreenath for serving on my qualifying exam committee. I would like to thank my coauthors including Pierre-Jean Meyer, Stanley Smith, Ming jin, Monimoy Bujarbaruah, Jens E Bremnes, Yash Vardhan Pant, and Alex Devonport.

I would also like to acknowledge organizations that provided financial support throughout my graduate education. Specifically, the Air Force Office of Scientific Research under grant FA9550-18-1-0253, the Office of Naval Research under grant N00014-18-1-2209, the U.S. National Science Foundation under grant ECCS-1906164, and the Fanuc Corportation through the Fanuc Chair in Mechanical Engineering at UC Berkeley.

I am grateful with my friends from the BCCI lab. Octavio Narvaez, Emmanuel Sin, Akhil Shetty, Sen Li, Junjie Qin, Jared Porter, Wei Chen, Chris Meissen, Maria Vrakopoulou, Pratyush Chakraborty, Kate Schweidel, Hamidreza Tavafoghi, Arun Hegde, and John Mather. We shared sadness and happiness with each other. We were together through Karaoke nights, Lake Tahoe trip, Bodage Bay trip, Pine Crest trip, Oakland night, Jupiter time, night shifts, and Daimo early morning. Thanks to my friends from the DOP center, Alex Devonport, Stan Smith, Pierre-Jean Meyer, Eric Kim, Mikhael Burov, and Mindy Perkins, for the interesting chats and lunch outings.

Finally I want to thank Xiang Yin, Li Zhang, and Le Ma for their love and support.

Chapter 1

Introduction

Robust control has shown its effectiveness in controlling various types of linear systems, including time invariant [1], time varying [2, 3], and parameter varying systems [4]. However, modern control design and analysis methods for more complex systems, like nonlinear systems and neural network (NN) controlled systems, are still suffering from lack of robustness guarantees, which are important in safety-critical applications. A shortcoming of the existing nonlinear control and analysis methods is that they rely on accurate system models. Only limited forms of uncertainty have been addressed, such as parametric uncertainty [5–10]. As for NN controlled systems, they have long been suffering from lack of robustness certificates. Only a few results have attempted to assess their stability and safety, including a work on reachability analysis [11], and a result regarding global stability analysis [12]. The complexity of the NN structure, e.g., various types of nonlinear activation functions, potentially numerous layers, and a large number of hidden neurons, makes it difficult to apply classical analysis methods, e.g. Lyapunov theory.

In this dissertation, we develop analysis and control synthesis tools for uncertain nonlinear, and NN controlled systems with robustness guarantees using a unified framework based on integral quadratic constraints (IQCs) [13] from robust control theory. We quantify robustness of nonlinear and NN controlled systems using metrics including region of attractions, forward/backward reachable sets, and tracking error bounds. In each case, we account for external disturbances, and a general class of perturbations beyond parametric uncertainty.

In the framework, we model both uncertain nonlinear, and NN controlled systems as an interconnection (as shown in Fig. 1.1) of a nominal dynamical system and perturbations, whose input-output properties are characterized by IQCs, with both soft and hard factorizations. The use of IQCs allows for various types of uncertainties and nonlinearities, including unmodeled dynamics, and the nonlinear activation functions in the NNs. The robustness metrics are all characterized by sublevel sets of storage functions that satisfy appropriate dissipation inequalities merged with IQCs. The (generalized) S-procedure is used to formulate these dissipation inequalities as linear matrix inequalities (LMIs) and sum-of-squares (SOS) constraints for computing robustness metrics, and synthesizing controllers.

The content and contributions of each chapter are outlined below.

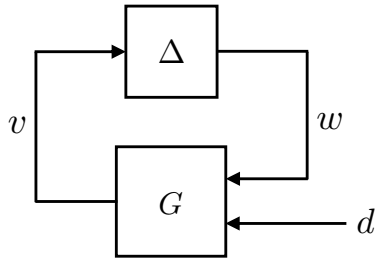

Figure 1.1: Interconnection $F_u(G, \Delta)$ of a nominal system G and a perturbation Δ

Chapter 2 presents the notation and required background materials from the controls literature. First, the system level properties (dissipativity) are introduced. Several types of dissipativity (e.g., stability, finite $\mathcal{L}_2$ gain, $\mathcal{L}_2$ reachability) and their corresponding supply rates are discussed. Then we describe IQCs, a special class of dynamic supply rate, that will be used to characterize perturbations. Finally, we briefly discuss how SOS problems can be translated to semidefinite problems (SDPs), and the computation complexity of the resulting SDPs.

Chapter 3 proposes a method to outer-approximate forward reachable sets on finite horizons for uncertain nonlinear systems with polynomial dynamics. This method makes use of time-dependent polynomial storage functions that satisfy appropriate dissipation inequalities to characterize the outer-approximations. The dissipation inequalities, combined with the SOS technique allows us to simultaneously accommodate multiple sources of uncertainty, including time-varying uncertain parameters, $\mathcal{L}_2$ disturbances, and perturbations Δ characterized by IQCs. The use of IQCs allows for various types of uncertainty, including unmodeled dynamics.

The proposed analysis framework considers both hard and soft IQC factorizations. Dissipation inequalities usually require IQCs to hold over all finite horizons (hard IQCs). However, IQCs are often available in the infinite-time horizon (soft IQCs), while the hard IQCs are not. To mitigate this issue, we incorporate soft IQCs in dissipation inequalities by making use of a lower bound derived from [14], which is valid for soft IQCs over all finite horizons.

Chapter 4 extends the results from Chapter 3 to control synthesis. A method is proposed to compute robust inner-approximations to the backward reachable set of uncertain nonlinear systems, and to generate a robust control law that drives trajectories starting in these inner-approximations to a target set. The method also incorporates both hard and soft IQC factorizations. Moreover, we overcome a technical challenge that arises when the input of the perturbation depends directly on the control command, as in the case of actuator uncertainty. This dependence creates a source of nonconvexity, which we circumvent by introducing auxiliary states in the control law.

Chapter 5 proposes a hierarchical trajectory planning and control framework for nonlinear systems. In this framework, a low-fidelity model is used to generate planning trajectories satisfying planning constraints, and a high-fidelity model is used to design tracking controllers to track planned trajectories with a bounded tracking error. An SOS optimization

is proposed to compute the polynomial tracking controller and its associated tracking error bound. The tracking error bound is then used to redesign the planning constraints to guarantee safety of the system.

Chapter 6 presents a method to analyze the stability of feedback systems with NN controllers. Two stability theorems are given to prove asymptotic stability and to compute an ellipsoidal inner-approximation to the ROA. The first theorem addresses linear time-invariant systems, and merges Lyapunov theory with local (sector) quadratic constraints to bound the nonlinear activation functions in the NNs. The second theorem allows the system to include perturbations using IQCs to capture their input/output behavior. This in turn allows for off-by-one IQCs to refine the description of activation functions by capturing their slope restrictions. Both results rely on semidefinite programming to approximate the ROA.

Chapter 7 presents a method to learn NN controllers with stability and safety guarantees through imitation learning. The stability condition from Chapter 6 is nonconvex in the Lyapunov function, and the weights of NN controllers, and thus computationally intractable for NN control synthesis; here we convexify this constraint (using loop transformation) for its efficient enforcement in the learning process. A cost function is proposed for the learning process, which encourages small imitation learning loss, and a large-volume ROA, simultaneously. An alternating direction method of multipliers based algorithm is proposed to solve the constrained imitation learning problem. Notably, a well-known challenge in imitation learning is the existence of suboptimal demonstrations. As demonstrated in the case studies, while the proposed approach can train a policy that imitates the expert demonstrations, it can potentially improve local stability over suboptimal expert policies, thus enhance the robustness of imitation learning.

Chapter 2

Background

2.1 Notation

$\mathbb{R}^{m\times n}$ and $\mathbb{S}^n$ denote the set of m-by-n real matrices and n-by-n real, symmetric matrices. $\mathbb{S}^n_+$ and $\mathbb{S}^n_{++}$ denote the sets of n-by-n symmetric, positive semidefinite and positive definite matrices, respectively. $\mathbb{RL}_\infty$ is the set of rational functions with real coefficients that have no poles on the imaginary axis. $\mathbb{RH}_\infty \subset \mathbb{RL}_\infty$ contains functions that are analytic in the closed right-half of the complex plane.

For a vector $w \in \mathbb{R}^n$, the Euclidean norm is denoted as $|w|$. $\mathcal{L}_2^{n_r}$ is the space of measurable functions $r : [0,\infty) \to \mathbb{R}^{n_r}$, with $\|r\|_{\mathcal{L}_2} := (\int_0^\infty |r(t)|^2 dt)^{1/2} < \infty$. Associated with $\mathcal{L}_2^{n_r}$ is the extended space $\mathcal{L}_{2e}^{n_r}$, consisting of functions whose truncation $r_T(t) := r(t)$ for $t \le T$; $r_T(t) := 0$ for $t > T$, is in $\mathcal{L}_2^{n_r}$ for all $T > 0$. Define the finite-horizon $\mathcal{L}_2$ norm as $\|r\|_{\mathcal{L}_2,[t_0,T]} := \left(\int_{t_0}^T |r(t)|^2 dt\right)^{1/2}$. If r is measurable, and $\|r\|_{\mathcal{L}_2,[t_0,T]} < \infty$ then $r \in \mathcal{L}_2^{n_r}[t_0,T]$. The finite horizon induced $\mathcal{L}_2$ to $\mathcal{L}_2$ norm is denoted as $\|\cdot\|_{\mathcal{L}_2\to\mathcal{L}_2,[t_0,T]}$. $\ell_2^{n_x}$ is the set of sequences $x : \mathbb{N} \to \mathbb{R}^{n_x}$ with $\|x\|_{\ell_2} := (\sum_{k=0}^\infty |x(k)|^2)^{1/2} < \infty$. The extended space $\ell_{2e}^{n_x}$ consists of sequences whose truncation $x_N(k) = x(k)$ for $k \le N$; $x_N(k) = 0$ for $k > N$, is in $\ell_2^{n_x}$ for all $N \ge 0$. When applied to vectors, the orders $>, \le$ are applied elementwise.

For $\xi \in \mathbb{R}^n$, $\mathbb{R}[\xi]$ represents the set of polynomials in ξ with real coefficients, and $\mathbb{R}^m[\xi]$ and $\mathbb{R}^{m\times p}[\xi]$ to denote all vector and matrix valued polynomial functions. The subset $\Sigma[\xi] := \{\pi = \pi_1^2 + \pi_2^2 + ... + \pi_M^2 : \pi_1, ..., \pi_M \in \mathbb{R}[\xi]\}$ of $\mathbb{R}[\xi]$ is the set of SOS polynomials in ξ. For $\eta \in \mathbb{R}$, and continuous $r : \mathbb{R}^n \to \mathbb{R}$, define the sublevel set:

$$\Omega_\eta^r := \{x \in \mathbb{R}^n : r(x) \le \eta\}. \tag{2.1}$$

For $\eta \in \mathbb{R}$, and continuous $g : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$, define the t-dependent sublevel set:

$$\Omega_{t,\eta}^g := \{x \in \mathbb{R}^n : g(t,x) \le \eta\}. \tag{2.2}$$

For $P \in \mathbb{S}^n_{++}$, $x_* \in \mathbb{R}^n$, define the ellipsoid

$$\mathcal{E}(P, x_*) := \{x \in \mathbb{R}^n : (x - x_*)^\top P(x - x_*) \le 1\}. \tag{2.3}$$

If $x_* = 0$, the ellipsoid centered around the origin is defined as

$$\mathcal{E}(P) := \{x \in \mathbb{R}^n : x^\top P x \leq 1\}. \tag{2.4}$$

KYP denotes a mapping to the block 2-by-2 matrix:

$$KYP(Y, A, B, C, D, M) := \begin{bmatrix} A^\top Y + YA & YB \\ B^\top Y & 0 \end{bmatrix} + \begin{bmatrix} C^\top \\ D^\top \end{bmatrix} M \begin{bmatrix} C & D \end{bmatrix}. \tag{2.5}$$

2.2 Dissipativity

The notion of dissipativity, introduced in [15, 16], describes how the inputs and outputs of systems correlate. This correlation is characterized by the chosen scalar-valued supply rate. Different choice of supply rate determines different types of dissipativity. Consider the following dynamical system

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)), \quad f(0,0) = 0 \\ y(t) &= h(x(t), u(t)), \quad h(0,0) = 0 \end{aligned} \tag{2.6}$$

with $x(t) \in \mathbb{R}^{n_x}$, $u(t) \in \mathbb{R}^{n_u}$, $y(t) \in \mathbb{R}^{n_y}$, and continuously differentiable mappings $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_x}$ and $h : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_y}$.

Definition 1. *The system* (2.6) *is dissipative with respect to a supply rate* $s(u, y)$ *if there exists* $V : \mathbb{R}^{n_x} \to \mathbb{R}$ *such that* $V(0) = 0$, $V(x) \geq 0$ *for all* $x \in \mathbb{R}^{n_x}$, *and*

$$V(x(\tau)) - V(x(0)) \leq \int_0^\tau s(u(t), y(t))dt \tag{2.7}$$

for every input signal $u(\cdot)$ *and every* $\tau \geq 0$ *in the interval of existence of the solution* $x(t)$. $V(\cdot)$ *is called a storage function, and Equation* (2.7) *is referred to as the dissipation inequality.*

Important types of dissipativity and their corresponding supply rates are described below.

- Stability: $s(0,0) = 0$ and $s(0, y) \leq 0$ for all $y \in \mathbb{R}^{n_y}$

  If V is positive definite, then this supply rate implies that the origin is Lyapunov stable.

- Finite $\mathcal{L}_2$ gain: $s(u, y) = \gamma^2 |u|^2 - |y|^2$ with $\gamma > 0$

  Substituting this supply rate into (2.7), and using a zero initial condition $x(0) = 0$ yields

  $$V(x(\tau)) \leq \gamma^2 \int_0^\tau |u(t)|^2 dt - \int_0^\tau |y(t)|^2 dt \quad \text{for all } u, \text{ and } \tau \geq 0. \tag{2.8}$$

  Apply $V(x) \geq 0$ to show that the $\mathcal{L}_2$ gain of the system is γ.

- $\mathcal{L}_2$ Reachability: $s(u,y) = |u|^2$

  Substituting this supply rate into (2.7), and using a zero initial condition $x(0) = 0$ yields

  $$V(x(\tau)) \leq \int_0^\tau |u(t)|^2 dt \quad \text{for all } u, \text{ and } \tau \geq 0. \tag{2.9}$$

  If the energy of the input is bounded: $\|u\|_{\mathcal{L}_2}^2 \leq \gamma$, then all the trajectories starting from $x(0) = 0$ are bounded by the sublevel set Ω_γ^V, defined in (2.1).

- Passivity: $s(u,y) = u^\top y$

  Using $x(0) = 0$, this supply rate implies $\int_0^\tau u(t)^\top y(t) dt \geq 0$ for all u and $\tau \geq 0$.

In the rest of the thesis, we will mainly use supply rates related to stability and $\mathcal{L}_2$ reachability to certify the corresponding properties of systems.

2.3 Integral Quadratic Constraints (IQCs)

Integral quadratic constraints (IQCs) play a key role in the rest of the thesis. We will replace the perturbation Δ with IQCs that encapsulate the input-output properties of Δ. The definitions of IQCs are given as follows.

Continuous-time Case

Definition 2. *Let $\Pi = \Pi^\sim \in \mathbb{RL}_\infty^{(n_v+n_w)\times(n_v+n_w)}$ be given. A bounded, causal operator $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ satisfies the frequency domain IQC defined by the multiplier Π, if the following inequality holds for all $v \in \mathcal{L}_2^{n_v}$ and $w = \Delta(v)$,*

$$\int_{-\infty}^{\infty} \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix}^* \Pi(j\omega) \begin{bmatrix} \hat{v}(j\omega) \\ \hat{w}(j\omega) \end{bmatrix} d\omega \geq 0, \tag{2.10}$$

where $\hat{v}$ and $\hat{w}$ are Fourier transforms of v and w.

To help define IQCs in time domain, we introduced a virtual filter Ψ (shown in Fig 2.1) that is a continuous-time linear time invariant (LTI) system, driven by the input v and output w of Δ, and with zero initial condition $\psi(0) = 0_{n_\psi \times 1}$. Its dynamics are given by

$$\dot{\psi}(t) = A_\psi \psi(t) + B_{\psi 1} v(t) + B_{\psi 2} w(t), \tag{2.11a}$$
$$z(t) = C_\psi \psi(t) + D_{\psi 1} v(t) + D_{\psi 2} w(t), \tag{2.11b}$$

where $\psi(t) \in \mathbb{R}^{n_\psi}$ is the state, and $z(t) \in \mathbb{R}^{n_z}$ is the output. For many types of perturbations (e.g. the ones in Example 1–3), we can choose Ψ to be an identity matrix, i.e., $z = [v; w]$. But dynamic filters are able to capture the relation between the input and output signals of Δ across time, which enriches the description of Δ. For examples on dynamic filters, the reader is referred to [13, 17, 18].

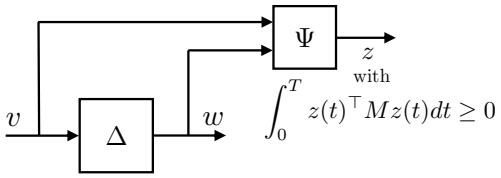

Figure 2.1: Graphical interpretation for time domain IQCs

Definition 3. *Let $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$ and $M \in \mathbb{S}^{n_z}$ be given. A bounded, causal operator $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ satisfies the hard IQC defined by (Ψ, M) if the following condition holds for all $v \in \mathcal{L}_{2e}^{n_v}$, and $w = \Delta(v)$:*

$$\int_{t_0}^{T} z(t)^\top M z(t) dt \geq 0, \ \forall T \geq 0, \tag{2.12}$$

where $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ (Eq. 2.11b) is the output of Ψ driven by the inputs (v, w).

Definition 4. *Let $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$ and $M \in \mathbb{S}^{n_z}$ be given. A bounded, causal operator $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ satisfies the soft IQC defined by (Ψ, M) if the following inequality holds for all $v \in \mathcal{L}_2^{n_v}$ and $w = \Delta(v)$:*

$$\int_{t_0}^{\infty} z(t)^\top M z(t) dt \geq 0, \tag{2.13}$$

where $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ (Eq. 2.11b) is the output of Ψ driven by the inputs (v, w).

Discrete-time Case

Definition 5. *Let $\Pi = \Pi^\sim \in \mathbb{RL}_\infty^{(n_v+n_w)\times(n_v+n_w)}$ be given. A bounded, causal operator $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ satisfies the frequency domain IQC defined by the multiplier Π, if the following inequality holds for all $v \in \ell_2^{n_v}$ and $w = \Delta(v)$,*

$$\int_0^{2\pi} \begin{bmatrix} V(e^{j\omega}) \\ W(e^{j\omega}) \end{bmatrix}^* \Pi(e^{j\omega}) \begin{bmatrix} V(e^{j\omega}) \\ W(e^{j\omega}) \end{bmatrix} d\omega \geq 0 \tag{2.14}$$

where V and W are discrete-time Fourier transforms of v and w.

Again, to help define time domain IQCs, we introduce a 'virtual' filter Ψ applied to the input v and output w of Δ. The filter Ψ is a discrete-time LTI system with zero initial condition $\psi(0) = 0_{n_\psi \times 1}$. Its dynamics are of the form:

$$\psi(k+1) = A_\psi \ \psi(k) + B_{\psi 1} \ v(k) + B_{\psi 2} \ w(k) \tag{2.15a}$$
$$z(k) = C_\psi \ \psi(k) + D_{\psi 1} \ v(k) + D_{\psi 2} \ w(k) \tag{2.15b}$$

where $\psi(k) \in \mathbb{R}^{n_\psi}$ is the state, $z(k) \in \mathbb{R}^{n_z}$ is the output, and A_ψ is a Schur matrix. The state matrices have compatible dimensions. The dynamics of Ψ can be compactly denoted by $\left[\begin{array}{c|cc} A_\psi & B_{\psi 1} & B_{\psi 2} \\ \hline C_\psi & D_{\psi 1} & D_{\psi 2} \end{array}\right]$.

Definition 6. *Let $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$ and $M \in \mathbb{S}^{n_z}$ be given. A bounded, causal operator $\Delta : \ell_{2e}^{n_v} \to \ell_{2e}^{n_w}$ satisfies the time domain hard IQC defined by (Ψ, M) if the following inequality holds for all $v \in \ell_{2e}^{n_v}$, $w = \Delta(v)$ and for all $N \geq 0$*

$$\sum_{k=0}^{N} z(k)^\top M z(k) \geq 0, \tag{2.16}$$

where $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ (Eq. 2.15b) is the output of Ψ driven by the inputs (v, w).

Definition 7. *Let $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$ and $M \in \mathbb{S}^{n_z}$ be given. A bounded, causal operator $\Delta : \ell_{2e}^{n_v} \to \ell_{2e}^{n_w}$ satisfies the time domain soft IQC defined by (Ψ, M) if the following inequality holds for all $v \in \ell_2^{n_v}$, and $w = \Delta(v)$*

$$\sum_{k=0}^{\infty} z(k)^\top M z(k) \geq 0. \tag{2.17}$$

where $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ (Eq. 2.15b) is the output of Ψ driven by the inputs (v, w).

We will use the notations $\Delta \in \text{FreqIQC}(\Pi)$, $\Delta \in \text{SoftIQC}(\Psi, M)$, and $\Delta \in \text{HardIQC}(\Psi, M)$ to indicate that Δ satisfies the corresponding (continuous-time or discrete-time) frequency domain, time domain soft, and time domain hard IQC, respectively.

2.4 Sum-of-Squares (SOS) Programming

If we restrict our attention to LTI systems, searching for their storage functions such that an algebraic expression holds for all values of the independent variables can be formulated as an LMI. However, if the systems under consideration have polynomial vector fields, the corresponding algebraic expressions are higher order polynomials, and checking nonnegativity of polynomials is generally an NP-hard problem [19]. If we can show that a polynomial is a sum of squares of finitely many polynomials, that is to say, it is a sum-of-squares (SOS) polynomial, then it is nonnegative. Moreover, checking whether a polynomial is a SOS polynomial can be formulated as a semidefinite programming. As a result, in the following sections, we will replace polynomial nonnegativity constraints with SOS constraints.

Definition 8. *For $x \in \mathbb{R}^n$, a polynomial $p \in \mathbb{R}[x]$ is a SOS polynomial, if there exist polynomials $p_1, ..., p_N \in \mathbb{R}[x]$ such that $p = \sum_{i=1}^{N} p_i^2$.*

Let $z(x)$ be a vector of all monomials in x up to degree d,

$$z(x) := [1, x_1, x_2, ..., x_n, x_1 x_2, ..., x_n^d]^\top. \tag{2.18}$$

Definition 9. *For every polynomial p with degree $2d$, there is a symmetric matrix Q such that*

$$p(x) = z(x)^\top Q z(x). \tag{2.19}$$

This is called a Gram matrix representation of p.

In general, given a polynomial p, the Q matrix that satisfies (2.19) is not unique. Let Q_0 be a particular solution of (2.19), and let $\{N_i\}_{i=1}^h$ be a basis for the homogeneous solutions, i.e, $z(x)^\top N_i z(x) = 0$. Then Q belongs to the set $\{Q_0 + \sum_{i=1}^h \lambda_i N_i : \lambda \in \mathbb{R}^h\}$. Gram matrix representation plays a key role in sum-of-squares verification.

Theorem 1. *A polynomial p in $x \in \mathbb{R}^n$ of degree 2d, is SOS if and only of there exists $Q \succeq 0$ such that*

$$p(x) = z(x)^\top Q z(x). \tag{2.20}$$

Consequently, p is a SOS polynomial if and only if there exists $\lambda \in \mathbb{R}^h$ such that $Q_0 + \sum_{i=1}^h \lambda_i N_i \succeq 0$. That is to say, checking if a polynomial is a SOS polynomial can be done by solving a SDP. For a polynomial of degree $2d$ in n variables, its corresponding vector of monomials $z(x)$ defined in (2.18), and Gram matrix representation are of the size $m \times 1$, and $m \times m$, where $m := \binom{n+d}{d}$.

Chapter 3

Robust Forward Reachability Analysis for Uncertain Nonlinear Systems

In this chapter, we present a method for finding the smallest achievable outer bounds to the forward reachable sets (FRS) on finite horizons. The FRS is the set of all the successors to a set of initial conditions subject to the given dynamics under all possible model uncertainties and disturbances on a finite horizon. The computation of the FRS plays an important role in safety-critical systems, as it can verify whether a system is able to reach a target and avoid an obstacle [8, 20]. Indeed, if an outer bound avoids obstacles and is encompassed by the target set at the final time, then one can ascertain the same properties for all trajectories.

The algorithm presented in this chapter uses a storage function that satisfies a dissipation inequality to characterize the outer bound. The dissipation inequality framework, combined with the Sum-of-Squares (SOS) technique [21] and the generalized S-procedure [22], allows us to simultaneously accommodate multiple sources of uncertainty, including time varying uncertain parameters, $\mathcal{L}_2$ disturbances, and perturbations Δ whose input output properties are characterized by integral quadratic constraints (IQCs) [13]. IQCs can model a rich class of uncertainties and nonlinearities, including hard nonlinearities (e.g. saturation), and unmodeled dynamics, as summarized in [13] and [18]. Therefore, although our nominal systems are assumed to be polynomials, including IQCs allows us to extend our analysis framework to the class of systems beyond polynomial systems.

The proposed analysis framework considers both hard and soft IQC factorizations. Dissipation inequalities usually require IQCs to hold over all finite horizons (hard IQCs) [23, 24]. However, IQCs are often available in the infinite-time horizon (soft IQCs), while the hard IQCs are not. To mitigate this issue, we incorporate soft IQCs in dissipation inequalities by making use of a lower bound derived from [14], which is valid for soft IQCs over all finite horizons. We formulate the reachable set computation as a set of SOS optimization problems, which can be solved effectively by bisection. In addition, our optimization problems do not require a feasible initialization of the storage function.

In this chapter, we first present the reachability analysis framework for outer-bounding the reachable sets for the nominal system: nonlinear system with $\mathcal{L}_2$ disturbances and time

varying uncertain parameters. Then we extend the framework to the uncertain nonlinear system, which is modeled as an interconnection of nominal system and perturbations Δ described by soft/hard IQCs. Finally, we illustrate the method on several aircraft examples.

3.1 Nominal Reachability Analysis

Consider the nominal nonlinear system N defined on $[t_0, T]$:

$$\dot{x}(t) = f(t, x(t), d(t), \delta(t)), \tag{3.1}$$

where $x(t) \in \mathbb{R}^n$ is the state, $d(t) \in \mathbb{R}^{n_d}$ is the external disturbance, $\delta(t) \in \mathbb{R}^{n_\delta}$ is the time varying uncertain parameter, and the vector field $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_\delta} \to \mathbb{R}^n$ is locally Lipschitz continuous.

Assumption 1. *(i) Functions d and δ are measurable and locally essentially bounded, (ii) the disturbance d satisfies $d \in \mathcal{L}_2^{n_d}$ with $\|d\|_{\mathcal{L}_2,[t_0,T]} < R$ for some $R > 0$, (iii) there exists a non-decreasing polynomial function $h : \mathbb{R} \to \mathbb{R}_{\geq 0}$ with $h(t_0) = 0$, $h(T) = 1$ such that*

$$\int_{t_0}^{t} |d(\tau)|^2 d\tau < R^2 h(t), \ \forall t \in [t_0, T], \tag{3.2}$$

(iv) for each $t \in [t_0, T]$, $\delta(t) \in \mathcal{D} := \{\delta \in \mathbb{R}^{n_\delta} : p_\delta(\delta) \geq 0\}$, where the polyomial $p_\delta \in \mathbb{R}[\delta]$ describes the prior knowledge that bounds the uncertainty δ.

The function h is used to describe how fast the energy of d can be released on the interval $[t_0, T]$. Next, the definition of the forward reachable set (FRS) is given as follows:

Definition 10. *The FRS of the system N (3.1) from $\mathcal{X}_0$ at time T is defined as*

$$FRS(T; N, t_0, \mathcal{X}_0, R, h, \mathcal{D}) := \{x(T) \in \mathbb{R}^n : \exists x(t_0) \in \mathcal{X}_0, \ d \text{ satisfying (3.2) and } \delta(t) \in \mathcal{D}, \text{ such that } x(\cdot) \text{ is a solution to (3.1)}\}.$$

Our goal is to outer bound this FRS, and the following theorem provides a way of achieving it based on dissipation inequalities.

Theorem 2. *Let Assumption 1 hold. Given vector field $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_\delta} \to \mathbb{R}^n$, time interval $[t_0, T]$, local region $\mathcal{X}_l \subset \mathbb{R}^n$, set of initial conditions $\mathcal{X}_0 \subset \mathbb{R}^n$, disturbance bound R, function h, and set of uncertain parameters $\mathcal{D}$, suppose there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ that satisfies*

$$\partial_t V(t,x) + \partial_x V(t,x) \cdot f(t,x,d,\delta) \leq d^\top d, \quad \forall (t,x,d,\delta) \in [t_0,T] \times \mathcal{X}_l \times \mathbb{R}^{n_d} \times \mathcal{D}, \tag{3.3a}$$

$$\mathcal{X}_0 \subseteq \Omega^V_{t_0,0}, \tag{3.3b}$$

$$\Omega^V_{t,R^2h(t)} \subseteq \mathcal{X}_l, \quad \forall t \in [t_0, T]. \tag{3.3c}$$

Then $x(T) \in \Omega^V_{T,R^2}$ for all $x(t_0) \in \mathcal{X}_0$, where $x(\cdot)$ is a solution to the system N (3.1), and $\Omega^V_{t_0,0}, \Omega^V_{t,R^2h(t)}$ and Ω^V_{T,R^2} are defined in (2.2). Therefore Ω^V_{T,R^2} is an outer bound to the $FRS(T; N, t_0, \mathcal{X}_0, R, h, \mathcal{D})$.

Proof. Combining constraints (3.3a) and (3.3c), we have the following dissipation inequality:

$$\partial_t V(t,x) + \partial_x V(t,x) \cdot f(t,x,d,\delta) \leq d^\top d, \quad \forall (t,x,d,\delta), \\ \text{s.t. } t \in [t_0, T],\ x \in \Omega^V_{R^2 h(t)},\ d \in \mathbb{R}^{n_d},\ \delta \in \mathcal{D}.$$

Since this dissipation inequality only holds on the set $\Omega^V_{t,R^2h(t)}$, we need to first prove that all the states starting from $\mathcal{X}_0$ won't leave $\Omega^V_{t,R^2h(t)}$, for all $t \in [t_0, T]$. Assume there exist a time instance $T_1 \in [t_0, T]$, $x_0 \in \mathcal{X}_0$, and signals d satisfying (3.2), $\delta(t) \in \mathcal{D}$, such that a trajectory of the system N starting from $x(t_0) = x_0$ satisfies $V(T_1, x(T_1)) > R^2 h(T_1)$. Define $T_2 = \inf_{V(t,x(t)) > R^2 h(t)} t$. Therefore, the dissipation inequality holds on $[t_0, T_2]$, and we can integrate it over $[t_0, T_2]$:

$$V(T_2, x(T_2)) - V(t_0, x(t_0)) \leq \int_{t_0}^{T_2} |d(t)|^2 dt.$$

By assumption $x_0 \in \mathcal{X}_0$, it follows from (3.3b) that $V(t_0, x(t_0)) \leq 0$. Combing it with d satisfying (3.2) to show

$$R^2 h(T_2) = V(T_2, x(T_2)) < R^2 h(T_2).$$

This is contradictory. Therefore there doesn't exist a $T_1 \in [t_0, T]$, such that $x(T_1) \notin \Omega^V_{T_1,R^2h(T_1)}$. As a result, for all $x(t_0) \in \mathcal{X}_0$, we have $x(t) \in \Omega^V_{t,R^2h(t)}$, for all $t \in [t_0, T]$, and thus $x(T) \in \Omega^V_{T,R^2}$. □

If the function h is not given, then there is no *a priori* knowledge on how $\int_{t_0}^t |d(\tau)|^2 d\tau$ depends on t. In this case the constraint (3.3c) is modified to be

$$\Omega^V_{t,R^2} \subseteq \mathcal{X}_l, \quad \forall t \in [t_0, T]. \tag{3.4}$$

This case is more restrictive for the storage function and yields larger outer bounds on the FRS.

We are interested in a tight outer bound to the FRS. Thus it is natural to search for a storage function V that minimizes the volume of Ω^V_{T,R^2}. However, an explicit expression is not available for the volume of Ω^V_{T,R^2} for a generic storage function. Instead, we introduce a user-specified shape function q and its corresponding variable sized region $\Omega^q_\alpha = \{x \in \mathbb{R}^n : q(x) \leq \alpha\}$. The shape function q can be associated with the user's initial guess of the shape of the actual reachable set or can signify the desired shape of the outer bound. An example of q is given in Section 3.1. The volume of Ω^V_{T,R^2} can be shrunk, by enforcing

$$\Omega^V_{T,R^2} \subseteq \Omega^q_\alpha, \tag{3.5}$$

while minimizing α. For more heuristic metrics for the volume of semi-algebraic sets, the reader is referred to [25].

To find a storage function V that satisfies the constraints in (3.3) and (3.5), we leverage SOS programming. To do so, we assume that $\mathcal{X}_0$ and $\mathcal{X}_l$ are semi-algebraic sets: $\mathcal{X}_0 := \{x \in \mathbb{R}^n : r_0(x) \leq 0\}$, and

$$\mathcal{X}_l := \{x \in \mathbb{R}^n : p(x) \leq \eta\}, \tag{3.6}$$

where r_0, $p \in \mathbb{R}[x]$ are specified by the user, and $\eta \in \mathbb{R}$ is a decision variable that determines that volume of $\mathcal{X}_l$. Additionally, we restrict the system model, shape function, and storage function to polynomials, i.e., $f \in \mathbb{R}^n[(t, x, d, \delta)]$, $q \in \mathbb{R}[x]$, $V \in \mathbb{R}[(t, x)]$. Also define $g(t) := (t - t_0)(T - t)$, whose value is nonnegative when $t \in [t_0, T]$. The polynomial functions are used to formulate the set containment constraints. With these ideas, sufficient SOS conditions for the set containment constraints (3.3) and (3.5) are obtained. Also by choosing α as the cost function, we obtain the following SOS optimization problem, denoted as $\mathbf{sosopt}_1(f, p, g, q, r_0, R, h, p_\delta)$,

$$\begin{aligned}
&\min_{\alpha,\eta,s,V} \quad \alpha \\
&\text{s.t.} \quad s_5 - \epsilon_1 \in \Sigma[x], s_6 - \epsilon_2 \in \Sigma[(x,t)], \epsilon_1 > 0, \epsilon_2 > 0, \\
&\qquad s_i \in \Sigma[(x,d,\delta,t)], \forall i \in \{1,2,3\}, s_4 \in \Sigma[x], s_7 \in \Sigma[(x,t)], V \in \mathbb{R}[(t,x)], && \text{(3.7a)} \\
&\qquad -\left(\partial_t V + \partial_x V \cdot f - d^\top d\right) + (p - \eta)s_1 - s_2 g - s_3 p_\delta \in \Sigma[(x,d,\delta,t)], && \text{(3.7b)} \\
&\qquad -V|_{t=t_0} + s_4 r_0 \in \Sigma[x], && \text{(3.7c)} \\
&\qquad -(p - \eta)s_6 + V - R^2 h - s_7 g \in \Sigma[(x,t)], && \text{(3.7d)} \\
&\qquad -(q - \alpha)s_5 + V|_{t=T} - R^2 \in \Sigma[x], && \text{(3.7e)}
\end{aligned}$$

where $s_i, i \in \{1, ..., 7\}$, are SOS polynomials, called multipliers, whose coefficients are to be determined, ϵ_1 and ϵ_2 are small positive numbers chosen by the user to guarantee that s_5 and s_6 cannot take the value of zero. The optimization sosopt_1 is nonconvex as it is bilinear in two groups of decision variables (α, η) and (s_1, s_5, s_6). Since we can't bisect on both α and η at the same time, we propose Algorithm 1 that solves the problem in two steps, and bisects on one decision variable at one step.

Algorithm 1 Computing the outer bound

Input: $f, p, g, q, r_0, R, h, p_\delta$

1: **Preparation Step:** solve for $\eta^\star = \arg\min \eta$ s.t. (3.7a)–(3.7d) by bisecting on η.

2: **Main Step:** solve for $\alpha^\star = \arg\min \alpha$ s.t. (3.7a)–(3.7e) by using $\eta = \eta^\star$ and bisecting on α.

Output: Minimized $\alpha^\star$, outer bound Ω^V_{T,R^2}.

The first step of Algorithm 1 is to find the smallest feasible local region $\mathcal{X}_l$ (with respect to p) by setting aside the original objective function and constraint (3.7e), and minimizing η. The second is to find the least conservative outer bound. The first and second steps bisect

on η and α, respectively. Each iteration of bisection involves holding α/η fixed and solving a feasibility problem, which is a standard semidefinite programming problem and is convex. If the fixed value of α/η leads to infeasibility of the problem, then try to solve it with a larger α/η; otherwise, decrease the value of α/η.

Proposition 1. *The SOS constraints* (3.7b)–(3.7e) *are sufficient conditions for (3.3) and (3.5).*

Proof. (3.7b) $\Rightarrow$ (3.3a): The proof follows from the generalized S-procedure [22]. In (3.7b), when (x, t, δ) satisfies $p(x) \leq \eta$ (i.e. $x \in \mathcal{X}_l$), $g(t) \geq 0$ (i.e. $t \in [t_0, T]$), $p_\delta \geq 0$ (i.e. $\delta \in \mathcal{D}$), for the polynomial in (3.7b) to be nonnegative, then $-(\partial_t V(t,x) + \partial_x V(t,x) \cdot f(t,x,d,\delta) - d^\top d)$ must be nonnegative. Thus (3.7b) implies (3.3a).

(3.7c) $\Rightarrow$ (3.3b): In (3.7c), when a state x satisfies $r_0(x) \leq 0$ (i.e. $x \in \mathcal{X}_0$), for the polynomial in (3.7c) to be nonnegative, then $-V(t_0, x)$ must be nonnegative (i.e. $x \in \Omega^V_{t_0,0}$).

(3.7d) $\Rightarrow$ (3.3c): In (3.7d), when a state and time pair (x, t) satisfies $V(t,x) \leq R^2 h(t)$ (i.e. $x \in \Omega^V_{t,R^2h(t)}$) and $g(t) \geq 0$ (i.e. $t \in [t_0, T]$), for the polynomial in (3.7d) to be nonnegative, then $-s_6(t,x)(p(x) - \eta)$ must be nonnegative (i.e. $x \in \mathcal{X}_l$).

(3.7e) $\Rightarrow$ (3.5): In (3.7e), when a state x satisfies $V(T,x) \leq R^2$ (i.e. $x \in \Omega^V_{T,R^2}$), for the polynomial in (3.7e) to be nonnegative, then $-(q(x) - \alpha)s_5(x)$ must be nonnegative (i.e. $x \in \Omega^q_\alpha$). □

Application to a 2-state example

Consider the following academic example from [26]:

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2 - x_1 x_2^2, \\ \dot{x}_2 &= -x_2 - x_1^2 x_2 + d, \end{aligned} \tag{3.8}$$

where d is the disturbance satisfies (3.2) with $R = 1$ and $h(t) = t^2/T^2$. In this example the uncertain parameter is not considered. We take $[t_0, T] = [0, 1 \text{ sec}]$, $r_0(x) = x^\top x - 1$. In Figure 3.1, the green points are simulation points $x(T)$, at $T = 1$ sec, for the system (3.8) using disturbance signals d, with initial conditions inside $\mathcal{X}_0$, which is shown with the red dotted curve. In this example, the shape function q is obtained by computing the minimum volume ellipsoid Ω^q_1 that contains all the simulation points $x(T)$ at $T = 1$ sec, and $q(x) = 4.84x_1^2 - 3.05x_1x_2 + 1.50x_2^2$. A more accurate shape function can be obtained by fitting a higher degree polynomial to the simulation points [27]. Here, the polynomial p that defines the local region is obtained by computing the minimum volume ellipsoid Ω^p_1 that contains some sampling points on the simulation trajectories $x(t)$, for $t \in [0, T]$, and $p = 0.989x_1^2 - 0.051x_1x_2 + 0.949x_2^2 + 0.001x_1 + 0.001x_2$. Solving the first step of Algorithm 1 we obtain $\eta^\star = 1.044$, and $\mathcal{X}_l$ is determined. Solving the second step gives $\alpha^\star = 1.37$. The outer bound is shown with the black curve, which tightly encloses all $x(T)$.

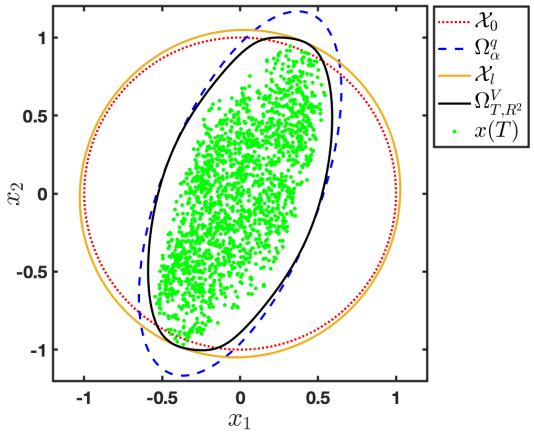

Figure 3.1: Outer bound of reachable set at $T = 1$ sec for the 2-state example with $\mathcal{L}_2$ disturbance.

3.2 Robust Reachability Analysis with Hard IQCs

Consider the uncertain nonlinear system shown in Figure 1.1, which is an interconnection $F_u(G, \Delta)$ of a nonlinear system G and a perturbation Δ. The dynamics of the nonlinear system G are of the form:

$$\begin{aligned} \dot{x}_G(t) &= f(t, x_G(t), w(t), d(t), \delta(t)), \\ v(t) &= r(t, x_G(t), w(t), d(t), \delta(t)), \end{aligned} \tag{3.9}$$

where $x_G(t) \in \mathbb{R}^{n_G}$ is the state of G, and $\delta(t) \in \mathbb{R}^{n_\delta}$ is the uncertain parameter. The inputs of G are $d(t) \in \mathbb{R}^{n_d}$ and $w(t) \in \mathbb{R}^{n_w}$, while the output is $v(t) \in \mathbb{R}^{n_v}$. The system G is defined by the mappings $f : \mathbb{R} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_\delta} \to \mathbb{R}^{n_G}$ and $r : \mathbb{R} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d} \times \mathbb{R}^{n_\delta} \to \mathbb{R}^{v}$. The perturbation is a bounded and causal operator $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$. Assume the interconnection $F_u(G, \Delta)$ formed by G and Δ through the constraint

$$w(\cdot) = \Delta(v(\cdot)) \tag{3.10}$$

is well-posed. The well-posedness of the interconnection $F_u(G, \Delta)$ is defined as follows.

Definition 11. *$F_u(G, \Delta)$ is well-posed if for all $x_G(t_0) \in \mathbb{R}^{n_G}$ and $d \in \mathcal{L}_{2e}^{n_d}$ there exist unique solutions $x_G \in \mathcal{L}_{2e}^{n_G}$, $v \in \mathcal{L}_{2e}^{n_v}$, and $w \in \mathcal{L}_{2e}^{n_w}$ satisfying* (3.9) *and* (3.10) *with a causal dependence on d.*

Again, assume all the trajectories of $F_u(G, \Delta)$ start from $x_G(t_0) \in \mathcal{X}_0 \subset \mathbb{R}^{n_G}$. The FRS of $F_u(G, \Delta)$ from $\mathcal{X}_0$ at time T is then defined as

$$\begin{aligned} FRS(T; F_u(G, \Delta), t_0, \mathcal{X}_0, R, h, \mathcal{D}) := \{ & x_G(T) \in \mathbb{R}^{n_G} : \exists x_G(t_0) \in \mathcal{X}_0, \ d \text{ satisfying (3.2) and} \\ & \delta(t) \in \mathcal{D}, \text{ such that } x_G(\cdot) \text{ is a solution to (3.9) and (3.10)} \}. \end{aligned} \tag{3.11}$$

From robust control modeling [28], the perturbation Δ can represent various types of nonlinearity and uncertainty, including hard nonlinearities (e.g. saturation), and unmodeled dynamics. Different types of perturbation have different input-output properties, and each property can be described by its corresponding IQCs [13]. IQCs can be either defined in frequency domain or time domain. The use of time domain IQCs is required by the dissipation inequality used in the dissertation. Time domain IQCs consist of soft IQCs and hard IQCs, which are quadratic constraints on the output z associated with a matrix M over infinite (soft IQC) or finite (hard IQC) horizons. The definition for hard IQCs is given in Definition 3, the use of soft IQCs is discussed in Section 3.3.

We use the notation $\Delta \in \text{HardIQC}(\Psi, M)$ to indicate that Δ satisfies the hard IQC specified (Ψ, M), i.e., given any input v of Δ, the output w must be such that $z = \Psi \left[\begin{smallmatrix} v \\ w \end{smallmatrix}\right]$ satisfies the constraint (2.12) characterized by (Ψ, M). Next, we give two examples on different types of uncertainties and the corresponding hard IQCs.

Example 1. *Consider the set $\mathcal{S}_1$ of LTI uncertainties with a given norm bound $\sigma > 0$, i.e., $\Delta \in \mathcal{S}_1$, if $\Delta \in \mathbb{RH}_\infty$ with $\|\Delta\|_\infty \leq \sigma$. It's proved in [29] that $\Delta \in HardIQC(\Psi, M_D)$ over any finite horizon $T < \infty$, where $\Psi := \left[\begin{smallmatrix} \Psi_{11} & 0 \\ 0 & \Psi_{11} \end{smallmatrix}\right]$ with $\Psi_{11} \in \mathbb{RH}_\infty^{n_z \times 1}$ and*

$$M_D \in \mathcal{M}_1 := \left\{ \begin{bmatrix} \sigma^2 M_{11} & 0 \\ 0 & -M_{11} \end{bmatrix} : M_{11} \succeq 0 \right\}. \tag{3.12}$$

A typical choice for Ψ_{11} [18] is

$$\Psi_{11}^{d,m} = \left[1, \tfrac{1}{(s+m)}, \cdots, \tfrac{1}{(s+m)^d}\right]^\top, \text{with} \;\; m > 0, \tag{3.13}$$

where m and d are selected by the user.

Example 2. *Consider the set $\mathcal{S}_2$ of nonlinear, time varying, uncertainties with a given norm-bound σ, i.e. $\Delta \in \mathcal{S}_2$, if $\|\Delta\|_{\mathcal{L}_2 \to \mathcal{L}_2, [t_0, T]} \leq \sigma$. If $\Delta \in \mathcal{S}_2$, then the perturbation Δ satisfies the hard IQCs defined by (Ψ, M) defined below:*

$$\Psi = I_{n_v + n_w}, \;\; M \in \mathcal{M}_2 := \left\{ \begin{bmatrix} \sigma^2 \lambda I_{n_v} & 0 \\ 0 & -\lambda I_{n_w} \end{bmatrix} : \;\; \lambda \geq 0 \right\}. \tag{3.14}$$

Since the behavior of the perturbation Δ can be described by an IQC associated with a filter Ψ and a matrix M, then the robust analysis on the original uncertain system $F_u(G, \Delta)$ can be instead conducted on the extended system shown in Fig. 3.2 with an additional constraint (2.12). The precise relation $w = \Delta(v)$, for analysis, is replaced by the constraint on z. This extended system is an interconnection of G and Ψ, with Δ been removed. The dynamics of the extended system are of the form:

$$\dot{x}(t) = F(t, x(t), w(t), d(t), \delta(t)), \tag{3.15a}$$

$$z(t) = H(t, x(t), w(t), d(t), \delta(t)), \tag{3.15b}$$

where the state $x := [x_G; \psi] \in \mathbb{R}^n, n = n_G + n_\psi$, gathers the state of G and Ψ. The mappings F, and H are given by (dropping the dependence on t):

$$\begin{aligned} F(t, x, w, d, \delta) := &\begin{bmatrix} f(t, x_G, w, d, \delta) \\ A_\psi \psi + B_{\psi 1} r(t, x_G, w, d, \delta) + B_{\psi 2} w \end{bmatrix}, \\ H(t, x, w, d, \delta) := &\; C_\psi \psi + D_{\psi 1} r(t, x_G, w, d, \delta) + D_{\psi 2} w, \end{aligned} \tag{3.16}$$

where A_ψ, $B_{\psi 1}$, $B_{\psi 2}$, C_ψ, $D_{\psi 1}$, and $D_{\psi 2}$ are state space matrices of Ψ defined in (2.11).

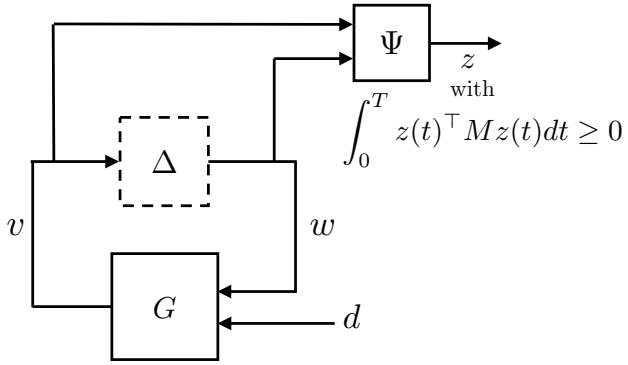

Figure 3.2: Extended system of G and Ψ

The original uncertain system to be analyzed is $F_u(G, \Delta)$, which has a set of initial conditions $\mathcal{X}_0$ and an input d. The analysis is instead conducted on the extended system (3.15), which has a set of initial conditions $\mathcal{X}_0 \times \{0^{n_\psi}\}$, and two inputs d and w. For any input $d \in \mathcal{L}_2^{n_d}$ and initial condition $x_G(t_0) \in \mathbb{R}^{n_G}$, the solutions $v \in \mathcal{L}_{2e}^{n_v}$ and $w \in \mathcal{L}_{2e}^{n_w}$ to the original system $F_u(G, \Delta)$ satisfy the constraint (2.12). The extended system (3.15) with the IQC (2.12) "covers" the responses of the original uncertain system $F_u(G, \Delta)$. Specifically, given any input $d \in \mathcal{L}_2^{n_d}$ and initial condition $x_G(t_0) \in \mathbb{R}^{n_G}$, the input $w \in \mathcal{L}_{2e}^{n_w}$ is implicitly constrained in the extended system so that the pair (v, w) satisfies the IQC (2.12). This set of (v, w) that satisfies the IQC (2.12) includes all input/output pairs of Δ. Therefore, the response of this extended system subject to this implicit constraint (2.12) includes all behaviors of the original uncertain system $F_u(G, \Delta)$. The following theorem provides the method for outer bounding the FRS of the uncertain system $F_u(G, \Delta)$ by conducting analysis on the constrained extended system (3.15).

Theorem 3. *Let G be a nonlinear system defined by (3.9), and $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ be a bounded and causal operator. Let Assumption 1 hold. Additionally, assume (i) $F_u(G, \Delta)$ is well-posed, (ii) $\Delta \in$ HardIQC(Ψ, M), with Ψ and M given, and (iii) all the trajectories of the extended system start from $\mathcal{X}_0 \times \{0^{n_\psi}\}$. For some F, H defined in (3.16), time interval $[t_0, T]$, local region $\mathcal{X}_l \subset \mathbb{R}^{n_G}$, set of initial conditions $\mathcal{X}_0 \subset \mathbb{R}^{n_G}$, disturbance bound R, function h, and set of uncertain parameters $\mathcal{D}$, function $q : \mathbb{R}^{n_G} \to \mathbb{R}$, and $\alpha \in \mathbb{R}$, suppose there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ that satisfies*

$$\partial_t V(t, x) + \partial_x V(t, x) \cdot F(t, x, w, d, \delta) + z^\top M z \leq d^\top d,$$

$$\forall (x,t,w,d,\delta) \in \mathcal{X}_l \times \mathbb{R}^{n_\psi} \times [t_0,T] \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d} \times \mathcal{D}, \quad (3.17a)$$

$$\mathcal{X}_0 \times \{0^{n_\psi}\} \subseteq \{x \in \mathbb{R}^n : V(t_0,x) \leq 0\}, \quad (3.17b)$$

$$\left\{x_G \in \mathbb{R}^{n_G} : V(T,x) \leq R^2\right\} \subseteq \Omega^q_\alpha, \ \forall \psi \in \mathbb{R}^{n_\psi}, \quad (3.17c)$$

$$\left\{x_G \in \mathbb{R}^{n_G} : V(t,x) \leq R^2 h(t)\right\} \subseteq \mathcal{X}_l, \ \forall (t,\psi) \in [t_0,T] \times \mathbb{R}^{n_\psi}, \quad (3.17d)$$

where z is the output of the map H. Then all trajectories of $F_u(G,\Delta)$ (defined by (3.9)–(3.10)) starting from $x_G(t_0) \in \mathcal{X}_0$ satisfy $x_G(T) \in \Omega^q_\alpha$. Therefore Ω^q_α is an outer bound to the $FRS(T; F_u(G,\Delta), t_0, \mathcal{X}_0, R, h, \mathcal{D})$ (3.11).

Proof. By assumption that $F_u(G,\Delta)$ is well-posed, the signals (x,v,w,z) generated for the extended system for the input $d \in \mathcal{L}_2^{n_d}$ are $\mathcal{L}_{2e}$ signals. By combining (3.17a) and (3.17d) we have the following dissipation inequality:

$$\partial_t V(t,x) + \partial_x V(t,x) \cdot F(t,x,w,d,\delta) + z^\top M z \leq d^\top d,$$
$$\forall (x,t,w,d,\delta) \text{ s.t. } x \in \Omega^V_{t,R^2h(t)}, t \in [t_0,T], w \in \mathbb{R}^{n_w}, d \in \mathbb{R}^{n_d}, \delta \in \mathcal{D}. \quad (3.18)$$

Since (3.18) only holds on the set $\Omega^V_{t,R^2h(t)}$, we need to first prove that all the states starting from $\mathcal{X}_0 \times \{0^{n_\psi}\}$ won't leave $\Omega^V_{t,R^2h(t)}$, for all $t \in [t_0,T]$. Assume there exist a time instance $T_1 \in [t_0,T]$, $x_0 \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, and signals d satisfying (3.2), $\delta(t) \in \mathcal{D}$, $w(t) \in \mathbb{R}^{n_w}$, such that a trajectory of the extended system starting from $x(t_0) = x_0$ satisfies $V(T_1, x(T_1)) > R^2 h(T_1)$. Define $T_2 = \inf_{V(t,x(t)) > R^2h(t)} t$, and integrate (3.18) over $[t_0, T_2]$:

$$V(T_2,x(T_2)) - V(t_0,x(t_0)) + \int_{t_0}^{T_2} z(t)^\top M z(t) dt \leq \int_{t_0}^{T_2} d(t)^\top d(t) dt.$$

By assumnption $x_0 \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, it follows from constraint (3.17b) that $V(t_0,x(t_0)) \leq 0$. Comibining it with w satisfing (3.2) yields

$$V(T_2,x(T_2)) + \int_{t_0}^{T_2} z(t)^\top M z(t) dt < R^2 h(T_2). \quad (3.19)$$

Next it follows from the hypothesis that $\Delta \in \text{HardIQC}(\Psi, M)$ that

$$R^2 h(T_2) = V(T_2,x(T_2)) < R^2 h(T_2). \quad (3.20)$$

We can see the contradiction in (3.20). Therefore there doesn't exist a $T_1 \in [t_0,T]$, such that $x(T_1) \notin \Omega^V_{T_1,R^2h(T_1)}$. As a result, for all $x(t_0) \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, we have $x(t) \in \Omega^V_{t,R^2h(t)}$, for all $t \in [t_0,T]$, and thus $x(T) \in \Omega^V_{T,R^2}$. Finally, it follows from (3.17c) that $x_G(T) \in \Omega^q_\alpha$. □

Notice that from the proof, Ω^V_{T,R^2} is an outer bound to the FRS of the extended system from $\mathcal{X}_0 \times \{0^{n_\psi}\}$. The set Ω^q_α, a projection of Ω^V_{T,R^2} on the x_G space, is an outer bound to the FRS of the actual uncertain system $F_u(G,\Delta)$.

There is a large library of IQCs for various types of perturbations Δ [13]. It is common to formulate optimization problems that search over combinations of valid IQCs. Specifically, let $\{(\Psi_k, M_k)\}_{k=1}^N$ be a collection of valid time-domain IQCs for a particular Δ. If z_k is the output of the filter Ψ_k and $\lambda_1, ..., \lambda_N$ are non-negative scalars then it follows that:

$$\int_{t_0}^{T} \sum_{k=1}^{N} \lambda_k z_k(t)^\top M_k(t) z_k(t) dt \geq 0, \ \forall v_k \in \mathcal{L}_{2e}^{n_{v_k}}, \ \ w_k = \Delta(v_k), \ \text{ and } \ T \geq t_0.$$

In other words, a conic combination of time-domain IQCs is also an IQC. This conic combination can be represented as $\Psi := [\Psi_1; ...; \Psi_N]$ and $M := blkdiag(\lambda_1 M_1, ..., \lambda_N M_N)$. The scalars $\lambda_1, ...\lambda_N \geq 0$ are typically decision variables in an optimization used to find the best IQC for the robustness analysis. In this parameterization Ψ is fixed and M is a linear function of variables $\lambda_1, ..., \lambda_N$ subject to non-negativity constraints. More general IQC parameterizations can be found in [18]: given the type of the perturbation, the corresponding IQCs are parametrized by a fixed filter Ψ chosen by the analyst and M in a feasible set $\mathcal{M}$ described by linear matrix inequality (LMI) constraints. These general parametrizations will be used in the rest of the dissertation. Note that Example 1 and 2 also provide instances of the general parametrization, where M_D and M are restricted to convex sets $\mathcal{M}_1$ and $\mathcal{M}_2$.

Along with V, we also treat $M \in \mathcal{M}$ as a decision variable to give the optimization more flexibility. Assume the set $\mathcal{M}$ is convex and described by LMIs. Again, assume $\mathcal{X}_l$, $\mathcal{X}_0$ are parametrized by $p \in \mathbb{R}[x_G]$ and $r_0 \in \mathbb{R}[x_G]$, respectively, and restrict $q \in \mathbb{R}[x_G]$, $f \in \mathbb{R}^{n_G}[(t, x_G, w, d, \delta)]$, and $r \in \mathbb{R}^{n_v}[(t, x_G, w, d, \delta)]$. Therefore, F and H in (3.15) are polynomials. By applying the generalized S-procedure [22] to (3.17), we obtain the following SOS optimization problem, $\mathbf{sosopt}_2(F, H, p, g, q, r_0, R, h, p_\delta, \Psi, \mathcal{M})$,

$$\begin{aligned}
\min_{\alpha,\eta,s,V,M,\epsilon_1,\epsilon_2} \quad & \alpha \\
\text{s.t.} \quad & s_5 - \epsilon_1 \in \Sigma[x], s_6 - \epsilon_2 \in \Sigma[(x,t)], \epsilon_1 > 0, \epsilon_2 > 0, \ M \in \mathcal{M}, \ V \in \mathbb{R}[(t,x)], \\
& s_4 \in \Sigma[x_G], \ s_7 \in \Sigma[(x,t)], \ s_i \in \Sigma[(x,d,w,t,\delta)], \ \forall i \in \{1,2,3\}, \\
& -\left(\partial_t V + \partial_x V \cdot F + z^\top M z - d^\top d\right) + (p-\eta)s_1 - s_2 g \\
& \qquad\qquad - s_3 p_\delta \in \Sigma[(x,d,w,t,\delta)], \quad \text{(3.21a)} \\
& -V|_{t=t_0, x=[x_G; 0^{n_\psi}]} + s_4 r_0 \in \Sigma[x_G], \quad \text{(3.21b)} \\
& -(q-\alpha)s_5 + V|_{t=T} - R^2 \in \Sigma[x], \quad \text{(3.21c)} \\
& -(p-\eta)s_6 + V - R^2 h - s_7 g \in \Sigma[(x,t)], \quad \text{(3.21d)}
\end{aligned}$$

which is again bilinear in (α, η) and (s_1, s_5, s_6), and can be solved by using Algorithm 1. Although in the SOS formulation, M is restricted to be time-invariant, extensions to allow for time-varying M are possible.

To keep track of all the tuning parameters in the dissertation, we provide a table that summarizes them, their corresponding physical meanings, and some of their examples:

Table 3.1: List of tuning parameters

Physical meanings	Shape of $\mathcal{X}_l$	Outer bound shape	Energy releasing rate	Filter for Δ
Parameters	p	q	h	Ψ
Examples	Sections 3.1, 3.4	Sections 3.1, 3.4	Section 3.1	Sections 3.4, 3.4

3.3 Robust Reachability Analysis with Soft IQCs

The previous section gives the result using hard IQCs, however, the library of IQCs are usually provided in frequency domain [13], whose definition is given in Definition 2.

The frequency domain multiplier can be factorized as $\Pi = \Psi^{\sim} M \Psi$ where $M \in \mathbb{S}^{n_z}$ and Ψ is a stable, LTI system of appropriate dimension. Such a factorization always exists [18] but is not unique. This factorization (Ψ, M) gives rise to a time-domain soft IQC as defined in Definition 4.

We use the notation $\Delta \in \text{FreqIQC}(\Pi)$ and $\Delta \in \text{SoftIQC}(\Psi, M)$ to indicate that Δ satisfies the corresponding frequency domain and time domain soft IQC, meaning that given any v, the output w of Δ must be such that (2.10) and (2.13) hold, respectively. By Parseval's theorem [28], frequency domain and time domain soft IQCs are equivalent. Specifically, if $\Delta \in \text{FreqIQC}(\Pi)$ then $\Delta \in \text{SoftIQC}(\Psi, M)$ for any factorization $\Pi = \Psi^{\sim} M \Psi$ with Ψ stable. Conversely if $\Delta \in \text{SoftIQC}(\Psi, M)$ then $\Delta \in \text{FreqIQC}(\Psi^{\sim} M \Psi)$ as well. It also follows that $\Delta \in \text{HardIQC}(\Psi, M)$ implies $\Delta \in \text{FreqIQC}(\Psi^{\sim} M \Psi)$. However, $\Delta \in \text{FreqIQC}(\Pi)$ does not imply, for general factorizations, that $\Delta \in \text{HardIQC}(\Psi, M)$. As a result, soft IQCs are always available while hard ones are not, which necessitates the use of soft IQCs in the dissipation inequality. Next, we give one example of uncertainty and its corresponding soft IQC.

Example 3. *Consider the set $\mathcal{S}_3$ of real constant parametric uncertainties with given norm bound $\sigma > 0$, i.e. $\Delta \in \mathcal{S}_3$, if $w(t) = \Delta(v(t)) = \delta_{TI} v(t)$ with $|\delta_{TI}| \leq \sigma$. From [13], the frequency domain filter is chosen as $\Pi_\delta = \begin{bmatrix} \sigma^2 \Pi_{11}(j\omega) & \Pi_{12}(j\omega) \\ \Pi_{12}^*(j\omega) & -\Pi_{11}(j\omega) \end{bmatrix}$, where $\Pi_{11}(j\omega) = \Pi_{11}^*(j\omega) \geq 0$ and $\Pi_{12}(j\omega) = -\Pi_{12}^*(j\omega)$ for all ω. A soft IQC factorization for Π_δ is $\Psi = \begin{bmatrix} \Psi_{11}^{d,m} & 0 \\ 0 & \Psi_{11}^{d,m} \end{bmatrix}$, where $\Psi_{11}^{d,m}$ is defined in (3.13), and $M_{DG} = \begin{bmatrix} \sigma^2 M_{11} & M_{12} \\ M_{12}^\top & -M_{11} \end{bmatrix}$, where decision matrices are subject to $M_{11} = M_{11}^\top$, $M_{12} = -M_{12}^\top$ and $\Psi_{11}^{d,m\sim} M_{11} \Psi_{11}^{d,m} \geq 0$. The constraints $\Psi_{11}^{d,m\sim} M_{11} \Psi_{11}^{d,m} \geq 0$ can be enforced by a KYP LMI [30]. Notice that δ_{TI} is a special case of the perturbation considered in Example 1, and thus $\delta_{TI} \in \text{HardIQC}(\Psi, M_D)$ as well. However, since M_D is a special case of M_{DG} with $M_{12} \equiv 0$, the reachability analysis using (Ψ, M_{DG}) can be less conservative than using (Ψ, M_D). A method is proposed in [31] to iteratively refine the choice of Ψ.*

Soft IQCs are constraints that hold over the infinite time horizon and hence they cannot be directly incorporated in the analysis based on finite-horizon dissipation inequalities. The

following Lemma is a remedy for this issue: it provides a lower-bound for soft IQCs on finite horizons then enabling their use for reachability analysis. This in turn enables us to: (i) conduct reachability analysis when the hard IQC factorization does not exist; (ii) reduce conservatism resulting from the hard IQC factorization when it exists, as discussed in Example 3 .

Lemma 1. *([14]) Let $\Psi \in \mathbb{RH}_\infty^{n_z \times (n_v+n_w)}$ and $M \in \mathbb{S}^{n_z}$ be given. Define $\Pi := \Psi^\sim M \Psi$. If $\Pi_{22}(j\omega) < 0 \ \forall \omega$, then*[*]

- *$D_{\psi 2}^\top M D_{\psi 2} < 0$ and there exists a $Y_{22} \in \mathbb{S}^{n_\psi}$ satisfying*

$$KYP(Y_{22}, A_\psi, B_{\psi 2}, C_\psi, D_{\psi 2}, M) < 0. \tag{3.22}$$

- *If $\Delta \in SoftIQC(\Psi, M)$ then for all $T \geq 0$, $v \in \mathcal{L}_{2e}^{n_v}$ and $w = \Delta(v)$,*

$$\int_0^T z(t)^\top M z(t) dt \geq -\psi(T)^\top Y_{22} \psi(T) \tag{3.23}$$

  for any $Y_{22} \in \mathbb{S}^{n_\psi}$ satisfying (3.22)*.*

Lemma 1 is valid for multipliers that satisfy $\Pi_{22} < 0$. Multipliers satisfying the non-strict conditions $\Pi_{22} \leq 10$ can be handled by a perturbation argument [32]. Based on the lemma given above, the following theorem considers the analysis for the interconnection $F_u(G, \Delta)$ with Δ that has a soft IQC factorization.

Theorem 4. *Let G be a nonlinear system defined by* (3.9)*, and $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ be a bounded and causal operator. Let Assumption 1 hold. Additionally, assume (i) $F_u(G, \Delta)$ is well-posed, (ii) $\Delta \in SoftIQC(\Psi, M)$, with Ψ and M given, (iii) $\Pi := \Psi^\sim M \Psi$ satisfying $\Pi_{22} < 0 \ \forall \omega$, and (iv) all the trajectories of the extended system start from $\mathcal{X}_0 \times \{0^{n_\psi}\}$. For some F, H defined in* (3.16)*, time interval $[t_0, T]$, local region $\mathcal{X}_l \subset \mathbb{R}^{n_G}$, set of initial conditions $\mathcal{X}_0 \subset \mathbb{R}^{n_G}$, disturbance bound R, function h, and set of uncertain parameters $\mathcal{D}$, function $q : \mathbb{R}^{n_G} \to \mathbb{R}$, and $\alpha \in \mathbb{R}$, suppose there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$, and a matrix $Y_{22} \in \mathbb{S}^{n_\psi}$ satisfying* (3.22)*, such that the following constraints hold*

$$\partial_t V(t,x) + \partial_x V(t,x) \cdot F(t,x,w,d,\delta) + z^\top M z \leq d^\top d, \quad \forall (x,t,w,d,\delta) \in \mathcal{X}_l \times \mathbb{R}^{n_\psi} \times [t_0, T] \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d} \times \mathcal{D}, \tag{3.24a}$$

$$\mathcal{X}_0 \times \{0^{n_\psi}\} \subseteq \{x \in \mathbb{R}^n : V(t_0, x) \leq 0\}, \tag{3.24b}$$

$$\left\{x_G \in \mathbb{R}^{n_G} : \mathcal{V}(T,x) \leq R^2\right\} \subseteq \Omega_\alpha^q, \forall \psi \in \mathbb{R}^{n_\psi}, \tag{3.24c}$$

$$\left\{x_G \in \mathbb{R}^{n_G} : \mathcal{V}(t,x) \leq R^2 h(t)\right\} \subseteq \mathcal{X}_l, \forall (t,\psi) \in [t_0, T] \times \mathbb{R}^{n_\psi}, \tag{3.24d}$$

where $\mathcal{V} = V - \psi^\top Y_{22} \psi$, and z is the output of the map H. Then all trajectories of $F_u(G, \Delta)$ (defined by (3.9)*–*(3.10)*) starting from $x_G(t_0) \in \mathcal{X}_0$ satisfy $x_G(T) \in \Omega_\alpha^q$. Therefore Ω_α^q is an outer bound to $FRS(T; F_u(G, \Delta), t_0, \mathcal{X}_0, R, h, \mathcal{D})$* (3.11)*.*

[*]The notation Π_{22} refers to the partitioning $\Pi = \begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^\sim & \Pi_{22} \end{bmatrix}$ conformably with the dimensions of v and w.

Proof. By assumption that $F_u(G,\Delta)$ is well-posed, the signals (x,v,w,z) generated for the extended system for the input $d \in \mathcal{L}_2^{n_d}$ are $\mathcal{L}_{2e}$ signals. By combining (3.24a) and (3.24d) we have the following dissipation inequality:

$$\partial_t V(t,x) + \partial_x V(t,x)\cdot F(t,x,w,d,\delta) + z^\top M z \le d^\top d,$$
$$\forall (x,t,w,d,\delta) \text{ s.t. } x \in \Omega^{\mathcal{V}}_{t,R^2h(t)}, t \in [t_0,T], w \in \mathbb{R}^{n_w}, d \in \mathbb{R}^{n_d}, \delta \in \mathcal{D}. \quad (3.25)$$

Since (3.25) only holds on the set $\Omega^{\mathcal{V}}_{t,R^2h(t)}$, we need to first prove that all the states starting from $\mathcal{X}_0 \times \{0^{n_\psi}\}$ won't leave $\Omega^{\mathcal{V}}_{t,R^2h(t)}$, for all $t \in [t_0,T]$. Assume there exist a time instance $T_1 \in [t_0,T]$, $x_0 \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, and signals w satisfying (3.2), $\delta(t) \in \mathcal{D}$, $w(t) \in \mathbb{R}^{n_w}$, such that a trajectory of the extended system starting from $x(t_0) = x_0$ satisfies $\mathcal{V}(T_1, x(T_1)) > R^2 h(T_1)$. Define $T_2 = \inf_{\mathcal{V}(t,x(t))>R^2h(t)} t$, and integrate (3.25) over $[t_0,T_2]$:

$$V(T_2,x(T_2)) - V(t_0,x(t_0)) + \int_{t_0}^{T_2} z(t)^\top M z(t) dt \le \int_{t_0}^{T_2} d(t)^\top d(t) dt.$$

By assumption $x_0 \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, it follows from constraint (3.24b) that $V(t_0,x(t_0)) \le 0$. Combining it with d satisfying (3.2) to show

$$V(T_2,x(T_2)) + \int_{t_0}^{T_2} z(t)^\top M z(t) dt < R^2 h(T_2). \quad (3.26)$$

It follows from Lemma 1, $\Delta \in \text{SoftIQC}(\Psi, M)$, $\Pi_{22} < 0$ $\forall \omega$ and Y_{22} satisfies (3.22) that (3.23) holds for the $\mathcal{L}_{2e}$ signals (v,w,z), and thus

$$V(T_2,x(T_2)) - \psi(T_2)^\top Y_{22} \psi(T_2) < R^2 h(T_2). \quad (3.27)$$

Thus (3.27) is a contradiction, since $V(T_2,x(T_2)) - \psi(T_2)^\top Y_{22}\psi(T_2) = \mathcal{V}(T_2,x(T_2)) = R^2h(T_2)$. Therefore there doesn't exist a $T_1 \in [t_0,T]$, such that $x(T_1) \notin \Omega^{\mathcal{V}}_{T_1,R^2h(T_1)}$. As a result, for all $x(t_0) \in \mathcal{X}_0 \times \{0^{n_\psi}\}$, we have $x(t) \in \Omega^{\mathcal{V}}_{t,R^2h(t)}$, for all $t \in [t_0,T]$, and thus $x(T) \in \Omega^{\mathcal{V}}_{T,R^2}$. Finally, it follows from (3.24c) that $x_G(T) \in \Omega^q_\alpha$. □

Remark 1. *The use of soft IQCs requires some care as they are only defined in the frequency domain for $\mathcal{L}_2$ inputs and yet the analysis must be performed using $\mathcal{L}_{2e}$ signals (to prevent circular arguments). Section 3.2 is restricted to the use of hard IQCs for which the time-domain IQC holds over finite time horizons. This removes the technical details associated with soft IQCs. This restricts the analysis to IQCs that can be parameterized so that they are hard. In Section 3.3, however, analysis conditions are derived based on soft IQCs. The issues related to soft IQCs are resolved by constructing a finite horizon lower bound valid for $\mathcal{L}_{2e}$ signals (Lemma 1). This lower bound is then incorporated in the reachability analysis in Theorem 4. The proof of Theorem 4 demonstrates that that the reachability analysis uses the lower bound* (3.23) *valid for $\mathcal{L}_{2e}$ signals (v,w,z), instead of using* (2.13)*, which requires (v,w,z) to be $\mathcal{L}_2$ signals.*

Note that the characterization of a frequency domain IQC as "soft" vs. "hard" depends on the factorization of the frequency domain multiplier. The J-spectral factorization in [32] always yields a "hard" IQC for any frequency domain multiplier (although this may not be an ideal parameterization for numerical implmentations)

By applying the generalized S-procedure [22] to (3.24), and using α as the cost function, we obtain the following SOS problem, $\mathbf{sosopt}_3(F, H, p, g, q, r_0, R, h, p_\delta, \Psi, \mathcal{M})$:

$$\begin{aligned}
\min_{\alpha,\eta,s,V,M,Y_{22},\epsilon_1,\epsilon_2} \quad & \alpha \\
\text{s.t.} \quad & V \in \mathbb{R}[(t,x)], M \in \mathcal{M} \text{ and } Y_{22} \in \mathbb{S}^{n_\psi} \text{ satisfying (3.22)}, \\
& s_5 - \epsilon_1 \in \Sigma[x], s_6 - \epsilon_2 \in \Sigma[(x,t)], \epsilon_1 > 0, \epsilon_2 > 0, \\
& s_4 \in \Sigma[x_G],\ s_7 \in \Sigma[(x,t)],\ s_i \in \Sigma[(x,d,w,t,\delta)],\ \forall i \in \{1,2,3\}, \\
& -(\partial_t V + \partial_x V \cdot F + z^\top M z - d^\top d) + (p-\eta)s_1 - s_2 g \\
& \qquad\qquad - s_3 p_\delta \in \Sigma[(x,d,w,t,\delta)], \qquad (3.28a) \\
& -V|_{t=t_0, x=[x_G; 0^{n_\psi}]} + s_4 r_0 \in \Sigma[x_G], \qquad (3.28b) \\
& -(q-\alpha)s_5 + \mathcal{V}|_{t=T} - R^2 \in \Sigma[x], \qquad (3.28c) \\
& -(p-\eta)s_6 + \mathcal{V} - R^2 h - s_7 g \in \Sigma[(x,t)]. \qquad (3.28d)
\end{aligned}$$

Compared with sosopt_2, the optimization sosopt_3 has one more decision matrix Y_{22} and an associated KYP LMI convex constraint, and it can also be solved by using Algorithm 1.

3.4 Examples

A workstation with a 2.7 [GHz] Intel Core i5 64 bit processors and 8[GB] of RAM was used for performing all computations in the following examples. The SOS optimization problem is formulated and translated into SDP using the Sum-of-Squares module in Yalmip [33] on MATLAB, and solved by the SDP solver Mosek [34]. Table 3.2 shows the degree of polynomials we chose, and the computation time it took for each example.

Table 3.2: Computation times for each example

Examples / sections	# of x	Degree of f	Degree of V	Degree of s	Time[sec]
Section 3.1	2	3	8	6	6.1×10^1
Section 3.4	4	3	6	4	1.1×10^2
Section 3.4: GTM	4	6	8	6	1.1×10^3
Section 3.4: GTM with d	4	3	8	6	3.2×10^3
Section 3.4: GTM with d, δ	4	3	8	6	5.0×10^3
Section 3.4: GTM with d, Δ	6	3	6	4	8.2×10^3
Section 3.4	7	3	6	6	3.7×10^3

The dynamics f in the following examples are all time-invariant, but since our reachability analysis is addressed in finite-time horizon, we use time-varying storage functions.

Van der Pol example

Consider the following Van der Pol oscillator dynamics in reverse time with time-invariant uncertain parameter $\delta_{TI} \in [-3, 3]$:

$$\begin{aligned} \dot{x}_1 &= x_2(1 + 0.2\delta_{TI}), \\ \dot{x}_2 &= x_1 + (x_1^2 - 1)x_2. \end{aligned}$$

In this case δ_{TI} is treated as a perturbation, where $w = \Delta(v) = \delta_{TI} v$, and $v = 0.2x_2$. As discussed in Example 3, the time invariant uncertain parameter δ_{TI} satisfies both hard and soft IQCs: $\delta_{TI} \in \text{HardIQC}(\Psi, M_D)$ and $\delta_{TI} \in \text{SoftIQC}(\Psi, M_{DG})$, where the constraints for M_D and M_{DG} are given in Example 1 and 3, respectively. The robust reachability analysis is performed using both kinds of IQCs. In both cases, we use the same filter Ψ, and choose d and m from (3.13) to be $d = 1$, $m = 4$, which correspond to Ψ described by the following dynamics:

$$\begin{aligned} A_\psi &= \begin{bmatrix} -4 & 0 \\ 0 & -4 \end{bmatrix}, \quad B_{\psi 1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad B_{\psi 2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \\ C_\psi &= \begin{bmatrix} 0, 1, 0, 0 \\ 0, 0, 0, 1 \end{bmatrix}^\top, D_{\psi 1} = \begin{bmatrix} 1, 0, 0, 0 \end{bmatrix}^\top, D_{\psi 2} = \begin{bmatrix} 0, 0, 1, 0 \end{bmatrix}^\top. \end{aligned}$$

Therefore the filter Ψ introduces two filter states $\psi \in \mathbb{R}^2$ to the extended system. Take the time horizon as $[t_0, T] = [0, 1.5]$ and the initial set as $\mathcal{X}_0 = \{(x_1, x_2) : x_1^2 + x_2^2 \leq 1\}$. Choose polynomials $q = p = 0.3150x_1^2 - 0.0976x_1x_2 + 0.0816x_2^2 - 0.0023x_1 + 0.0002x_2$. The local region $\mathcal{X}_l$ is picked as Ω_4^p. The optimal α computed using soft and hard IQCs are 1.21 and 1.60, respectively, which states the fact that the soft IQC achieves a less conservative outer bound and captures the nature of the uncertainty. In Fig 3.3, the simulation points $x(T)$ of the Van der Pol dynamics with the initial set $\mathcal{X}_0$, and with values of δ_{TI} randomly drawn from $[-3, 3]$ are shown with green dots. We can see from Fig 3.3 that the outer bound obtained using the soft IQC (shown with the black solid curve) is enclosed by the one computed using the hard IQC (shown with the purple dash-dotted curve). It also indicates that the outer bound obtained using the soft IQC is less conservative.

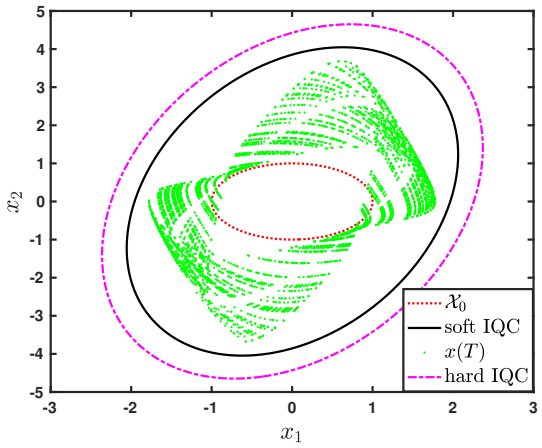

Figure 3.3: Outer bounds using soft/hard IQCs and simulation points $x(T)$ at $T = 1.5$ under uncertain parameter, with the initial condition set $\mathcal{X}_0$

NASA's Generic Transport Model (GTM) around straight and level flight condition

The GTM is a remote-controlled 5.5% scale commercial aircraft [35]. Its longitudinal model [36] is

$$\begin{aligned}
\dot{x}_1 =& \frac{1}{m}(-D - mg\sin(x_4 - x_2) + T_x\cos(x_2) + T_z\sin(x_2)),\\
\dot{x}_2 =& \frac{1}{mx_1}(-L + mg\cos(x_4 - x_2) - T_x\sin(x_2) + T_z\cos(x_2) + x_3),\\
\dot{x}_3 =& \frac{M + T_m}{I_{yy}},\\
\dot{x}_4 =& x_3,
\end{aligned} \tag{3.29}$$

where states x_1 to x_4 represent air speed (m/s), angle of attack (rad), pitch rate (rad/s) and pitch angle (rad) respectively. The control inputs are elevator deflection u_{elev} (rad) and engine throttle u_{th} (percent). The drag force D (N), lift force L (N), and aerodynamic pitching moment M (Nm) are given by $D = \bar{q}SC_D(x_2, u_{elev}, \hat{q})$, $L = \bar{q}SC_L(x_2, u_{elev}, \hat{q})$, and $M = \bar{q}S\bar{c}C_m(x_2, u_{elev}, \hat{q})$,where $\bar{q} := \frac{1}{2}\rho x_1^2$ is the dynamic pressure (N/m^2), and $\hat{q} := (\bar{c}/2x_1)x_3$ is the normalized pitch rate (unitless). C_D, C_L, and C_m are unitless aerodynamic coefficients computed from look-up tables provided by NASA.

A degree-6 polynomial model, provided in [37], is obtained after replacing all nonpolynomial terms with their polynomial approximations. The polynomial model takes the form $\dot{x} = f_6(x, u)$, where $x := [x_1, x_2, x_3, x_4]^\top$ and $u = [u_{elev}, u_{th}]^\top$. The following straight and level flight condition is computed for this model: $x_{1,t} = 45$ m/s, $x_{2,t} = 0.04924$ rad, $x_{3,t} = 0$ rad/s, $x_{4,t} = 0.04924$ rad, with $u_{elev,t} = 0.04892$ rad, and $u_{th,t} = 14.33\%$. The

subscript t denotes a trim value. A polynomial closed-loop model, denoted as $\dot{x} = f_6(x)$, is obtained by holding u_{th} at its trim value, applying a proportional pitch rate feedback $u_{elev} = K_q x_3 + u_{elev,t} = 0.0698 x_3 + u_{elev,t}$.

Analysis for the GTM

Reachability analysis is carried out on $\dot{x} = f_6(x)$ around its trim point. The set of initial conditions $\mathcal{X}_0 = \{x \in \mathbb{R}^4 : (x - x_t)^\top C^{-1}(x - x_t) - 1 \leq 0\}$ is a 4-dimensional ellipsoid inside the region of attraction, where $C = diag(20^2, (20\pi/180)^2, (50\pi/180)^2, (20\pi/180)^2)$, x_t is the trim point. Take the local region $\mathcal{X}_l = \{x \in \mathbb{R}^4 : (x - x_t)^\top C_l^{-1}(x - x_t) - 1 \leq 0\}$, where $C_l = diag(30^2, (30\pi/180)^2, (75\pi/180)^2, (30\pi/180)^2)$. Ω_1^q is chosen as the minimum volume ellipsoid containing all the simulation points at terminal time.

To improve the numerical conditioning, we define the scaled states $x_{scl} = N_{scl}^{-1} x$, where we set $N_{scl} := diag(20,\ 20\pi/180,\ 50\pi/180,\ 20\pi/180)$, since 20 m/s, $20\pi/180$ rad, $50\pi/180$ rad/s, $20\pi/180$ rad are farthest distances observed in simulation that each state can be away from their trim point value given the initial condition set $\mathcal{X}_0$. Then we have the dynamics for the scaled states: $\dot{x}_{scl} = N_{scl}^{-1} f_6(N_{scl} x_{scl})$, and this scaled dynamics is the one we will use in the SOS optimization problem. Before scaling, the coefficients of $f_6(x)$ vary from 1.6×10^{-5} to 4.5×10^{1}; after scaling, they vary from 4.5×10^{-3} to 1.8×10^{1}. Before plugging the polynomial functions r_0, q, p into the SOS optimization problem, the parameters were scaled accordingly.

Figure 3.4 and Figure 3.5 show the outer bound of reachable set in $x_2 - x_3$ space and $x_1 - x_4$ space respectively, at different simulation times. We can observe that $\Omega_{T,0}^V$ (black curve) contains all the simulation points $x(T)$ (green points) at each terminal time T.

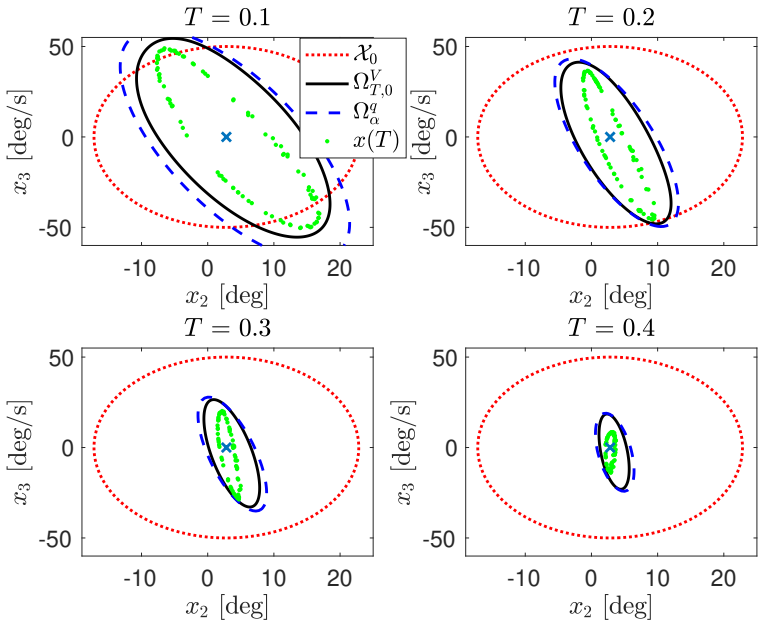

Figure 3.4: Outer bounds for GTM model in $x_2 - x_3$ plane.

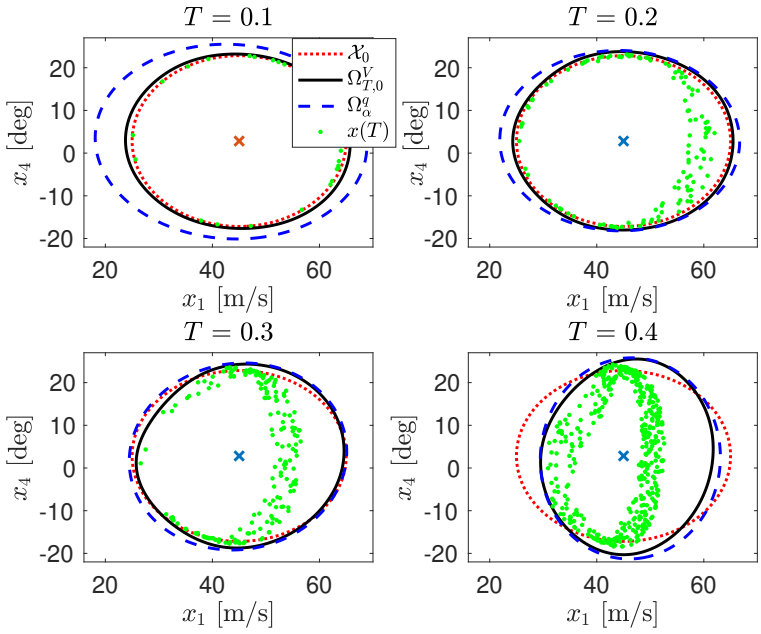

Figure 3.5: Outer bounds for GTM model in $x_1 - x_4$ plane.

GTM with $\mathcal{L}_2$ disturbance

To save computation time, reachability analysis is conducted on a 4-state degree-3 model obtained from the 4-state degree-6 model, with the same initial condition set $\mathcal{X}_0$ as that from the previous section. But an input disturbance d at the elevator channel is taken into consideration this time. The control input becomes $u_{elev} = K_q x_3 + u_{elev,t} + d = 0.0698 x_3 + u_{elev,t} + d$. Figure 3.6 shows outer bounds at time $T = 0.4$ s with disturbances of different $\mathcal{L}_2$ bounds.

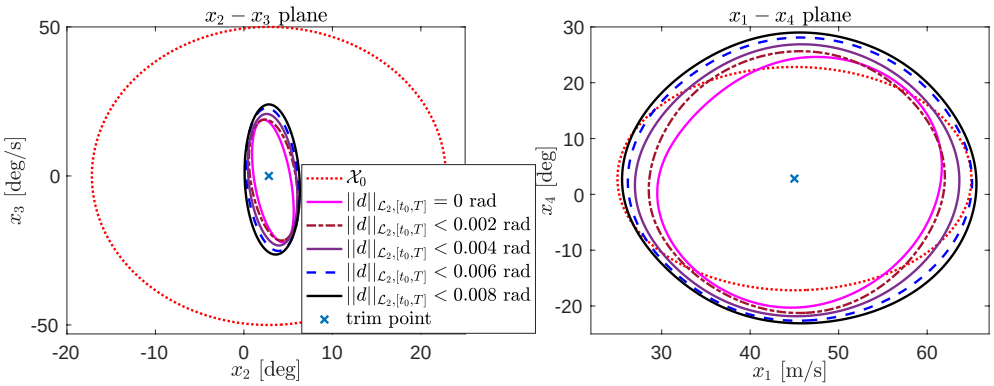

Figure 3.6: Outer bounds for GTM model at $T = 0.4$ sec with $\mathcal{L}_2$ disturbances.

GTM with $\mathcal{L}_2$ disturbances and time varying uncertain parameters

In addition to an input disturbance d at the elevator channel, satisfying $\|d\|_{\mathcal{L}_2,[t_0,T]} < 0.004$ rad, assume that the inertia I_{yy} in (3.29) is also uncertain: $I_{yy} = \gamma(t)\bar{I}_{yy}$, where $\gamma(t)$ is a time varying uncertain parameter and $\bar{I}_{yy}$ is the nominal value of inertia. Define $\delta := 1/\gamma$, assume $\gamma(t) \in [\frac{10}{11}, \frac{10}{9}]$, then $\delta(t) \in [0.9, 1.1] =: \mathcal{D}$. Equation (3.29) becomes

$$\dot{x}_3 = \frac{M + T_m}{I_{yy}} = \frac{M + T_m}{\gamma \bar{I}_{yy}} = \delta \frac{M + T_m}{\bar{I}_{yy}}.$$

The result is shown in Figure 3.7, where the outer bounds with and without uncertain parameter are shown with blue and magenta curves, respectively.

GTM with $\mathcal{L}_2$ disturbance and unmodeled dynamics Δ

Assume the control input at elevator actuator of the GTM system is corrupted by an $\mathcal{L}_2$ disturbance d, satisfying $\|d\|_{\mathcal{L}_2,[t_0,T]} < 0.004$ rad, and an LTI uncertainty Δ with $\|\Delta\|_\infty \leq \sigma$, where $\sigma > 0$. Figure 3.8 shows a block diagram for the uncertain GTM system. The input to the perturbation is $v = K_q x_3 + u_{elev,t} + d$, and the signal that actually goes into the elevator channel is $u_{elev} = v + w$, where $w = \Delta(v)$. As discussed in Example 1, this LTI uncertainty

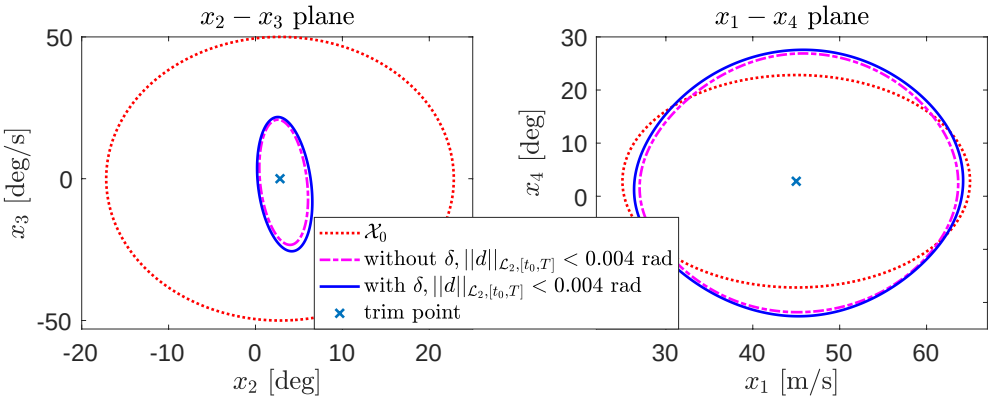

Figure 3.7: Over bounds for GTM model at $T = 0.4$ sec with $\mathcal{L}_2$ disturbance and parameter.

Δ satisfies hard IQCs defined by (Ψ, M_D) from Example 1. In this example, we choose d and m from (3.13) to be $d = 1$, $m = 1$, and they correspond to Ψ of the following dynamics:

$$A_\psi = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad B_{\psi 1} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \qquad B_{\psi 2} = \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$
$$C_\psi = \begin{bmatrix} 0,1,0,0 \\ 0,0,0,1 \end{bmatrix}^\top, D_{\psi 1} = \begin{bmatrix} 1,0,0,0 \end{bmatrix}^\top, D_{\psi 2} = \begin{bmatrix} 0,0,1,0 \end{bmatrix}^\top.$$

Again, the filter Ψ introduces two filter states $\psi \in \mathbb{R}^2$ to the extended system. We solve for the outer bounds with two values of σ using sosopt_2 with the constraint set $\mathcal{M}_1$ defined in (3.12). The results are shown in Fig 3.9, where the outer bound with $\sigma = 0.1$ is shown with the magenta curve, and the one with $\sigma = 0.4$ is shown with the blue curve.

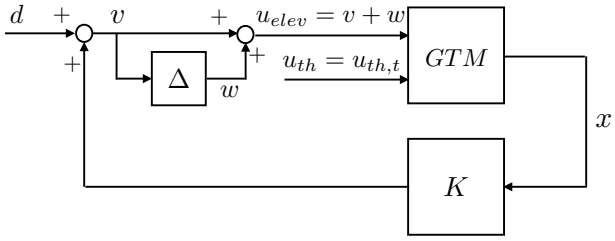

Figure 3.8: Uncertain nonlinear model for GTM.

F-18 around falling-leaf mode flight condition

In this example, we analyze a 7-state, cubic degree F-18 closed-loop polynomial model $\dot{x} = f_3(x)$ from [38], where the states $x_1, ..., x_7$ represent sideslip angle (rad), angle-of-attack (rad), roll rate (rad/s), pitch rate (rad/s), yaw rate (rad/s), bank angle (rad), controller state (rad) respectively. The trim point of the states is $x_t =$ [0 degree, 20.17 degree, -1.083

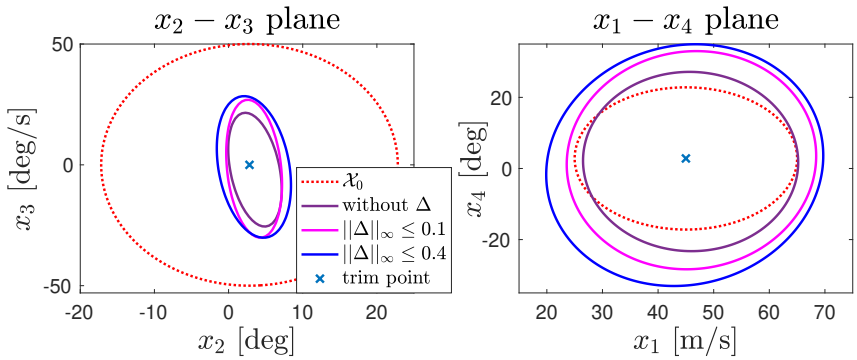

Figure 3.9: Outer bounds for GTM model at $T = 0.4$ sec with $\mathcal{L}_2$ disturbance and perturbation Δ.

degree/s, 1.855 degree/s, 2.634 degree/s, 35 degree, 0 degree]. Consider the flight condition for a coordinated turn ($x_{1,t} = 0°$) at a 35° bank angle and at the air speed of 350 ft/s. Around this condition the aircraft is more likely to experience the falling-leaf motion. The analysis is performed around this flight condition.

The given initial set $\mathcal{X}_0 = \{x \in \mathbb{R}^7 : (x - x_t)^\top C^{-1}(x - x_t) - 1 \leq 0\}$ is a 7-dimensional ellipsoid inside the region of attraction, where $C = diag((10\pi/180)^2$,$(25\pi/180)^2$, $(35\pi/180)^2$, $(30\pi/180)^2$, $(15\pi/180)^2$, $(25\pi/180)^2$, $(20\pi/180)^2)$. Again, in order to improve the numerical conditioning, we scale the states $x_{scl} = N^{-1}x$, where $N = diag(10\pi/180$, $25\pi/180$, $35\pi/180$, $5\pi/180$, $15\pi/180$, $25\pi/180$, $20\pi/180)$. Take $\mathcal{X}_l$ with radii twice as long as those of $\mathcal{X}_0$. Take Ω_1^q as the minimum volume ellipsoid containing all the simulation points $x(T)$.

Figure 3.10 and Figure 3.11 show the outer bound of reachable set in $x_1 - x_2$ space and $x_3 - x_5$ space respectively, at different simulation times. The red dotted curve is a slice of initial set $\mathcal{X}_0$. We can see that $\Omega_{T,0}^V$ (shown with the black curve) tightly contains $x(T)$ (shown with green points). We verified the reliability of the solutions of this example from SOS programming with a post-processing step as advocated in [39].

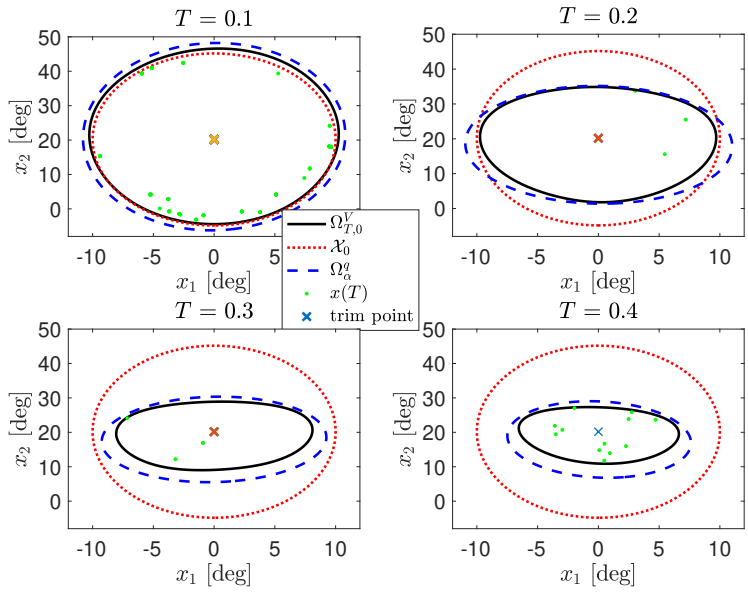

Figure 3.10: Outer bounds for F-18 model in $x_1 - x_2$ plane.

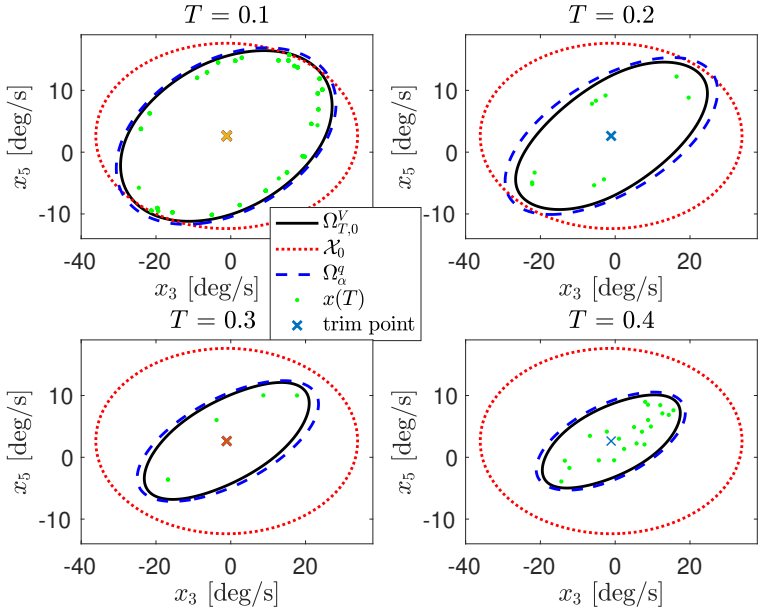

Figure 3.11: Outer bounds for F-18 model in $x_3 - x_5$ plane.

Comparison to the V, s iterations method

The outer bound of the reachable set at $T = 0.4$ sec is also computed using the V, s iterations method from [7] with the same shape function q as the one we used before. The outer-approximations obtained using the V, s iterations and quasi-convex methods are shown with the brown curves and black curves in Figure 3.12, respectively. We can see that the brown curves enclose the black curves in both plots, and thus the outer bound from the quasi-convex method is less conservative.

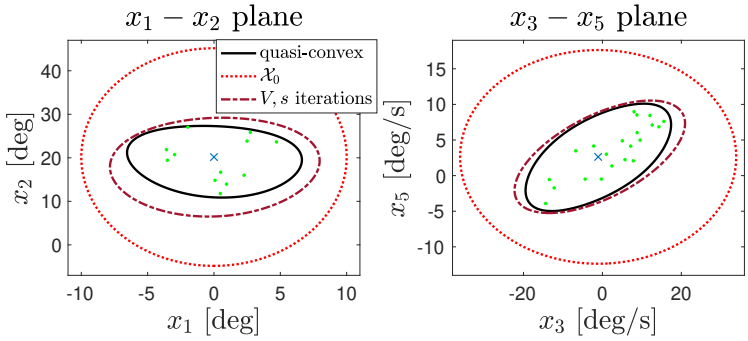

Figure 3.12: Comparison of outer bounds at $T = 0.4$ sec for F-18 model.

The computation details are shown in Table 3.3, including the obtained α and computation time. We can see that compared with the V, s iterations, within the similar amount of computation time, using the same shape function, the quasi-convex method from this dissertation is able to achieve smaller α, i.e. less conservative outer bound. Also, from the value of $\alpha^\star$ reported in Table 3.3, we can see that the outer bound obtained using our method is contained by $\Omega^q_{1.36}$, whose radii are 1.166 times those of Ω^q_1, the minimum-volume ellipsoid that contains all the simulation points. This indicates the tightness of the outer bound.

Table 3.3: Computation results and details for the two methods

Methods	α^*	Degree of V	Degree of s	Time[sec]
V, s iterations	1.70	4	4	5.2×10^3
quasi-convex	1.36	6	6	3.7×10^3

3.5 Chapter Summary

In this chapter a method for computing outer bounds of reachable sets using time varying storage functions that satisfy "local" dissipation inequalities is proposed. The method is developed for nonlinear systems with polynomial vector fields and simultaneously accounts for $\mathcal{L}_2$ disturbances, parametric uncertainties, and perturbations described by time-domain integral quadratic constraints (IQCs). A key aspect is that IQCs can be used to account

for unmodeled dynamics. The computational algorithms rely on SOS programming and the generalized S-procedure. This leads to quasi-convex optimizations for computing the tightest outer bound of the reachable set. It is thus possible to compute the global optima for this optimization and no initialization is required for the storage function. We applied the proposed method to several examples including several using nonlinear aircraft models.

Chapter 4

Robust Backward Reachability Analysis and Control Synthesis for Uncertain Nonlinear Systems

The backward reachable set (BRS) is the set of initial conditions whose successors can be driven to a target set at the end of a finite time horizon with an admissible controller. The BRS is of vital importance for safety-critical systems, since it indicates where the trajectories should start from in order to reach the target set.

Backward reachability has been extensively studied with several approaches, including occupation measure-based methods [10, 40, 41], Hamilton-Jacobi methods [8, 42, 43], relaxed Hamilton-Jacobi methods [6, 44], and Lyapunov-based methods [5]. A shortcoming of the existing reachability tools is that they rely on accurate system models. Only limited forms of uncertainty have been addressed, such as parametric uncertainty in [5, 6, 8, 10, 44] and both parametric uncertainty and $\mathcal{L}_2$ disturbances in our earlier work [45, 46].

In this chapter, we propose a method to compute inner-approximations to the BRS that are robust to a more general class of perturbations. We model the uncertain nonlinear system as an interconnection of the nominal system G and the perturbation Δ, as in Fig. 1.1. The input-output relationship of Δ is described using the integral quadratic constraint (IQC) framework [13, 18], which accounts for parametric uncertainties, and unmodeled dynamics. We characterize BRS inner-approximations by sublevel sets of storage functions that satisfy a dissipation inequality that is compatible with IQCs. We then formulate an iterative convex optimization procedure to compute storage functions and associated control laws using the generalized S-procedure [22] and sum-of-squares (SOS) techniques [21].

The proposed framework incorporates both hard and soft IQC factorizations. The use of dissipation inequalities typically requires IQCs that are valid over any finite time horizon, known as hard IQCs. However, many IQCs are specified in the frequency domain, which are equivalent to time-domain constraints over infinite horizons (soft IQCs). We obtain improved BRS bounds by incorporating soft IQCs by means of the finite-horizon bound derived in [18]. Moreover, we overcome a technical challenge that arises when the input of the perturbation

Δ depends directly on the control command, as in the case of actuator uncertainty. This dependence creates a source of nonconvexity, which we circumvent by introducing auxiliary states in the control law.

In this chapter, we first present the robust backward reachability framework using hard IQCs. Then we adapt the method to actuator uncertainties. Next, we extend the robust reachability analysis to soft IQCs. Finally, we illustrate the effectiveness of the proposed framework on a 6-state quadrotor system with actuator uncertainty.

4.1 Backward Reachability with Hard IQCs

Problem Setup

Consider an uncertain nonlinear system defined on $[0, T]$:

$$\dot{x}_G(t) = f(x_G(t), w(t), d(t)) + g(x_G(t), w(t), d(t))u(t), \tag{4.1a}$$
$$v(t) = h(x_G(t), w(t), d(t)), \tag{4.1b}$$
$$w(\cdot) = \Delta(v(\cdot)), \tag{4.1c}$$

which is an interconnection (Fig. 1.1) of the nominal system G and the perturbation Δ, denoted as $F_u(G, \Delta)$. In (4.1), $x_G(t) \in \mathbb{R}^{n_G}$ is the state, $u(t) \in U \subseteq \mathbb{R}^{n_u}$ is the control input, $d(t) \in \mathbb{R}^{n_d}$ is the external disturbance, and $v(t) \in \mathbb{R}^{n_v}$ and $w(t) \in \mathbb{R}^{n_w}$ are the inputs and outputs of Δ. The mappings f, g, and h define the nominal system G. The perturbation $\Delta : \mathcal{L}_{2e}^{n_v} \to \mathcal{L}_{2e}^{n_w}$ is a bounded and causal operator. Note that in (4.1b), v does not depend directly on u. Well-posedness of $F_u(G, \Delta)$ is defined as follows.

Definition 12. *$F_u(G, \Delta)$ is well-posed if for all $x_G(t_0) \in \mathbb{R}^{n_G}$ and $d \in \mathcal{L}_{2e}^{n_d}$ there exist unique solutions $x_G \in \mathcal{L}_{2e}^{n_G}$, $v \in \mathcal{L}_{2e}^{n_v}$, and $w \in \mathcal{L}_{2e}^{n_w}$ satisfying (4.1) with a causal dependence on d.*

Assumption 2. *(i) The disturbance d satisfies $d \in \mathcal{L}_2^{n_d}$ with:*

$$\|d\|_{\mathcal{L}_2,[0,T]} < R, \text{ for some } R > 0, \text{ and} \tag{4.2}$$

(ii) the set of control constraints is given as a polytope $U := \{u \in \mathbb{R}^{n_u} : Pu \leq b\}$, where $P \in \mathbb{R}^{n_p \times n_u}$ and $b \in \mathbb{R}^{n_p}$.

Let $x_G(t; \xi, u, d)$ define the solution to the uncertain system (4.1), at time t ($0 \leq t \leq T$), from the initial condition ξ, under the control u and the disturbance d. Let $X_T \subset \mathbb{R}^{n_G}$ denote the target set for $x_G(t; \xi, u, d)$ to reach at time T.

Definition 13. *Under Assumption 2, the BRS of $F_u(G, \Delta)$ (4.1) is defined as*

$$BRS(T, X_T, U, R, F_u(G, \Delta)) := \{\xi \in \mathbb{R}^{n_G} : \exists\ u,\ s.t.\ u(t) \in U\ \forall\ t \in [0, T],\ and \\ x_G(T; \xi, u, d) \in X_T\ \forall\ d\ with\ \|d\|_{2,[0,T]} < R\}.$$

Our goal is to compute an BRS inner-approximation and an associated controller.

Robust Backward Reachability

As discussed in the previous chapter, the perturbation Δ can represent various types of uncertainties and nonlinearities. In this section, we only consider perturbations that can be described by hard IQCs, and soft IQCs are incorporated in the framework in Section 4.3. Since the behavior of the perturbation Δ can be described by an IQC associated with a filter Ψ and a matrix M, and the analysis on $F_u(G,\Delta)$ can be instead performed on the extended system shown in Fig. 3.2, with an additional constraint $\Delta \in \text{HardIQC}(\Psi, M)$. The extended system is an interconnection of G and Ψ, with combined state vector $x := [x_G; \psi] \in \mathbb{R}^n$, $n = n_G + n_\psi$, whose dynamics can be rewritten as

$$\dot{x}(t) = F(x(t), w(t), d(t), u(t)), \tag{4.3a}$$
$$z(t) = H(x(t), w(t), d(t)), \tag{4.3b}$$

where F and H depend on the dynamics of G and Ψ. F is still affine in u. For any input $d \in \mathcal{L}_2^{n_d}$ and initial condition $x_G(t_0) \in \mathbb{R}^{n_G}$, the solutions $v \in \mathcal{L}_{2e}^{n_v}$ and $w \in \mathcal{L}_{2e}^{n_w}$ to $F_u(G,\Delta)$ (4.1) satisfy the IQC (2.12). The extended system (4.3) with the IQC (2.12) "covers" the responses of the original uncertain system $F_u(G,\Delta)$. Indeed, given any input $d \in \mathcal{L}_2^{n_d}$ and initial condition $x_G(t_0) \in \mathbb{R}^{n_G}$, the input $w \in \mathcal{L}_{2e}^{n_w}$ is implicitly constrained in the extended system so that the pair (v, w) satisfies the IQC (2.12). This set of (v, w) that satisfies the IQC (2.12) includes all input/output pairs of Δ.

We consider the memoryless, time-varying state-feedback control $u(t) = k(t, x_G(t))$, $k : \mathbb{R} \times \mathbb{R}^{n_G} \to \mathbb{R}^{n_u}$. We don't allow k to depend on ψ, since ψ is introduced by the virtual filter Ψ. The following theorem provides a BRS inner-approximation for the extended system G and Ψ, and therefore for the original uncertain system $F_u(G,\Delta)$.

Theorem 5. *Let Assumption 2 hold, and further assume (i) $F_u(G,\Delta)$ is well-posed, (ii) $\Delta \in$ HardIQC(Ψ, M), with Ψ and M given. Given $X_T \subset \mathbb{R}^{n_G}$, $P \in \mathbb{R}^{n_p \times n_u}$, $b \in \mathbb{R}^{n_p}$, $R > 0$, F, H defined in (4.3), $T > 0$, and $\gamma \in \mathbb{R}$, if there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$, and a control law $k : \mathbb{R} \times \mathbb{R}^{n_G} \to \mathbb{R}^{n_u}$ that is continuous in t and locally Lipschitz in x_G, such that*

$$\partial_t V(t,x) + \partial_x V(t,x) \cdot F(x,w,d,k) + z^\top M z \le d^\top d, \forall\ (t,x,w,d) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d},$$
$$s.t.\ V(t,x) \le \gamma + R^2, \tag{4.4}$$
$$\{x_G : V(T,x) \le \gamma + R^2\} \subseteq X_T,\ \forall\ \psi \in \mathbb{R}^{n_\psi}, \tag{4.5}$$
$$P_i k(t, x_G) \le b_i, \forall (t,x) \in [0,T] \times \mathbb{R}^n,\ s.t.\ V(t,x) \le \gamma + R^2,\ \forall\ i = 1, ..., n_p, \tag{4.6}$$

where $x = [x_G; \psi]$, then the intersection of $\Omega_{0,\gamma}^V$ with the hyperplane $\psi = 0$ is an inner-approximation to $BRS(T, X_T, U, R, F_u(G,\Delta))$ under the control law k.

Proof. Since the dissipation inequality (4.4) only holds on the region $\Omega_{t,\gamma+R^2}^V$, we first need to prove that all the state trajectories starting from $\Omega_{0,\gamma}^V$ won't leave $\Omega_{t,\gamma+R^2}^V$ for all $t \in [0,T]$. This is proven by contradiction. Assume there exists a time instance $T_1 \in [0,T]$,

such that a trajectory starting from $x(0) \in \Omega^V_{0,\gamma}$ satisfies $V(T_1, x(T_1)) > \gamma + R^2$. Define $T_2 = \inf_{V(t,x(t))>\gamma+R^2} t$, and integrate (4.4) over $[0, T_2]$:

$$V(T_2, x(T_2)) - V(0, x(0)) + \int_0^{T_2} z(t)^\top M z(t) dt \leq \int_0^{T_2} d(t)^\top d(t) dt.$$

Apply $x(0) \in \Omega^V_{0,\gamma}$ and $\Delta \in \text{HardIQC}(\Psi, M)$ to show

$$V(T_2, x(T_2)) \leq \gamma + \int_0^{T_2} d(t)^\top d(t) dt.$$

Next recall that d is assumed to satisfy (4.2):

$$\gamma + R^2 = V(T_2, x(T_2)) < \gamma + R^2,$$

which is a contradiction. As a result, $x(0) \in \Omega^V_{0,\gamma}$ implies $x(t) \in \Omega^V_{t,\gamma+R^2}$ for all $t \in [0, T]$, and thus $V(T, x(T)) \leq \gamma + R^2$. Combining it with (4.5) shows that $\Omega^V_{0,\gamma}$ is an inner-approximation to the BRS of the extended system, and the intersection of $\Omega^V_{0,\gamma}$ with $\psi = 0$ is an inner-approximation to $BRS(T, X_T, U, R, F_u(G, \Delta))$. Lastly, constraint (4.6) ensures the control signal derived from $u(t) = k(t, x_G(t))$ satisfies the control constraints $Pu(t) \leq b$, $\forall t \in [0, T]$. □

Robust Backward Reachability with SOS Programming

To find a V and a k satisfying (4.4)–(4.6), we make use of sum-of-squares (SOS) programming. To do so, we restrict the decision variables to polynomials $V \in \mathbb{R}[(t, x)]$, $k \in \mathbb{R}^{n_u}[(t, x_G)]$, and make the following assumption.

Assumption 3. *The nominal system G given in (4.1) has polynomial dynamics: $f \in \mathbb{R}^{n_G}[(x_G, w, d)]$, $g \in \mathbb{R}^{n_G \times n_u}[(x_G, w, d)]$, and $h \in \mathbb{R}^{n_v}[(x_G, w, d)]$. Therefore, F and H in (4.3) are polynomials. X_T is a semi-algebraic set: $X_T := \{x_G : p_x(x_G) \leq 0\}$, where $p_x \in \mathbb{R}[x_G]$ is provided.*

In Example 1 and 2, we have seen that for each type of perturbation, any IQC defined by a properly chosen Ψ and a M drawn from the constraint set $\mathcal{M}$ is valid. Therefore, along with V and k, we also treat $M \in \mathcal{M}$ as a decision variable. Assume $\mathcal{M}$ is described by linear matrix inequalities [18]. Define $p_t := t(T - t)$, which is nonnegative for all $t \in [0, T]$. By applying the generalized S-procedure [22] to (4.4)–(4.6), and choosing the volume of $\Omega^V_{0,\gamma}$ as the objective function (to be maximized), we obtain the following optimization:

$$\begin{aligned}
\sup_{V,M,k,s_i} \ & \text{Volume}(\Omega^V_{0,\gamma}) \\
\text{s.t.} \ & V \in \mathbb{R}[(t, x)], k \in \mathbb{R}^{n_u}[(t, x_G)], M \in \mathcal{M}, \\
& -(\partial_t V + \partial_x V \cdot F|_{u=k} + z^\top M z - d^\top d) - s_1 p_t \\
& \quad + (V - \gamma - R^2) s_2 \in \Sigma[(t, x, w, d)], && \text{(4.7a)} \\
& -s_3 p_x + V|_{t=T} - \gamma - R^2 \in \Sigma[x], && \text{(4.7b)} \\
& -(P_i k - b_i) - s_{4,i} p_t + (V - \gamma - R^2) s_{5,i} \in \Sigma[(t, x)], \ \forall\, i = 1, ..., n_p, && \text{(4.7c)}
\end{aligned}$$

where $s_1, s_2 \in \Sigma[(t,x,w,d)]$, $(s_3 - \epsilon) \in \Sigma[x]$, and $s_{4,i}, s_{5,i} \in \Sigma[(t,x)]$. The positive number ϵ ensures that s_3 is uniformly bounded away from 0. The optimization (4.7) is nonconvex, since it is bilinear in two sets of decision variables, V and $(k, s_2, s_{5,i})$. As summarized in Algorithm 2, the nonconvex optimization (4.7) is handled by alternating the search over these two sets of decision variables, since holding one set fixed and optimizing over the other results in a convex problem. Since an explicit expression is not available for the volume of $\Omega^V_{0,\gamma}$ for a generic V, we instead enlarge it by maximizing γ in the (k,γ)-step when V is fixed. Combining it with the constraint (4.8) in the V-step, which enforces $\Omega^{V^{j-1}}_{0,\gamma^j} \subseteq \Omega^{V^j}_{0,\gamma^j}$, we are able to prove that the volume of the inner-approximations will not decrease with each iteration [46, Theorem 2]. A linear state feedback for the linearization about the equilibrium point was used to compute the initial iterate V^0.

Algorithm 2 Alternating direction method for hard IQCs

Input: function V^0 such that constraints (4.7) are feasible by proper choice of s_i, k, γ, M.
Output: k, γ, V, M.

1: **for** $j = 1 : N_{\text{iter}}$ **do**
2: **(k,γ)-step**: decision variables (s_i, k, γ, M). Maximize γ subject to (4.7) using $V = V^{j-1}$. This yields $(s_2^j, s_{5,i}^j, k^j)$ and optimal reward γ^j.
3: **V-step**: decision variables $(s_0, s_1, s_3, s_{4,i}, V, M)$. Maximize the feasibility subject to (4.7) as well as $s_0 \in \Sigma[x]$, and

$$(\gamma^j - V|_{t=0}) + (V^{j-1}|_{t=0} - \gamma^j)s_0 \in \Sigma[x], \tag{4.8}$$

using $(\gamma = \gamma^j, s_2 = s_2^j, s_{5,i} = s_{5,i}^j, k = k^j)$. This yields V^j.
4: **end for**

4.2 Extension to Actuator Uncertainty

This section considers the case where the control inputs are subject to actuator uncertainty. For example, unmodeled actuator dynamics can be modeled by a plant input u_{pert} given as the sum of the controller command u and a norm-bounded nonlinearity Δ: $u_{\text{pert}} = u + \Delta(u)$. The input v to Δ and the IQC filter output z were previously defined (Equations (4.1b) and (4.3b)) to be independent of the control command u. However, the inclusion of the actuator uncertainty implies that v and z must now depend on u.

This motivates the following generalization of the proposed method. Assume the entire input vector u is subject to the actuator uncertainty. The perturbation input and IQC filter output are now given by the following modifications to Equations (4.1b) and (4.3b):

$$v(t) = h(x_G(t), w(t), d(t), u(t)), \tag{4.9}$$
$$z(t) = H(x(t), w(t), d(t), u(t)). \tag{4.10}$$

A consequence of this generalization is that optimization over k is nonconvex even when V is fixed, since $z^\top Mz$ in (4.4) depends nonlinearly on k. A remedy is to introduce auxiliary state $\tilde{x} \in \mathbb{R}^{n_u}$ for the perturbed control input u, and to design a dynamic controller of the form

$$\dot{\tilde{x}}(t) = \tilde{k}(t, x_G(t), \tilde{x}(t)), \tag{4.11a}$$
$$u(t) = \tilde{x}(t). \tag{4.11b}$$

where $\tilde{k} : \mathbb{R} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_u}$ is to be determined. If we restrict the initial condition of $\tilde{x}$ to be zero: $\tilde{x}(0) = 0^{n_u}$, allow $\tilde{k}$ to depend on $\tilde{x}$, but not on ψ, and V to depend on the new state $\tilde{x}$: $V : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_u} \to \mathbb{R}$, then the dissipation inequality becomes:

$$\begin{aligned}\partial_t V(t,x,\tilde{x}) + \partial_x V(t,x,\tilde{x}) \cdot F(x,w,d,\tilde{x}) + \partial_{\tilde{x}} V(t,x,\tilde{x}) \cdot \tilde{k}(t,x_G,\tilde{x}) + z^\top Mz \le d^\top d,\\ \forall\ (t,x,\tilde{x},w,d) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d}, \text{s.t. } V(t,x,\tilde{x}) \le \gamma + R^2.\end{aligned} \tag{4.12}$$

The term $z^\top Mz$ in (4.12) is then nonlinear in the state variable $\tilde{x}$, rather than in the control law. The dissipation inequality is therefore bilinear in V and $\tilde{k}$, and can be solved in a way similar to Algorithm 2. Next, we provide the theorem that incorporates actuator uncertainties.

Theorem 6. *Let Assumption 2 hold, and further assume (i) $F_u(G,\Delta)$ is well-posed, (ii) $\Delta \in$ HardIQC(Ψ, M), with Ψ and M given. Given $X_T \subset \mathbb{R}^{n_G}$, $P \in \mathbb{R}^{n_p \times n_u}$, $b \in \mathbb{R}^{n_p}$, $R > 0$, F defined in* (4.3a)*, H defined in* (4.10)*, $T > 0$, and $\gamma \in \mathbb{R}$, if there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_u} \to \mathbb{R}$, and control law $\tilde{k} : \mathbb{R} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_u}$, such that* (4.12)*,*

$$\{x_G : V(T,x,\tilde{x}) \le \gamma + R^2\} \subseteq X_T, \forall\ (\psi,\tilde{x}) \in \mathbb{R}^{n_\psi} \times U, \tag{4.13}$$
$$P_i \tilde{x} \le b_i, \forall (t,x,\tilde{x}) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n_u}, \text{s.t. } V(t,x,\tilde{x}) \le \gamma + R^2, \forall\ i = 1, ..., n_p, \tag{4.14}$$

where $x = [x_G; \psi]$, then the intersection of $\Omega^V_{0,\gamma}$ with the hyperplane $(\psi, \tilde{x}) = 0$ is an inner-approximation to $BRS(T, X_T, U, R, F_u(G,\Delta))$ under the control (4.11)*.*

The conditions of Theorem 6 can be formulated as an SOS optimization similar to (4.7), and is omitted. Although we assumed all control inputs are perturbed by uncertainty, the results can be extended to the case where a subset of the actuators are perturbed. This extension involves mainly notational changes and is also omitted.

4.3 Backward Reachability with soft IQCs

Previously we assumed $\Delta \in$ HardIQC(Ψ, M). However, many IQCs are specified in the frequency domain [13]. Their time domain representation results in so called 'soft IQC'. The definitions for frequency domain and time domain soft IQCs are given in Definition 2 and 4.

Let $\Delta \in$ FreqIQC(Π) and $\Delta \in$ SoftIQC(Ψ, M) indicate that Δ satisfies corresponding frequency domain and time domain soft IQCs, respectively. As discussed in the previous

chapter, the library of IQCs specified in frequency domain can always be translated into soft IQCs, but not into hard IQCs, and when both hard and soft factorizations exist, the latter is usually less restrictive. Therefore, it is helpful to incorporate soft IQCs in the analysis.

Since soft IQCs hold over the infinite horizon, they cannot be incorporated in the analysis based on a finite-horizon dissipation inequality directly. To alleviate this issue, we use the results from Lemma 1 again, which provides lower bounds for soft IQCs over all finite horizons, and thus allows for soft IQCs in the finite horizon reachability analysis.

Based on this lemma, the following theorem provides a BRS inner-approximation for $F_u(G,\Delta)$ with $\Delta \in \text{SoftIQC}(\Psi, M)$, also allowing for actuator uncertainties.

Theorem 7. *Let Assumption 2 hold, and further assume (i) $F_u(G,\Delta)$ is well-posed, (ii) $\Delta \in$ SoftIQC(Ψ, M), with Ψ and M given, (iii) $\Pi = \Psi^* M \Psi$ satisfying $\Pi_{22} < 0$ $\forall \omega$.* Given $X_T \subset \mathbb{R}^{n_G}$, $P \in \mathbb{R}^{n_p \times n_u}$, $b \in \mathbb{R}^{n_p}$, $R > 0$, F defined in (4.3a), H defined in (4.10), $T > 0$, and $\gamma \in \mathbb{R}$, if there exists a $\mathcal{C}^1$ function $V : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^{n_u} \to \mathbb{R}$, a matrix $Y_{22} \in \mathbb{S}^{n_\psi}$ satisfying (3.22), and a $\tilde{k} : \mathbb{R} \times \mathbb{R}^{n_G} \times \mathbb{R}^{n_u} \to \mathbb{R}^{n_u}$, such that*

$$\partial_t V(t,x,\tilde{x}) + \partial_x V(t,x,\tilde{x}) \cdot F(x,w,d,\tilde{x}) + \partial_{\tilde{x}} V(t,x,\tilde{x}) \cdot \tilde{k}(t,x_G,\tilde{x}) + z^\top M z \le d^\top d,$$
$$\forall\ (t,x,\tilde{x},w,d) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d}, s.t.\ \mathcal{V}(t,x,\tilde{x}) \le \gamma + R^2, \tag{4.15a}$$
$$\{x_G : \mathcal{V}(T,x,\tilde{x}) \le \gamma + R^2\} \subseteq X_T, \forall\ (\psi,\tilde{x}) \in \mathbb{R}^{n_\psi} \times U, \tag{4.15b}$$
$$P_i \tilde{x} \le b_i, \forall (t,x,\tilde{x}) \in [0,T] \times \mathbb{R}^n \times \mathbb{R}^{n_u}, s.t.\ \mathcal{V}(t,x,\tilde{x}) \le \gamma + R^2, \forall\ i = 1,...,n_p, \tag{4.15c}$$

where $\mathcal{V} = V - \psi^\top Y_{22} \psi$, $x = [x_G; \psi]$, then the intersection of $\Omega^V_{0,\gamma}$ with the hyperplane $(\psi, \tilde{x}) = 0$ is an inner-approximation to $BRS(T, X_T, U, R, F_u(G,\Delta))$ under (4.11).

Proof. Similar to the proof of Theorem 5, it follows by contradiction that $(x(0), \tilde{x}(0)) \in \Omega^V_{0,\gamma}$ implies $(x(t), \tilde{x}(t)) \in \Omega^{\mathcal{V}}_{t,\gamma+R^2}$, for all $t \in [0,T]$. Therefore, we are able to integrate (4.15a) over $[0,T]$:

$$V(T,x(T),\tilde{x}(T)) - V(0,x(0),\tilde{x}(0)) + \int_0^T z(t)^\top M z(t) dt \le \int_0^T d(t)^\top d(t) dt.$$

Use $(x(0),\tilde{x}(0)) \in \Omega^V_{0,\gamma}$ and $\|d\|_{\mathcal{L}_2,[0,T]} < R$ to show

$$V(T,x(T),\tilde{x}(T)) + \int_0^T z(t)^\top M z(t) dt < \gamma + R^2. \tag{4.16}$$

Next it follows from $\Delta \in \text{SoftIQC}(\Psi, M)$ and Lemma 1 that

$$V(T,x(T),\tilde{x}(T)) - \psi(T)^\top Y_{22} \psi(T) < \gamma + R^2. \tag{4.17}$$

Combining (4.17) with (4.15b), it holds $x_G(T) \in X_T$ for all $(x(0),\tilde{x}(0)) \in \Omega^V_{0,\gamma}$. Therefore, the intersection of $\Omega^V_{0,\gamma}$ with $(\psi,\tilde{x}) = 0$ is an inner-approximation to $BRS(T, X_T, U, R, F_u(G,\Delta))$. □

*The notation Π_{22} refers to the partitioning $\Pi = \left[\begin{smallmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{12}^* & \Pi_{22} \end{smallmatrix}\right]$ conformably with the dimensions of v and w.

Similar to (4.7), we formulate an SOS optimization using the constraints of Theorem 7 :

$$\begin{aligned}
\sup_{V,M,Y_{22},\tilde{k},s_i} \quad & \text{Volume}(\Omega^V_{0,\gamma}) \\
\text{s.t.} \quad & V \in \mathbb{R}[(t,x,\tilde{x})], \tilde{k} \in \mathbb{R}^{n_u}[(t,x_G,\tilde{x})], \\
& M \in \mathcal{M} \text{ and } Y_{22} \in \mathbb{S}^{n_\psi} \text{ satisfy (3.22)}, \\
& -(\partial_t V + \partial_x V \cdot F|_{u=\tilde{x}} + \partial_{\tilde{x}} V \cdot \tilde{k} \\
& \quad + z^\top M z - d^\top d) + (\mathcal{V} - \gamma - R^2)s_2 - s_1 p_t \in \Sigma[(t,x,\tilde{x},w,d)], && \text{(4.18a)} \\
& -s_3 p_x + \mathcal{V}|_{t=T} - \gamma - R^2 + \Sigma_{i=1}^{n_p}(P_i\tilde{x} - b_i)s_{6,i} \in \Sigma[(x,\tilde{x})], && \text{(4.18b)} \\
& -(P_i\tilde{x} - b_i) + (\mathcal{V} - \gamma - R^2)s_{5,i} - s_{4,i}p_t \in \Sigma[(t,x,\tilde{x})], \forall\ i = 1,...,n_p, && \text{(4.18c)}
\end{aligned}$$

where $s_1, s_2 \in \Sigma[(t,x,\tilde{x},w,d)]$, $(s_3 - \epsilon), s_{6,i} \in \Sigma[(x,\tilde{x})]$ and $s_{4,i}, s_{5,i} \in \Sigma[(t,x,\tilde{x})]$. The optimization (4.18) is bilinear in (V, Y_{22}) and $(s_2, s_{5,i}, \tilde{k})$. Similar to Algorithm 2, Algorithm 3 tackles (4.18) by decomposing it into convex subproblems, and it also guarantees the improvement of the quality of the inner-approximation through iterations. $Y_{22}^0 = 0^{n_\psi}$ and a $M^0 \in \mathcal{M}$ can be used as initializations.

Algorithm 3 Alternating direction method for soft IQCs

Input: V^0, M^0 and Y_{22}^0 such that constraints (4.18) are feasible by proper choice of $s_i, \tilde{k}, \gamma$.
Output: $\tilde{k}$, γ, V, M, Y_{22}.

1: **for** $j = 1 : N_{\text{iter}}$ **do**
2: **$(\boldsymbol{k}, \boldsymbol{\gamma})$-step**: decision variables $(s_i, \tilde{k}, \gamma)$. Maximize γ subject to (4.18) using $V = V^{j-1}$, $M = M^{j-1}$, and $Y_{22} = Y_{22}^{j-1}$. This yields $(s_2^j, s_{5,i}^j, \tilde{k}^j, \gamma^j)$.
3: **$\boldsymbol{V}$-step**: $(s_0, s_1, s_3, s_{4,i}, s_{6,i}, V, M, Y_{22})$ are decision variables. Maximize the feasibility subject to (4.18) as well as $s_0 \in \Sigma[(x,\tilde{x})]$, and

$$(\gamma^j - V|_{t=0}) + (V^{j-1}|_{t=0} - \gamma^j)s_0 \in \Sigma[(x,\tilde{x})],$$

using $\gamma = \gamma^j, s_2 = s_2^j, s_{5,i} = s_{5,i}^j, \tilde{k} = \tilde{k}^j$. This yields V^j, M^j and Y_{22}^j.
4: **end for**

Computational complexity: A polynomial decision variable of degree $2d_{\text{poly}}$ in n_{var} variables has a $m \times m$ Gram matrix representation where $m := \binom{n_{\text{var}}+d_{\text{poly}}}{d_{\text{poly}}}$, and thus introduces $\mathcal{O}(m^2)$ decision variables due to the Gram matrix. Using higher-degree polynomial decision variables can provide a less conservative BRS estimate, but it takes longer to solve and might be intractable for high-dimensional systems.

4.4 Numerical Examples

In the following examples, the SOS optimization problem is formulated using the SOS module in SOSOPT [47] on MATLAB, and solved by the SDP solver MOSEK [34].

Generic Transport Model (GTM) Example

The GTM is a remote-controlled 5.5% scale commercial aircraft [35]. The longitudinal dynamics are approximated by a cubic degree polynomial model provided in [37]:

$$\dot{x}_1 = -1.492x_1^3 + 4.239x_1^2 + 0.003x_1x_2 + 0.006x_2^2 - 3.236x_1 + 0.923x_2 + (0.240x_1 - 0.317)u,$$
$$\dot{x}_2 = -7.228x_1^3 + 1.103x_2^3 + 18.365x_1^2 - 45.339x_1 - 4.373x_2 + (41.505x_1 - 59.989)u,$$

where $x_G = [x_1; x_2]$ is the state, x_1 is the angle of attack (rad), x_2 is the pitch rate (rad/s), and the control input u is the elevator defection (rad). Assume the control input u generated by the controller C is corrupted by an additive uncertainty Δ exerted on the actuator, as shown in Fig. 4.1. The actual signal that goes into the elevator channel is $u_{elev} := u + w$, where w is the output of Δ.

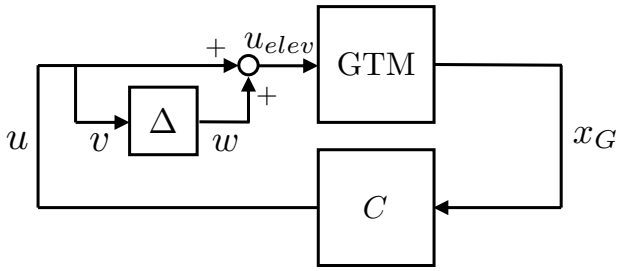

Figure 4.1: The diagram of the GTM with input perturbation

Sector IQCs

Assume that Δ lies within the sector $[\alpha, \beta]$, where $\alpha = 0$, and $\beta = 0.2$. The filter Ψ and constraint set $\mathcal{M}$ given below define a hard IQC:

$$\Psi = I_2,\ \mathcal{M} = \left\{ \lambda \begin{bmatrix} -2\alpha\beta & \alpha+\beta \\ \alpha+\beta & -2 \end{bmatrix} : \lambda \in \Sigma[(x_G, \tilde{x}, w)] \right\},$$

where λ is a polynomial decision variable, which introduces more freedom to the optimization.

Take the target set as $X_T = \{x_G : x_G^\top x_G \leq (\pi/27)^2\}$ (shown in Fig. 4.2 with red solid curve), and assume the actuator limit on u is $|u(t)| \leq 0.261$ rad. Degree-4 polynomial storage functions are used to compute two inner-approximations on time horizons [0, 1 sec] and [0, 2 sec], which correspond to the blue dashed curve and black dotted curve in Fig. 4.2, respectively. Solid curves with crosses represent simulation trajectories starting from the inner-approximation with time horizon [0, 2 sec] in the presence of actuator uncertainty, and crosses represent different initial conditions. In Fig. 4.3, the simulations of control inputs for different initial conditions are shown. We note that they are all within the control limits during the time horizon.

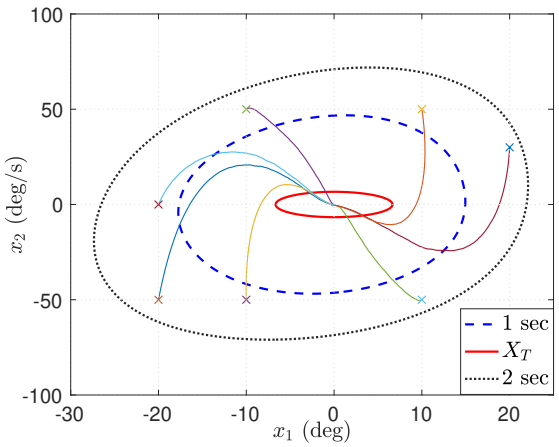

Figure 4.2: Simulation trajectories, and inner-approximations of the GTM example with the sector IQC for two time horizons.

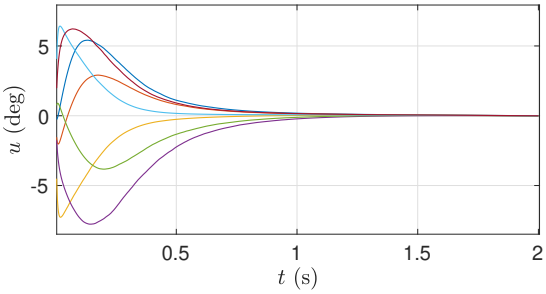

Figure 4.3: Simulations of control inputs

Hard and soft IQCs

This time we assume that the perturbation Δ in Fig. 4.1 is a time invariant parametric uncertainty: $w(t) = \Delta(v(t)) = \delta v(t)$, with $\delta \in \mathbb{R}$, $|\delta| \leq 0.2$. Therefore, the actual signal that goes into the elevator channel is $u_{elev} = u+w = (1+\delta)u$. As discussed in Example 3, δ satisfies both HardIQC(Ψ, M_D) and SoftIQC(Ψ, M_{DG}). The backward reachability is performed using both kinds of IQCs. In both cases, we use the same filter Ψ, and choose $\Psi_{11}^{d,m}$ from (3.13) with $m = 10$ and $d = 1$. Therefore, Ψ introduces two filters states $\psi \in \mathbb{R}^2$ to the extended system. Take the time horizon as [0, 2 sec], and use the same target set and actuator limits from the previous example.

In Fig. 4.4, the inner-approximations computed using the hard and soft IQCs are shown with the dashed purple curve, and the dash-dotted black curve. We see that with soft IQC we are able to certify a larger inner-approximation. This is because the soft IQC has richer knowledge of the time invariant parametric uncertainty than the hard IQC.

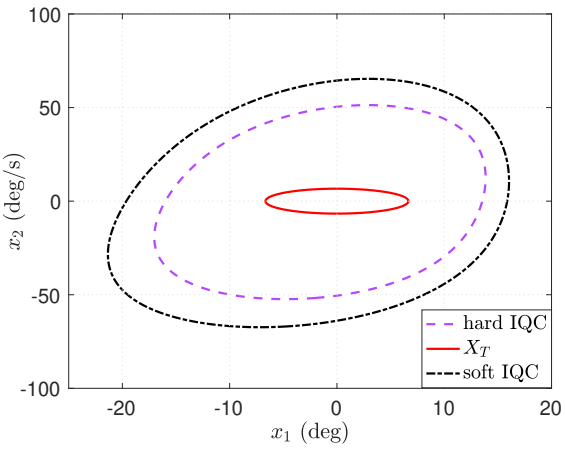

Figure 4.4: Inner-approximations with soft and hard IQCs

Quadrotor Example

Consider the 6-state planar quadrotor dynamics [48]:

$$\begin{aligned}
\dot{x}_1 &= x_3, \\
\dot{x}_2 &= x_4, \\
\dot{x}_3 &= u_1 K \sin(x_5), \\
\dot{x}_4 &= u_1 K \cos(x_5) - g_n, \\
\dot{x}_5 &= x_6, \\
\dot{x}_6 &= -d_0 x_5 - d_1 x_6 + n_0 u_2,
\end{aligned}$$

where $x_G = [x_1, ..., x_6]$ is the state, x_1 to x_6 are horizontal position (m), vertical position (m), horizontal velocity (m/s), vertical velocity (m/s), roll (rad), and roll velocity (rad/s), respectively. u_1 and u_2 are total thrust and desired roll angle. Control saturation limits are $u_1(t) \in [-1.5, 1.5] + g_n/K$, and $u_2(t) \in [-\pi/12, \pi/12]$. $g_n = 9.8$, $K = 0.89/1.4$, $d_0 = 70$, $d_1 = 17$, and $n_0 = 55$ are taken from [48].

The control objective of this example is to design controllers for u_1 and u_2 to maintain the trajectories of the quadrotor starting from the BRS to stay within the safe set X_t during the time horizon $[0, T]$ with $T = 2$. X_t is given as $X_t = \{x_G : x_G^\top N x_G \leq 1\}$, where $N = diag(1/1.7^2, 1/0.85^2, 1/0.8^2, 1/1^2, 1/(\pi/12)^2, 1/(\pi/2)^2)$. $\sin(x_5)$ is approximated by $(-0.166x_5^3 + x_5)$ and $\cos(x_5)$ is approximated by $(-0.498x_5^2 + 1)$, using least squares regression for $x_5 \in [-\pi/12, \pi/12]$. The validity of this bound on x_5 is guaranteed by the state constraint X_t. Assume that the control input u_2 is perturbed by an additive norm-bounded nonlinearity $\|\Delta\|_{\mathcal{L}_2 \to \mathcal{L}_2, [0,T]} \leq 0.2$, which introduces one auxiliary state $\tilde{x}$ to the analysis. We use the hard IQC discussed in Example 2 with a fixed filter Ψ and search for M over the constraint set given in (3.14). The computation of BRS inner-approximations takes 1.1×10^3 and 3.6×10^4 seconds using degree-2 and degree-4 polynomial storage functions.

Fig. 4.5 shows the projections of the resulting inner-approximations. The one computed using degree-2 storage function is shown with the solid magenta curve, and the one computed using degree-4 storage function is shown with the red dash-dotted curve. The projections of X_t are shown with the blue solid curves.

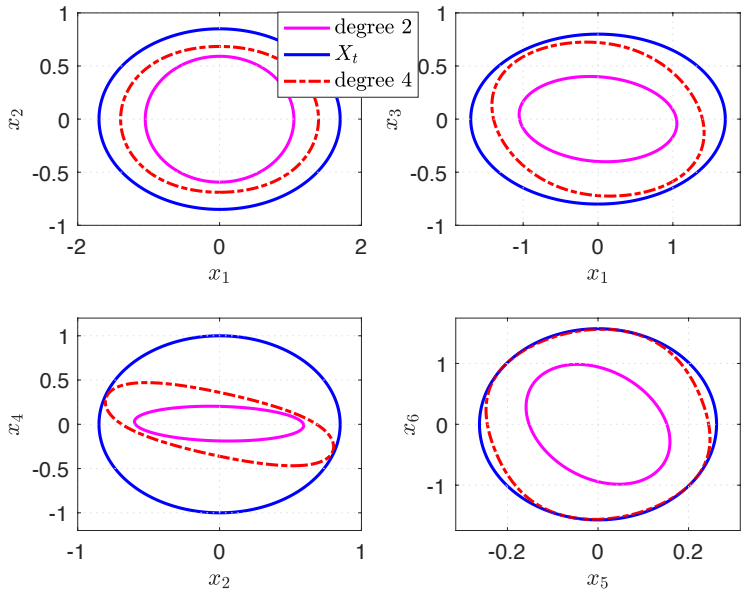

Figure 4.5: BRS inner-approximations for the quadrotor

4.5 Chapter Summary

In this chapter, a method for computing robust inner-approximations to the BRS and robust control laws is proposed for uncertain nonlinear systems, modeled as an interconnection of the nominal system G and the perturbation Δ. The proposed framework merges dissipation inequalities and IQCs, with both hard and soft factorizations. The use of IQCs enabled us to address a large class of perturbations, including unmodeled dynamics. The generalized S-procedure and sum-of-squares programming are used to derive computational algorithms. We applied the proposed method to uncertain nonlinear systems, including a 6-state quadrotor examples with actuator uncertainties.

Chapter 5

Safe by Design Motion Planning and Control for Nonlinear Systems

An important task for cyber-physical systems is to design controllers that perform complex tasks while providing safety guarantees. Formal methods approaches [49] allow one to specify desired behaviors and then compute control strategies that achieve them, but are often limited to systems with small state dimension. Thus, it is beneficial to represent the system of interest with a lower dimensional model that is amenable to such approaches, and to design controllers based on that model. However, applying such controllers directly to the original system might lose the safety guarantees established on its lower dimensional model.

This chapter tackles this issue, and presents a hierarchical trajectory planning and control framework for nonlinear dynamical systems. In this framework, a low-fidelity model (e.g., reduced order model, or linearized model) is used to plan feasible trajectories satisfying the planning constraints, and a high-fidelity model is used for designing a tracking controller to track the planned trajectories. With the tracking controller, the error state between the low- and high-fidelity models is guaranteed to be bounded, and this tracking error bound can be then used for redesigning the planning constraints to "robustify" the planning algorithm.

The hierarchical trajectory planning and control framework has also been explored by the trajectory planning community [7, 50–52]. In [50], a Hamilton-Jacobi equation based method is presented to synthesize a tracking error bound and an optimal tracking controller. An extension in [51] shows how this may also be done with SOS optimization. A robust forward reachable set used for planning is computed in [52] by considering the difference between the planning and tracking models. Compared with these works, in this chapter, the error state between low- and high-fidelity models explicitly depends on the control input of the low-fidelity model, which offers more freedom in the choice of the low-fidelity model and provides lower error bounds than when only the states of both models are considered.

In this chapter, we first describe the problem setup, and the hierarchical control framework. Then we present a method for computing tracking controllers, and their associated tracking error bounds. Next, we extend the method to the error state that depends on the planner's control input. Finally, we demonstrate the method on a ship docking example.

5.1 Problem setup

In this chapter we describe our hierarchical approach to safe-by-design trajectory planning and control which consists of two layers: a planning layer, which uses a low-fidelity system model, and a tracking layer, with a high-fidelity system model. For example, the low fidelity model might be a model with a lower state dimension than the high fidelity model to reduce the computational burden of planning, or a linearized model of the high fidelity model. By analyzing the dynamics of the error between these two systems' states, we will show how we can bound this error by synthesizing an appropriate tracking controller. This controller and the corresponding bound for the error are designed via sum-of-squares (SOS) programming.

High-Fidelity System Model

The high fidelity model used is of the form:

$$\dot{x}(t) = f(x(t), w(t)) + g(x(t), w(t)) \cdot u(t), \tag{5.1}$$

with state $x(t) \in X \subseteq \mathbb{R}^{n_x}$, disturbance $w(t)$ bounded in a set W, i.e., $w(t) \in W \subseteq \mathbb{R}^{n_w}$, and bounded control $u(t) \in U \subseteq \mathbb{R}^{n_u}$, $f : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$, and $g : \mathbb{R}^{n_x} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$. The sets X and U are the constraint sets imposed on the states and control inputs in high-fidelity model, respectively.

Low-Fidelity Planning Model

The low-fidelity model, which is a simplified version of (5.1), is of the form:

$$\dot{\hat{x}}(t) = \hat{f}(\hat{x}(t), \hat{u}(t)), \tag{5.2}$$

where $\hat{x}(t) \in \hat{X} \subseteq \mathbb{R}^{\hat{n}_x}$, $\hat{u}(t) \in \hat{U} \subseteq \mathbb{R}^{\hat{n}_u}$, and $\hat{f} : \mathbb{R}^{\hat{n}_x \times \hat{n}_u} \to \mathbb{R}^{\hat{n}_x}$. The sets $\hat{X}$ and $\hat{U}$ are constraint sets enforced by the planning layer. The control input for the low-fidelity model, computed via the planning algorithm of choice, is assumed to be a zero-order hold signal with sampling time $T_s > 0$. This means:

$$\hat{u}(t) = \hat{u}(\tau_k), \ \forall t \in [\tau_k, \tau_{k+1}), \text{ with } \tau_k = k \cdot T_s, \tag{5.3a}$$

$$\hat{u}(\tau_{k+1}) = \hat{u}(\tau_k) + \Delta\hat{u}(\tau_{k+1}), \tag{5.3b}$$

where T_s is the sampling time, $\Delta\hat{u}(t)$ is the periodic change in the control, restricted to a set $\Delta\hat{U} \subseteq \mathbb{R}^{\hat{n}_u}$. We note that the zero-order hold behavior of the input will become important in Section 5.2, where it will necessitate additional analysis in order to provide a tracking error bound.

In order to use a planning algorithm such as model-predictive control (MPC) we require a discrete-time version of our low-fidelity model. To this end, we discretize (5.2) using one step forward Euler integration, resulting in:

$$\hat{x}(\tau_{k+1}) = \hat{x}(\tau_k) + T_s \cdot \hat{f}(\hat{x}(\tau_k), \hat{u}(\tau_k)). \tag{5.4}$$

Remark 2. *Note that our planner-tracker synthesis framework is not restricted to any specific planning algorithm. The framework is applicable to any planning algorithm that is able to bound $\hat{x}(t), \hat{u}(t)$, and $\Delta\hat{u}(t)$. For example, our framework has been applied to different planning algorithms, including NMPC in [53, 54], signal temporal logic (STL) in [55], and discrete abstraction in [56].*

Error Dynamics

The goal is to design a controller for system (5.1) to track a reference trajectory planned using its approximation (5.2). In order to do so, we proceed by deriving the evolution of the error between (5.1) and (5.2).

Since $\hat{n}_x \leq n_x$ in general, we define a map $\pi : \mathbb{R}^{\hat{n}_x} \to \mathbb{R}^{n_x}$ and define the tracking error as:

$$e(t) = x(t) - \pi(\hat{x}(t)). \tag{5.5}$$

Differentiating (5.5) with respect to time (dropping time arguments to improve readability), and eliminating the variable x, we obtain:

$$\begin{aligned}
\dot{e} &= \dot{x} - \frac{\partial \pi}{\partial \hat{x}} \cdot \dot{\hat{x}} \\
&= f(x, w) + g(x, w) \cdot u - \frac{\partial \pi}{\partial \hat{x}} \cdot \hat{f}(\hat{x}, \hat{u})\Big|_{x=e+\pi(\hat{x})}, \\
&= f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, w) \cdot u,
\end{aligned} \tag{5.6}$$

where we have defined:

$$\begin{aligned}
f_e(e, \hat{x}, \hat{u}, w) &= f(\pi(\hat{x}) + e, w) - \frac{\partial \pi}{\partial \hat{x}} \cdot \hat{f}(\hat{x}, \hat{u}), \\
g_e(e, \hat{x}, w) &= g(\pi(\hat{x}) + e, w).
\end{aligned} \tag{5.7}$$

Assumption 4. *Assume the initial condition of error-state, $e(0)$, starts within the set $E_0 \subset \mathbb{R}^{n_x}$, i.e., $e(0) \in E_0$.*

For this dissertation, we consider a policy parametrization of the tracking controller given by:

$$u(t) = \kappa(e(t), \hat{x}(t), \hat{u}(t)), \ \kappa \in K_U \tag{5.8}$$

where the set $K_U := \{\kappa : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \to U\}$ defines a set of admissible error-state feedback control laws. The tracking controller (5.8) is to be designed such that $e(t) \in E$ for a bounded set E, for all $t \geq 0$, evolving with the dynamics (5.6) and (5.8). This set E is called the Robust Infinite-Time Forward Reachable Set of E_0, and is formally defined next.

Definition 14 (Robust Infinite-Time Forward Reachable Set)**.** *Consider* (5.6) *in closed-loop with* (5.8) *for all* $t \geq 0$ *as:*

$$\dot{e} = f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, w) \cdot \kappa(e, \hat{x}, \hat{u}), \tag{5.9}$$

with $\hat{x}$*,* $\hat{u}$*, and* w *constrained by* $\hat{X}$*,* $\hat{U}$*, and* W*. Then a* robust infinite-time forward reachable set E *of* E_0 *is defined as:*

$$E := \{e(t) \in \mathbb{R}^{n_x} : \exists\ e(0) \in E_0,\ \hat{x} : \mathbb{R}_+ \to \hat{X},\ \hat{u} : \mathbb{R}_+ \to \hat{U},\ w : \mathbb{R}_+ \to W,\ t \geq 0,\ \textit{s.t. } e(t) \textit{ is a solution to } (5.9)\}.$$

As computing the robust infinite-time forward reachable set E for a general nonlinear error dynamics (5.9) is hard, we find a tracking control law κ and compute an outer-bound $\varepsilon \supseteq E$. We refer to ε as a "tracking error bound" (TEB), and κ as the corresponding "tracking controller".

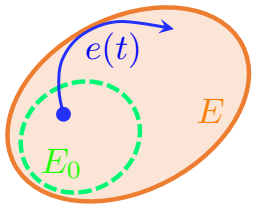

Figure 5.1: Illustration of Definition 14, with initial error set E_0, error trajectory $e(t)$, and robust infinite-time forward reachable set E.

Computing the TEB and Tracking Controller

The TEB ε and the tracking controller κ can be obtained with the following theorem.

Theorem 8. *Given the error dynamics* (5.6) *with mapping* $f_e : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$*,* $g_e : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$*,* $\gamma \in \mathbb{R}$*,* $\hat{X} \subseteq \mathbb{R}^{\hat{n}_x}$*,* $\hat{U} \subseteq \mathbb{R}^{\hat{n}_u}$*, and* $W \subseteq \mathbb{R}^{n_w}$*, if there exists a* $\mathcal{C}^1$ *function* $V : \mathbb{R}^{n_x} \to \mathbb{R}$ *and* $\kappa : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \to \mathbb{R}^{n_u}$ *such that*

$$E_0 \subseteq \Omega_\gamma^V, \tag{5.10a}$$

$$\frac{\partial V(e)}{\partial e} \cdot (f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, w) \cdot \kappa(e, \hat{x}, \hat{u})) < 0, \quad \forall e, \hat{x}, \hat{u}, w,\ \textit{s.t. } V(e) = \gamma,\ \hat{x} \in \hat{X},\ \hat{u} \in \hat{U},\ w \in W, \tag{5.10b}$$

$$\Omega_\gamma^V \subseteq \{e \in \mathbb{R}^{n_x} : \kappa(e, \hat{x}, \hat{u}) \in U\},\ \forall (\hat{x}, \hat{u}) \in \hat{X} \times \hat{U}, \tag{5.10c}$$

hold, then the sublevel set Ω_γ^V*, defined by* (2.1)*, is a TEB, denoted by* ε*, achieved by the tracking control law* κ*.*

Proof. The theorem is proved by contradiction. Assume there exist a time $t_3 > 0$, an initial condition $e_0 \in E_0$, and a trajectory $e(\cdot)$ such that $e(0) = e_0$, and $V(e(t_3)) > \gamma$. Since $V(e(0)) \leq \gamma$ from (5.10a), by continuity of V there exists t_1 and t_2, such that $0 \leq t_1 < t_2 < t_3$, $V(e(t_1)) = V(e(t_2)) = \gamma$, $t_1 = \sup_{V(t,e(t))<\gamma} t$, and $t_2 = \inf_{V(t,e(t))>\gamma} t$. Integrating (5.10b) over $[t_1, t_2]$ gives $V(e(t_2)) - V(e(t_1)) < 0$. Recall that $V(e(t_1)) = V(e(t_2)) = \gamma$, which is a contradiction. □

Finding generic functions V and κ that satisfy constraints (5.10) is a difficult problem. In this dissertation we use SOS programming to search for these functions by restricting ourselves to polynomial candidates $V \in \mathbb{R}[e]$, and $\kappa \in \mathbb{R}^{n_u}[(e, \hat{x}, \hat{u})]$. Besides this restriction, we make the following assumption:

Assumption 5. *The mappings $f_e \in \mathbb{R}^{n_x}[(e, \hat{x}, \hat{u}, w)]$, and $g_e \in \mathbb{R}^{n_x \times n_u}[(e, \hat{x}, w)]$ in error dynamics* (5.6) *are polynomials. Sets E_0, $\hat{X}$, $\hat{U}$, and W are semi-algebraic sets, i.e., there exists $p_0 \in \mathbb{R}[e]$ such that $E_0 = \{e \in \mathbb{R}^{n_x} \ : \ p_0(e) \leq 0\}$; with similar definitions for $\hat{X}$, $\hat{U}$, and W with polynomials $p_{\hat{x}} \in \mathbb{R}[\hat{x}]$, $p_{\hat{u}} \in \mathbb{R}[\hat{u}]$, and $p_w \in \mathbb{R}[w]$. The control constraint set U is a hypercube $U = \{u \in \mathbb{R}^{n_u} : \underline{u} \leq u \leq \overline{u}\}$, where $\underline{u}, \overline{u} \in \mathbb{R}^{n_u}$.*

By applying the generalized S-procedure [22] to the set containment constraints (5.10), and using the volume of Ω_γ^V as the cost function to minimize, we obtain the following SOS optimization problem for finding V and κ:

$$
\begin{aligned}
\min_{V,\kappa,s,l} \ & \text{volume}(\Omega_\gamma^V) \\
\text{s.t. } & s_0 \in \Sigma[e], s_{1\to 3} \in \Sigma[(e, \hat{x}, \hat{u}, w)], l \in \mathbb{R}[(e, \hat{x}, \hat{u}, w)] \\
& s_{4\to 9,i} \in \Sigma[(e, \hat{x}, \hat{u})], i \in \{1, ..., n_u\} && \text{(5.11a)} \\
& -(V(e) - \gamma) + s_0 \cdot p_0 \in \Sigma[e], && \text{(5.11b)} \\
& -\frac{\partial V}{\partial e} \cdot (f_e + g_e \cdot \kappa) - \epsilon e^\top e + l \cdot (V - \gamma) + s_1 \cdot p_{\hat{x}} \\
& \qquad + s_2 \cdot p_{\hat{u}} + s_3 \cdot p_w \in \Sigma[(e, \hat{x}, \hat{u}, w)], && \text{(5.11c)} \\
& \overline{u}_i - \kappa_i + s_{4,i} \cdot (V - \gamma) + s_{5,i} \cdot p_{\hat{x}} + s_{6,i} \cdot p_{\hat{u}} \in \Sigma[(e, \hat{x}, \hat{u})], \ i \in \{1, ..., n_u\}, && \text{(5.11d)} \\
& \kappa_i - \underline{u}_i + s_{7,i} \cdot (V - \gamma) + s_{8,i} \cdot p_{\hat{x}} + s_{9,i} \cdot p_{\hat{u}} \in \Sigma[(e, \hat{x}, \hat{u})], \ i \in \{1, ..., n_u\}. && \text{(5.11e)}
\end{aligned}
$$

In the formulation above, SOS polynomials $s_{1\to 3}$ and $s_{4\to 9,i}$ are multipliers used in the generalized S-procedure, and $\epsilon > 0$ is on the order of 10^{-6}. The optimization (5.11) is non-convex as there are two groups of decision variables V and $(\kappa, l, s_{4,i}, s_{7,i})$ bilinear in each other. To tackle this problem, similar to [57, Algorithm 1], we decompose it into two tractable subproblems to iteratively search between the two groups of decision variables. This is shown in the Appendix A.

So far, we have used a map π that only depends on the planning state $\hat{x}$ in (5.5). In some cases, however, this map may fail to provide reference signals for all the tracker states. That is why we now move to a more general map π that also depends on the planner inputs $\hat{u}$, following [56].

5.2 More General Map $\pi(\cdot)$

We begin this section with a motivating example to show why a more general map $\pi(\cdot)$ over the one in (5.5) is useful.

Example 4. *The ship motion at moderate speed can be modeled as in [58]:*

$$\begin{aligned} \dot{\eta} &= R(\psi)\nu + v_c, \\ M\dot{\nu} + C(\nu)\nu + D\nu &= \tau + R(\psi)^\top \tau_{wind}, \end{aligned} \tag{5.12}$$

where $\eta = [N; E; \psi]$ are the South-North and West-East positions and heading of the ship ($\psi = 0$ points North, $\psi = \pi/2$ points East), $\nu = [u; v; r]$ are the surge and sway velocities, and yaw rate of the ship. $R(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$ is a rotation matrix. $\tau \in \mathbb{R}^3$ is the control input affecting the three acceleration states of the ship. $v_c \in \mathbb{R}^3$ and $\tau_{wind} \in \mathbb{R}^3$ are disturbances corresponding to current velocities and wind forces. The inertia matrix including hydrodynamic added mass $M = \begin{bmatrix} 87.4 & 0 & 0 \\ 0 & 98.3 & 2.48 \\ 0 & 2.48 & 22.2 \end{bmatrix}$, damping matrix $D = \begin{bmatrix} 6.58 & 0 & 0 \\ 0 & 37.7 & 2.66 \\ 0 & 2.66 & 19.3 \end{bmatrix}$ and Coriolis matrix $C(\nu) = \nu(1) \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 98.3 \\ 0 & 0 & 2.48 \end{bmatrix}$ are chosen for a $1:30$ scale model of a platform supply vessel.

Using the notations from (5.1), we have state $x = [\eta; \nu] \in \mathbb{R}^6$, control input $u = \tau \in \mathbb{R}^3$ and disturbance input $w = [v_c; \tau_{wind}] \in \mathbb{R}^6$. The planning model is chosen as the kinematics part of the high-fidelity model (5.12)*:*

$$\dot{\hat{\eta}} = R(\hat{\psi})\hat{\nu} + \hat{v}_c \tag{5.13}$$

where the planning states, inputs and disturbances are $\hat{x} = \hat{\eta}$, $\hat{u} = \hat{\nu}$ and $\hat{w} = \hat{v}_c$. If we use the map $\pi(\hat{x}) = [\hat{x}; 0_{3\times 1}]$, then ν will become part of the resulting error state. As a result, the magnitude of the absolute state ν will be minimized in optimization (5.11)*, which doesn't make sense practically. To eliminate this issue, we can use a map $\pi(\hat{x}, \hat{u}) = [\hat{x}; \hat{u}]$, which also provides reference signals for ν.*

Modified Error Dynamics

As motivated above, we will use a more general map $\pi : \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \to \mathbb{R}^{n_x}$ to better provide reference trajectory for the tracking model, while all other methods in the literature [50, 54, 55, 59] only rely on the planning state. Correspondingly, the error state is redefined as

$$e = x - \pi(\hat{x}, \hat{u}). \tag{5.14}$$

The error dynamics resulting from (5.14) are given as

$$\dot{e} = f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, \hat{u}, w)u - \frac{\partial \pi(\hat{x}, \hat{u})}{\partial \hat{u}} \dot{\hat{u}}, \tag{5.15}$$

with $f_e(e, \hat{x}, \hat{u}, w) = f(e + \pi(\hat{x}, \hat{u}), w) - \frac{\partial \pi(\hat{x}, \hat{u})}{\partial \hat{x}} \hat{f}(\hat{x}, \hat{u})$ and $g_e(e, \hat{x}, \hat{u}, w) = g(e + \pi(\hat{x}, \hat{u}), w)$.

Time Varying Tracking Controller Synthesis

Note that planner input is applied in a zero order hold fashion between each sampling time as descirbed in (5.3). As the tracking error dynamics in (5.15) has a term containing $\dot{\hat{u}}$ (unlike (5.6)), this dynamics change discontinuously at each sampling time. Therefore, instead of considering a tracking controller for all times, we consider only a time interval between any two sampling steps. This leads to a time varying tracking controller. Since the signal $\hat{u}$ is piece-wise constant, we thus have

$$\dot{\hat{u}}(t) = 0, \ \forall t \in [\tau_k, \tau_{k+1}). \tag{5.16}$$

Therefore, the error dynamics (5.15) during the sampling time $[\tau_k, \tau_{k+1})$ is:

$$\dot{e} = f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, \hat{u}, w)u. \tag{5.17}$$

Given the bounded set of initial conditions E_0, we want to enforce the boundedness of the error state during $[0, T_s)$ by introducing a tracking controller

$$u(t) = \kappa(t, e(t), \hat{x}(t), \hat{u}(t)), \tag{5.18}$$

which is now defined by a *time-varying*, error-state feedback control law $\kappa : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \to \mathbb{R}^{n_u}$. Below, we provide the detailed explanation and the design requirements on κ to obtain such an error bound.

Proposition 2. *Given the error dynamics* (5.17) *with mappings* $f_e : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$, $g_e : \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \times \mathbb{R}^{n_w} \to \mathbb{R}^{n_x}$, *and* $\gamma \in \mathbb{R}$, $T_s > 0$, $\hat{X} \subseteq \mathbb{R}^{\hat{n}_x}$, $\hat{U} \subseteq \mathbb{R}^{\hat{n}_u}$, $W \subseteq \mathbb{R}^{n_w}$, *if there exists a* $\mathcal{C}^1$ *function* $V : \mathbb{R} \times \mathbb{R}^{n_x} \to \mathbb{R}$, *and* $\kappa : \mathbb{R} \times \mathbb{R}^{n_x} \times \mathbb{R}^{\hat{n}_x} \times \mathbb{R}^{\hat{n}_u} \to \mathbb{R}^{n_u}$, *such that*

$$E_0 \subseteq \Omega^V_{0,\gamma}, \tag{5.19a}$$

$$\frac{\partial V(t,e)}{\partial e} \cdot (f_e(e, \hat{x}, \hat{u}, w) + g_e(e, \hat{x}, \hat{u}, w) \cdot \kappa(t, e, \hat{x}, \hat{u})) + \frac{\partial V(t,e)}{\partial t} < 0,$$
$$\forall t, e, \hat{x}, \hat{u}, w, \ \text{s.t. } t \in [0, T_s), V(t,e) = \gamma, \ \hat{x} \in \hat{X}, \ \hat{u} \in \hat{U}, \ w \in W, \tag{5.19b}$$

$$\Omega^V_{t,\gamma} \subseteq \{e \in \mathbb{R}^{n_x} : \kappa(t, e, \hat{x}, \hat{u}) \in U\}, \ \forall (t, \hat{x}, \hat{u}) \in [0, T_s) \times \hat{X} \times \hat{U}, \tag{5.19c}$$

then for all $e(0) \in E_0$, *we have* $e(t) \in \Omega^V_{t,\gamma}$, *for all* $t \in [0, T_s)$.

Proof. The proof follows the proof of Theorem 8. □

Remark 3. *Although Proposition 2 is stated for the first sampling period* $[0, T_s)$, *it can be used for any other sampling period* $[\tau_k, \tau_{k+1})$ *with* $\tau_k = k \cdot T_s$. *Let* $e(\tau_k) \in \Omega^V_{0,\gamma}$. *Then we have* $e(\tau_k + t) \in \Omega^V_{t,\gamma}$, *for all* $t \in [0, T_s)$, *under the control signal* $u(\tau_k + t) = \kappa(t, e(\tau_k + t), \hat{x}(\tau_k + t), \hat{u}(\tau_k + t)))$.

Next, we focus on the effect of the input jump $\Delta\hat{u}$ at the end of each sampling period as in (5.3b). From (5.14), $\Delta\hat{u}$ induces a jump on the error. We quantify this jump next. For this we make the following simplifying assumption.

Assumption 6. *The map $\pi(\cdot,\cdot)$ is an affine map in its arguments: $\pi(\hat{x},\hat{u}) = P\cdot[\hat{x};\hat{u}]+\sigma$ where $P\in\mathbb{R}^{n_x\times(\hat{n}_x+\hat{n}_u)}$ and $\sigma\in\mathbb{R}^{n_x}$.*

Let τ_{k+1}^- and τ_{k+1}^+ denote sampling instant τ_{k+1} before and after the discrete jump, respectively. Then using Assumption 6, we write

$$\begin{aligned} & e(\tau_{k+1}^+) \\ &= x(\tau_{k+1}^+) - P\cdot[\hat{x}(\tau_{k+1}^+);\ \hat{u}(\tau_{k+1}^+)] - \sigma, \\ &= x(\tau_{k+1}^-) - P\cdot[\hat{x}(\tau_{k+1}^-);\ \hat{u}(\tau_{k+1}^-) + \Delta\hat{u}(\tau_{k+1}^+)] - \sigma, \\ &= e(\tau_{k+1}^-) - P\cdot\left[0;\ \Delta\hat{u}(\tau_{k+1}^+)\right]. \end{aligned} \tag{5.20}$$

We introduce the additional condition below to characterize the error jump induced by the control jump $\Delta\hat{u}$ in terms of the funnel $\Omega_{t,\gamma}^V$.

Proposition 3. *Given $\gamma\in\mathbb{R}$, $\Delta\hat{U}\in\mathbb{R}^{\hat{n}_u}$, if there exists a function $V:\mathbb{R}\times\mathbb{R}^{n_x}\to\mathbb{R}$ satisfying*

$$V(0, e - P\cdot[0;\Delta\hat{u}]) \le \gamma,\ \forall e,\Delta\hat{u},\ \text{s.t. } V(T_s,e)\le\gamma,\ \Delta\hat{u}\in\Delta\hat{U}, \tag{5.21}$$

then for all $e(\tau_{k+1}^-)\in\Omega_{T_s,\gamma}^V$, we have $e(\tau_{k+1}^+)\in\Omega_{0,\gamma}^V$.

Proof. Assume $e(\tau_{k+1}^-)\in\Omega_{T_s,\gamma}^V$, which is the same as $V(T_s, e(\tau_{k+1}^-))\le\gamma$. From (5.21) and $\Delta\hat{u}(\tau_{k+1}^-)\in\Delta\hat{U}$, it holds

$$V(0, e(\tau_{k+1}^-) - P\cdot[0;\ \Delta\hat{u}(\tau_{k+1}^-)]) \le \gamma. \tag{5.22}$$

Use (5.20) to show $V(0,e(\tau_{k+1}^+))\le\gamma$, that is to say $e(\tau_{k+1}^+)\in\Omega_{0,\gamma}^V$. □

We next combine the conditions for the error boundedness for each sampling period and discrete jump from Propositions 2 and 3, respectively, to obtain the main result on the boundedness of the error at all time, formulated below and illustrated in Figure 5.2.

Theorem 9. *If there exist V and κ satisfying* (5.19a)–(5.19c)*, and* (5.21)*, define $\varepsilon\subset\mathbb{R}^{n_x}$ such that*

$$\cup_{t\in[0,T_s)}\Omega_{t,\gamma}^V \subseteq \varepsilon.$$

Then for all $\hat{x}(t)\in\hat{X}$, $\hat{u}(t)\in\hat{U}$, $\Delta\hat{u}(t)\in\Delta\hat{U}$, and $w(t)\in W$, the error system (5.15) under control law $u(t)=\kappa(\tilde{t},e(t),\hat{x}(t),\hat{u}(t)))$ with $\tilde{t}=(t \mod T_s)\in[0,T_s)$ satisfies:

$$e(0)\in E_0 \implies e(t)\in\varepsilon,\ \forall t\ge 0,$$

that is to say, ε is a TEB achieved by the tracking control law κ.

Proof. From Remark 3 and for all $\tau_k = k \cdot T_s$, we have if $e(\tau_k) \in \Omega^V_{0,\gamma}$, then $e(\tau_k + \tilde{t}) \in \Omega^V_{\tilde{t},\gamma}$ and $e(\tau^-_{k+1}) \in \Omega^V_{T_s,\gamma}$. Then it follows from Proposition 3 that $e(\tau^+_{k+1}) \in \Omega^V_{0,\gamma}$. As a result, for all $e(0) \in E_0 \subseteq \Omega^V_{0,\gamma}$, we have $e(k \cdot T_s + \tilde{t}) \in \Omega^V_{\tilde{t},\gamma} \subseteq \varepsilon$, for all $k \geq 0$, and $\tilde{t} \in [0, T_s)$. □

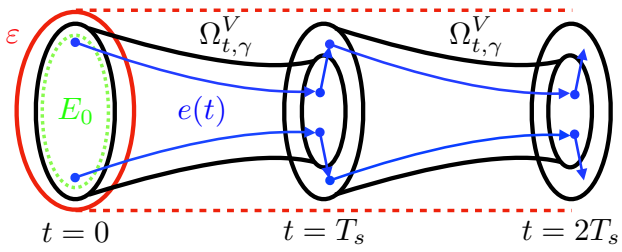

Figure 5.2: Illustration of Theorem 9, with initial error set E_0, funnels $\Omega^V_{t,\gamma}$ on each sampling period, bounded error jumps at sampling times, and TEB ε.

Again, to use SOS optimization to search for V and κ, we restrict them to polynomials: $V \in \mathbb{R}[(t, e)]$, and $\kappa \in \mathbb{R}[(t, e, \hat{x}, \hat{u})]$. In addition to Assumption 5, we assume $\Delta\hat{U} = \{\Delta\hat{u} \in \mathbb{R}^{\hat{n}_u} : p_\Delta(\Delta\hat{u}) \leq 0\}$, where $p_\Delta \in \mathbb{R}[\Delta\hat{u}]$. By choosing the integral of the volume of $\Omega(V, t, \gamma)$ over the time interval $[0, T_s]$ as the cost function, and applying the generalized S-procedure to (5.19a)–(5.19c), and (5.21), we obtain the following optimization problem:

$$
\begin{aligned}
\min_{V,\kappa,s,l} \quad & \int_0^{T_s} \text{volume}(\Omega^V_{t,\gamma}) dt \\
\text{s.t.} \quad & s_{1\to4} \in \Sigma[(t, e, \hat{x}, \hat{u}, w)], s_{5\to6} \in \Sigma[(e, \Delta\hat{u})], \\
& l \in \mathbb{R}[(t, e, \hat{x}, \hat{u}, w)], s_0 \in \Sigma[e], \\
& s_{7\to14,i} \in \Sigma[(t, e, \hat{x}, \hat{u})], i \in \{1, ..., n_u\}, && \text{(5.23a)} \\
& \gamma - V(0, e) + s_0 \cdot p_0 \in \Sigma[e], && \text{(5.23b)} \\
& -\left(\frac{\partial V}{\partial t} + \frac{\partial V}{\partial e} \cdot (f_e + g_e\kappa)\right) - \epsilon e^\top e + l \cdot (V - \gamma) \\
& \qquad + s_1 \cdot p_{\hat{x}} + s_2 \cdot p_{\hat{u}} + s_3 \cdot p_w - s_4 \cdot t(T_s - t) \in \Sigma[(t, e, \hat{x}, \hat{u}, w)], && \text{(5.23c)} \\
& -(V(0, e - P \cdot [0; \Delta\hat{u}]) - \gamma) + s_5 \cdot (V(T_s, e) - \gamma) + s_6 \cdot p_\Delta \in \Sigma[(e, \Delta\hat{u})], && \text{(5.23d)} \\
& \overline{u}_i - \kappa_i + s_{7,i} \cdot (V - \gamma) - s_{8,i} \cdot t(T_s - t) + s_{9,i} \cdot p_{\hat{x}} \\
& \qquad + s_{10,i} \cdot p_{\hat{u}} \in \Sigma[(t, e, \hat{x}, \hat{u})], i \in \{1, ..., n_u\}, && \text{(5.23e)} \\
& \kappa_i - \underline{u}_i + s_{11,i} \cdot (V - \gamma) - s_{12,i} \cdot t(T_s - t) + s_{13,i} \cdot p_{\hat{x}} \\
& \qquad + s_{14,i} \cdot p_{\hat{u}} \in \Sigma[(t, e, \hat{x}, \hat{u})], i \in \{1, ..., n_u\}. && \text{(5.23f)}
\end{aligned}
$$

The optimization is bilinear in two groups of decision variables V and $(\kappa, l, s_5, s_{7,i}, s_{11,i})$, and can also be solved using alternating direction method similar to Algorithm 4 in the Appendix A.

After the funnel $\Omega^V_{t,\gamma}$ is found, the next step is to compute a TEB ε by solving a convex optimization:

$$\begin{aligned} \min \ & \text{volume}(\varepsilon) \\ \text{s.t.} \ & \Omega^V_{t,\gamma} \subseteq \varepsilon, \ \forall t \in [0, T_s]. \end{aligned} \tag{5.24}$$

The set ε is restricted to a semi-algebraic set in order to convert the set containment constraint into an SOS constraint. Depending on the parameterization of ε, different cost function can be chosen. For example, if ε is an ellipsoid: $\varepsilon = \{e \in \mathbb{R}^{n_x} : e^\top P_\varepsilon e \leq 1\}$, where $P_\varepsilon \in \mathbb{S}^{n_x}_{++}$ is a decision variable, then $-\log\det(P_\varepsilon)$ can be used as a cost function; and if ε is a polytope: $\varepsilon = \{e \in \mathbb{R}^{n_x} : A_\varepsilon e \leq b_\varepsilon\}$, where $A_\varepsilon \in \mathbb{R}^{n_\varepsilon \times n_x}$ is fixed, and $b_\varepsilon \in \mathbb{R}^{n_\varepsilon}$ is a decision variable, then $\sum_{i=1}^{n_\varepsilon} b_{\varepsilon,i}$ can be used as a cost function, where $b_{\varepsilon,i}$ is the i-th element of b_ε.

Once a TEB ε is computed from the SOS optimizations (5.11), or (5.23)-(5.24), it can be used to "robustify" the planner, meaning that the planner will be redesigned to account for the computed TEB to guarantee safety. What we mean by safety is the high-fidelity state satisfying: $x(t) \in X$ for $t \geq 0$. Specifically, to ensure safety, the following condition needs to hold:

$$\hat{S} \oplus \varepsilon \subseteq X, \ \text{where } \hat{S} = \{\pi(\hat{x}, \hat{u}) : \ \hat{x} \in \hat{X}, \ \hat{u} \in \hat{U}\}. \tag{5.25}$$

Moreover, if the planner has the terminal constraint: $x(T) \in X_r$ for some T, where $X_r \subseteq X$. The target sets $\hat{X}_r \subseteq \hat{X}$ and $\hat{U}_r \subseteq \hat{U}$ for the planning layer should also be redesigned:

$$\hat{S}_r \oplus \varepsilon \subseteq X_r, \ \text{where } \hat{S}_r = \{\pi(\hat{x}, \hat{u}) : \ \hat{x} \in \hat{X}_r, \ \hat{u} \in \hat{U}_r\}, \tag{5.26}$$

and the planner need to enforce $\hat{x}(T) \in \hat{X}_r$ and $\hat{u}(T) \in \hat{U}_r$.

5.3 Numerical Examples

We continue with the ship motion planning and control problem for autonomous docking described in Example 4. The autonomous docking maneuver consists of four phases: transit, transition from high speed to low speed maneuvering, docking, and dockside keeping a steady contact force with the dock. In this dissertation, we focus on the transition phase, which is challenging due to large changes in the ship dynamics when the speed is reduced. This means that we consider a *reach-avoid* problem to reach the area near the dock (light blue in Figure 5.3) while avoiding the piers (gray areas). The chosen reach-avoid problem focuses on the first three states with the safety constraints $X = [0, 10] \times [0, 6.5] \times [-\pi, \pi] \times \mathbb{R}^3$, the obstacles $X_a = X_{a1} \cup X_{a2}$ with $X_{a1} = [2, 2.5] \times [0, 3] \times [-\pi, \pi] \times \mathbb{R}^3$ and $X_{a2} = [5, 5.5] \times [3.5, 6.5] \times [-\pi, \pi] \times \mathbb{R}^3$ (in grey in Figure 5.3), and the target set $X_r = [7, 10] \times [0, 6.5] \times [\pi/3, 2\pi/3] \times \mathbb{R}^3$ (light blue).

As described in Example 4, the map π is chosen as $\pi(\hat{x}, \hat{u}) = [\hat{x}; \hat{u}]$ to provide reference signals for all the states of the high-fidelity model. However, instead of defining error as in (5.14), we redefine the error state as $e = \phi \cdot (x - \pi(\hat{x}, \hat{u}))$, where $\phi = \left[R^{-1}(\hat{\psi}), \mathbf{0}_{3\times3}; \mathbf{0}_{3\times3}, \boldsymbol{I}_3\right]$.

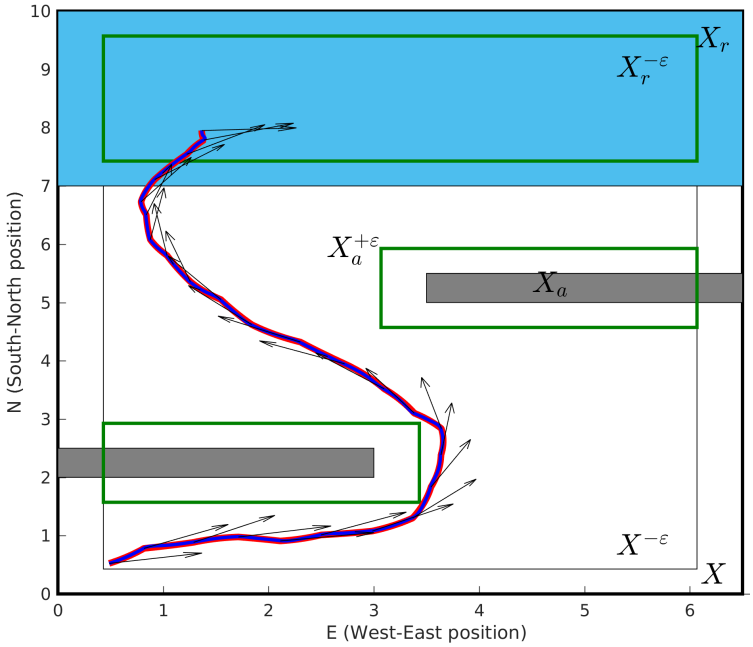

Figure 5.3: Closed-loop trajectories of the low-fidelity (red) and high-fidelity (blue) models in the (N, E)-plane with the ship heading ψ (black arrows), the initial and shrunk state constraints X and $X^{-\varepsilon}$ (thick and thin black lines), the target set X_r (light blue), the obstacles X_a (grey) and the shrunk target set $X_r^{-\varepsilon}$ and expanded obstacles $X_a^{+\varepsilon}$ (green).

The matrix ϕ allows to replace the trigonometric functions in $\hat{\psi}$ in the error dynamics (5.15) by trigonometric functions in $e(3) = (\psi - \hat{\psi})$, which can easily be approximated by polynomials in certain range of $e(3)$. The sampling period is $T_s = 3$.

For the high-fidelity model, the controls are unconstrained ($U = \mathbb{R}^3$) and the disturbances signals are assumed to be evolve in $W = [-0.01, 0.01]^5 \times [-0.05, 0.05]$. The input, input jump, and disturbances spaces for the planning model are $\hat{U} = [0, 0.18] \times [-0.05, 0.05] \times [-0.1, 0.1]$, $\Delta\hat{U} = [-0.18, 0.18] \times [-0.1, 0.1] \times [-0.2, 0.2]$, and $\hat{W} = [-0.01, 0.01]^3$. Optimization (5.23) is run with degree-2 polynomials to characterize the storage function V, control law κ, and multipliers s, l, and terminates in 6 minutes on a computer with 3.6GHz processor and 62GB of RAM. A TEB ε is computed by parameterizing it as a hypercube, and solving (5.24). The obtained TEB ε on (N, E, ψ) are $[-0.427, 0.427] \times [-0.432, 0.432] \times [-0.235, 0.235]$. Based on the TEB, define the shrunk state constraint set, the expanded obstacle, and the shrunk target set as $X^{-\varepsilon} := \{x \in \mathbb{R}^n \mid x + e \subseteq X, e \in \varepsilon\}$, $X_a^{+\varepsilon} := \{x + e \in \mathbb{R}^n \mid x \in X_a,\ e \in \varepsilon\}$, and $X_r^{-\varepsilon} := \{x \in \mathbb{R}^n \mid x + e \subseteq X_r, e \in \varepsilon\}$, respectively, and they are shown in Figure 5.3. According to (5.25) and (5.26) , the planner state $\hat{x}$ needs to stay in the shrunk state constraint set $X^{-\varepsilon}$, avoid the expanded obstacles $X_a^{+\varepsilon}$, and reach shrunk target set $X_r^{-\varepsilon}$ in order for the state of the high-fidelity model to stay in X, avoid X_a, and reach X_r.

In this dissertation, we use the discrete abstraction method proposed in [60] as the planning algorithm to enforce the constraint sets $\hat{U}$ and $\Delta\hat{U}$, and realize the reach-avoid

specification. The initial state is chosen as a random point in the bottom left corner of the (N, E)-plane, and both closed-loop trajectories with random disturbance signals are plotted in red for (5.13) and blue for (5.12) in Figure 5.3. The black arrows represent the orientation ψ of the ship at each discrete time step. We can first note that the low-level controller (5.18) provides a very efficient tracking of the planning model's trajectory (red) by the high-fidelity model (blue). Both models satisfy their reach-avoid specifications by reaching the (shrunk) target set in blue while avoiding the (expanded) obstacles in grey. Once the ship has reached the desired $[N; E]$ position (blue set) but not the correct orientation ψ, we can see it slowly drift sideways while it turns to face East.

5.4 Chapter Summary

This chapter presents a hierarchical trajectory planning and control framework for nonlinear dynamical systems. We use a hierarchy of system models for the planner-tracker synthesis: a low-fidelity model initially plans a feasible trajectory satisfying the *planning constraints*, and then the high-fidelity model is used to design the tracking controller and track the planned trajectory. With the tracking controller, the error states between the planner and the tracker are restricted to a bounded set, called the *tracking error bound*. We consider error states that are functions of both the planner states and inputs, which offers more freedom in the choice of the low-fidelity model. Both the tracking error bound and the controller are designed offline using SOS programming. Finally, the method is demonstrated on two detailed numerical examples.

Chapter 6

Robust Stability Analysis for Systems with Neural Network Controllers

This chapter presents a method to analyze the stability of feedback systems with neural network controllers. Two stability theorems are given to prove asymptotic stability and to compute an ellipsoidal inner-approximation to the region of attraction (ROA). The first theorem provides a condition to prove stability and to inner-approximate the region of attraction (ROA) for a linear time-invariant (LTI) plant. It uses Lyapunov theory, and local sector quadratic constraints (QCs) to bound the nonlinear activation functions in the NN. The second theorem allows the plant to include perturbations such as unmodeled dynamics, and slope-restricted nonlinearities, characterizing them with integral quadratic constraints (IQCs) [13, 18]. This in turn allows for the use of off-by-one IQCs [61] to capture the slope restrictions of activation functions. Both results rely on semidefinite programming to approximate the ROA.

The proposed framework provides robustness guarantees for the feedback system with uncertain plants, which is modeled as an interconnection of the nominal plant and perturbations that are described by IQCs. The use of IQCs also allows for plants that are not necessarily LTI. Moreover, the proposed framework allows for local (dynamic) off-by-one IQCs to further sharpen the description of activation functions by capturing their slope restrictions. This differs from [11, 12, 62–64], which derive only static QCs for activation functions.

Local (static) sector IQCs have been used in the stability analysis of linear systems with actuator saturation [65, 66], and unbounded nonlinearities [67]. The description of these nonlinearties are refined by incorporating soft (dynamic) IQCs in the stability analysis framework for linear systems [14], and polynomial systems [68]. Compared with these works, this work is specialized to NN-controlled systems: it exploits the specific properties of NNs and uses the Interval Bound Propagation method [69] to derive non-conservative static and dynamic local IQCs to describe NN controllers; and it also allows for the analysis of NN-controlled nonlinear systems by accommodating perturbations.

In this chapter, we first present the problem formulation and the nominal stability analysis

framework when the plant is LTI. Then we extend the framework to uncertain systems using IQCs. Finally, we provide numerical examples, including a nonlinear inverted pendulum and an uncertain vehicle model.

6.1 Nominal Stability Analysis

Problem Formulation

Consider the feedback system consisting of a plant G and state-feedback controller π as shown in Figure 6.1. As a first step, we assume the plant G is a linear, time-invariant (LTI) system defined by the following discrete-time model:

$$x(k+1) = A_G\, x(k) + B_G\, u(k), \tag{6.1}$$

where $x(k) \in \mathbb{R}^{n_G}$ is the state, $u(k) \in \mathbb{R}^{n_u}$ is the input, $A_G \in \mathbb{R}^{n_G \times n_G}$ and $B_G \in \mathbb{R}^{n_G \times n_u}$. The controller $\pi : \mathbb{R}^{n_G} \to \mathbb{R}^{n_u}$ is an ℓ-layer, feed-forward neural network (NN) defined as:

$$w^0(k) = x(k), \tag{6.2a}$$

$$w^i(k) = \phi^i\left(W^i w^{i-1}(k) + b^i \right),\ i = 1, \ldots, \ell, \tag{6.2b}$$

$$u(k) = W^{\ell+1} w^{\ell}(k) + b^{\ell+1}, \tag{6.2c}$$

where $w^i \in \mathbb{R}^{n_i}$ are the outputs (activations) from the i^{th} layer and $n_0 = n_G$. The operations for each layer are defined by a weight matrix $W^i \in \mathbb{R}^{n_i \times n_{i-1}}$, bias vector $b^i \in \mathbb{R}^{n_i}$, and activation function $\phi^i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$. The activation function ϕ^i is applied element-wise, i.e.

$$\phi^i(v) := \left[\varphi(v_1), \cdots, \varphi(v_{n_i})\right]^\top, \tag{6.3}$$

where $\varphi : \mathbb{R} \to \mathbb{R}$ is the (scalar) activation function selected for the NN. Common choices for the scalar activation function include $\varphi(\nu) := \tanh(\nu)$, sigmoid $\varphi(\nu) := \frac{1}{1+e^{-\nu}}$, ReLU $\varphi(\nu) := \max(0, \nu)$, and leaky ReLU $\varphi(\nu) := \max(a\nu, \nu)$ with $a \in (0, 1)$. We assume the activation φ is identical in all layers; this can be relaxed with minor changes to the notation.

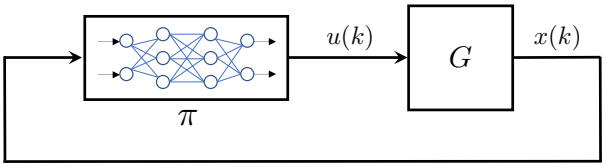

Figure 6.1: Feedback system with plant G and NN π

The state vector x_* is an equilibrium point of the feedback system with input u_* if the following conditions hold:

$$x_* = A_G\, x_* + B_G\, u_*, \tag{6.4a}$$

$$u_* = \pi(x_*). \tag{6.4b}$$

Let $\chi(k; x_0)$ denote the solution to the feedback system at time k from initial condition $x(0) = x_0$. Our goal is to analyze asymptotic stability of the equilibrium point and to find the largest estimate of the region of attraction, defined below, using an ellipsoidal inner approximation.

Definition 15. *The region of attraction (ROA) of the feedback system with plant G and NN π is defined as*

$$\mathcal{R} := \{x_0 \in \mathbb{R}^{n_G} : \lim_{k\to\infty} \chi(k; x_0) = x_*\}. \tag{6.5}$$

NN Representation: Isolation of Nonlinearities

It is useful to isolate the nonlinear activation functions from the linear operations of the NN as done in [62, 70]. Define v^i as the input to the activation function ϕ^i:

$$v^i(k) := W^i w^{i-1}(k) + b^i,\ i = 1, \ldots, \ell. \tag{6.6}$$

The nonlinear operation of the i^{th} layer (6.2b) is thus expressed as $w^i(k) = \phi^i(v^i(k))$. Gather the inputs and outputs of all activation functions:

$$v_\phi := \begin{bmatrix} v^1 \\ \vdots \\ v^\ell \end{bmatrix} \in \mathbb{R}^{n_\phi} \text{ and } w_\phi := \begin{bmatrix} w^1 \\ \vdots \\ w^\ell \end{bmatrix} \in \mathbb{R}^{n_\phi}, \tag{6.7}$$

where $n_\phi := n_1 + \cdots + n_\ell$, and define the combined nonlinearity $\phi : \mathbb{R}^{n_\phi} \to \mathbb{R}^{n_\phi}$ by stacking the activation functions:

$$\phi(v_\phi) := \begin{bmatrix} \phi^1(v^1) \\ \vdots \\ \phi^\ell(v^\ell) \end{bmatrix}. \tag{6.8}$$

Thus $w_\phi(k) = \phi(v_\phi(k))$, where the scalar activation function φ is applied element-wise to each entry of v_ϕ. Finally, the NN control policy π defined in (6.2) can be rewritten as:

$$\begin{bmatrix} u(k) \\ v_\phi(k) \end{bmatrix} = N \begin{bmatrix} x(k) \\ w_\phi(k) \\ 1 \end{bmatrix} \tag{6.9a}$$

$$w_\phi(k) = \phi(v_\phi(k)). \tag{6.9b}$$

The matrix N depends on the weights and biases as follows, where the vertical and horizontal bars partition N compatibly with the inputs $(x, w_\phi, 1)$ and outputs (u, v_ϕ):

$$N := \left[\begin{array}{c|cccc|c} 0 & 0 & 0 & \cdots & W^{\ell+1} & b^{\ell+1} \\ \hline W^1 & 0 & \cdots & 0 & 0 & b^1 \\ 0 & W^2 & \cdots & 0 & 0 & b^2 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & W^\ell & 0 & b^\ell \end{array}\right] \tag{6.10a}$$

$$:= \begin{bmatrix} N_{ux} & N_{uw} & N_{ub} \\ N_{vx} & N_{vw} & N_{vb} \end{bmatrix}. \tag{6.10b}$$

This decomposition of the NN, depicted in Figure 6.2, isolates the activation functions in preparation for the stability analysis.

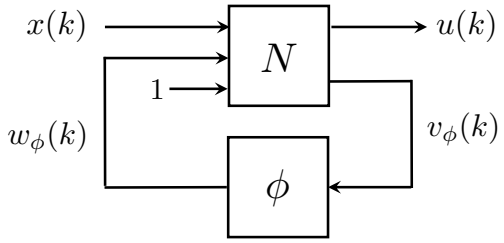

Figure 6.2: NN representation to isolate the nonlinearities ϕ.

Suppose (x_*, u_*) satisfies (6.4). Then x_* can be propagated through the NN to obtain equilibrium values v_*^i, w_*^i for the inputs/outputs of each activation function ($i = 1, \ldots, \ell$), yielding $(v_\phi, w_\phi) = (v_*, w_*)$. Thus (x_*, u_*, v_*, w_*) is an equilibrium point of (6.1) and (6.2) if:

$$x_* = A_G\, x_* + B_G\, u_*, \tag{6.11a}$$

$$\begin{bmatrix} u_* \\ v_* \end{bmatrix} = N \begin{bmatrix} x_* \\ w_* \\ 1 \end{bmatrix}, \tag{6.11b}$$

$$w_* = \phi(v_*). \tag{6.11c}$$

Quadratic Constraints: Scalar Activation Functions

The stability analysis relies on quadratic constraints (QCs) to bound the activation function. A typical constraint is the sector bound as defined next.

Definition 16. *Let $\alpha \le \beta$ be given. The function $\varphi : \mathbb{R} \to \mathbb{R}$ lies in the (global) sector $[\alpha, \beta]$ if:*

$$(\varphi(\nu) - \alpha\nu) \cdot (\beta\nu - \varphi(\nu)) \ge 0 \;\; \forall \nu \in \mathbb{R}. \tag{6.12}$$

The interpretation of the sector $[\alpha, \beta]$ is that φ lies between lines passing through the origin with slope α and β. Many activation functions are bounded in the sector $[0, 1]$, e.g. tanh and ReLU. Figure 6.3 illustrates $\varphi(\nu) = \tanh(\nu)$ (blue solid) and the global sector defined by $[0, 1]$ (red solid lines).

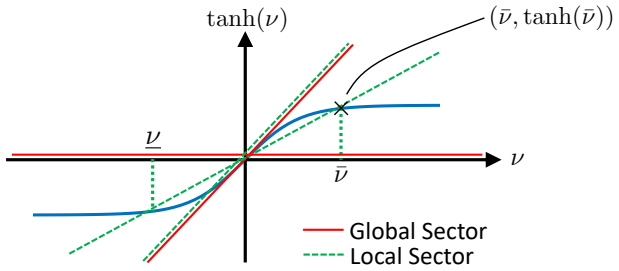

Figure 6.3: Sector constraints on tanh

The global sector constraint is often too coarse for stability analysis, and a local sector constraint provides tighter bounds.

Definition 17. *Let α, β, $\underline{\nu}$, $\bar{\nu} \in \mathbb{R}$ with $\alpha \leq \beta$ and $\underline{\nu} \leq 0 \leq \bar{\nu}$. The function $\varphi : \mathbb{R} \to \mathbb{R}$ satisfies the local sector $[\alpha, \beta]$ if*

$$(\varphi(\nu) - \alpha\,\nu) \cdot (\beta\,\nu - \varphi(\nu)) \geq 0 \;\; \forall \nu \in [\underline{\nu}, \bar{\nu}]. \tag{6.13}$$

As an example, $\varphi(\nu) := \tanh(\nu)$ restricted to the interval $[-\bar{\nu}, \bar{\nu}]$ satisfies the local sector bound $[\alpha, \beta]$ with $\alpha := \tanh(\bar{\nu})/\bar{\nu} > 0$ and $\beta := 1$. As shown in Figure 6.3 (green dashed lines), the local sector provides a tighter bound than the global sector. These bounds are valid for a symmetric interval around the origin with $\underline{\nu} = -\bar{\nu}$; non-symmetric intervals ($\underline{\nu} \neq -\bar{\nu}$) can be handled similarly.

The local and global sector constraints above were defined to be centered at the point $(\nu, \varphi(\nu)) = (0, 0)$. The stability analysis will require offset sectors centered around an arbitrary point $(\nu_*, \varphi(\nu_*))$ on the function. For example, $\varphi(\nu) = \tanh(\nu)$ satisfies the global sector bound (red solid) around the point $(\nu_*, \tanh(\nu_*))$ with $[\alpha, \beta] = [0, 1]$, as shown in Figure 6.4. It satisfies a tighter local sector bound (green dashed) when the input is restricted to $\nu \in [\underline{\nu}, \bar{\nu}]$. An explicit expression for this local sector is $\beta = 1$ and

$$\alpha := \min\left(\frac{\tanh(\bar{\nu}) - \tanh(\nu_*)}{\bar{\nu} - \nu_*}, \frac{\tanh(\nu_*) - \tanh(\underline{\nu})}{\nu_* - \underline{\nu}}\right).$$

The local sector upper bound β can be tightened further. This leads to the following definition of an offset local sector.

Definition 18. *Let α, β, $\underline{\nu}$, $\bar{\nu}$, $\nu_* \in \mathbb{R}$ be given with $\alpha \leq \beta$ and $\underline{\nu} \leq \nu_* \leq \bar{\nu}$. The function $\varphi : \mathbb{R} \to \mathbb{R}$ satisfies the offset local sector $[\alpha, \beta]$ around the point $(\nu_*, \varphi(\nu_*))$ if:*

$$(\Delta\varphi(\nu) - \alpha\,\Delta\nu) \cdot (\beta\,\Delta\nu - \Delta\varphi(\nu)) \geq 0 \;\; \forall \nu \in [\underline{\nu}, \bar{\nu}] \tag{6.14}$$

where $\Delta\varphi(\nu) := \varphi(\nu) - \varphi(\nu_*)$ and $\Delta\nu := \nu - \nu_*$.

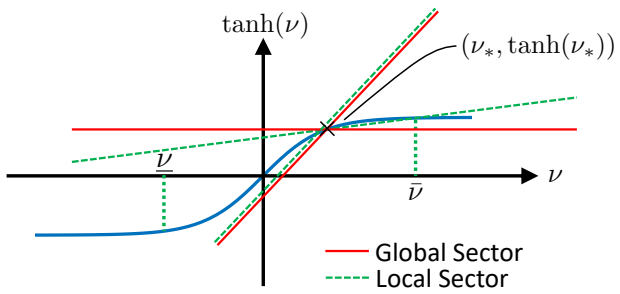

Figure 6.4: Offset local sector constraint on tanh

Quadratic Constraints: Combined Activation Functions

Offset local sector constraints can also be defined for the combined nonlinearity ϕ, given by (6.8). Let $\underline{v}, \bar{v}, v_* \in \mathbb{R}^{n_\phi}$ be given with $\underline{v} \leq v_* \leq \bar{v}$. Assume that the activation input $v_\phi \in \mathbb{R}^{n_\phi}$ lies, element-wise, in the interval $[\underline{v}, \bar{v}]$ and the i^{th} input/output pair is $w_{\phi,i} = \varphi(v_{\phi,i})$. Further assume the scalar activation function satisfies the local sector $[\alpha_i, \beta_i]$ around the point $v_{*,i}$ with the input restricted to $v_{\phi,i} \in [\underline{v}_i, \bar{v}_i]$ for $i = 1, \ldots, n_\phi$. The local sector bounds can be computed for φ on the given interval either analytically (as above for tanh) or numerically. These local sectors can be stacked into vectors $\alpha_\phi, \beta_\phi \in \mathbb{R}^{n_\phi}$ that provide QCs satisfied by the combined nonlinearity ϕ.

Lemma 2. *Let α_ϕ, β_ϕ, $\underline{v}$, $\bar{v}$, $v_* \in \mathbb{R}^{n_\phi}$ be given with $\alpha_\phi \leq \beta_\phi$, $\underline{v} \leq v_* \leq \bar{v}$, and $w_* := \phi(v_*)$. Assume ϕ satisfies the offset local sector $[\alpha_\phi, \beta_\phi]$ around the point (v_*, w_*) element-wise for all $v_\phi \in [\underline{v}, \bar{v}]$. If $\lambda \in \mathbb{R}^{n_\phi}$ with $\lambda \geq 0$ then:*

$$\begin{bmatrix} v_\phi - v_* \\ w_\phi - w_* \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi - v_* \\ w_\phi - w_* \end{bmatrix} \geq 0, \ \forall v_\phi \in [\underline{v}, \bar{v}], \ w_\phi = \phi(v_\phi), \tag{6.15}$$

where

$$\Psi_\phi := \begin{bmatrix} diag(\beta_\phi) & -I_{n_\phi} \\ -diag(\alpha_\phi) & I_{n_\phi} \end{bmatrix} \tag{6.16}$$

$$M_\phi(\lambda) := \begin{bmatrix} 0_{n_\phi} & diag(\lambda) \\ diag(\lambda) & 0_{n_\phi} \end{bmatrix}. \tag{6.17}$$

Proof. For any $v_\phi \in \mathbb{R}^{n_\phi}$ and $w_\phi = \phi(v_\phi)$:

$$\begin{bmatrix} v_\phi - v_* \\ w_\phi - w_* \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi - v_* \\ w_\phi - w_* \end{bmatrix} = \sum_{i=1}^{n_\phi} \lambda_i (\Delta w_i - \alpha_i \, \Delta v_i) \cdot (\beta_i \, \Delta v_i - \Delta w_i)$$

where $\Delta w_i := \varphi(v_{\phi,i}) - \varphi(v_{*,i})$ and $\Delta v_i := v_{\phi,i} - v_{*,i}$. If $v_\phi \in [\underline{v}, \bar{v}]$ then each term in the sum is non-negative by the offset local sector constraints and $\lambda \geq 0$. □

In order to apply the local sector and slope bounds in the stability analysis, we must first compute the bounds $\underline{v}, \overline{v} \in \mathbb{R}^{n_\phi}$ on the activation input v_ϕ. The process to compute the bounds is briefly discussed here with more details provided in [69]. Let v_*^1 be the equilibrium value at the first NN layer. Select $\underline{v}^1$, $\bar{v}^1 \in \mathbb{R}^{n_1}$ with $\underline{v}^1 \leq v_*^1 \leq \bar{v}^1$. If the activation functions ϕ^1 are monotonically non-decreasing, then the first activation output $w^1 = \phi^1(v^1)$ is bounded by $[\underline{w}^1, \bar{w}^1]$, where $\underline{w}^1 = \phi^1(\underline{v}^1)$, and $\overline{w}^1 = \phi^1(\overline{v}^1)$. The interval $[\underline{w}^1, \bar{w}^1]$ can then be used to compute bounds $[\underline{v}^2, \bar{v}^2]$ on the input v^2 to the next activation function, where $v^2 = W^2 w^1 + b^2$. Denote $y^\top$ as the i^{th} row of W^2, and define $c := \frac{1}{2}(\bar{w}^1 + \underline{w}^1)$ and $r := \frac{1}{2}(\bar{w}^1 - \underline{w}^1)$. Then $\underline{v}^2$ and $\bar{v}^2$ are given as

$$\bar{v}_i^2 = y^\top c + b_i^2 + \sum_{j=1}^{n_1} |y_j r_j|, \tag{6.18a}$$

$$\underline{v}_i^2 = y^\top c + b_i^2 - \sum_{j=1}^{n_1} |y_j r_j|. \tag{6.18b}$$

The intervals computed for w^1 and v^2 will contain their equilibrium value w_*^1 and v_*^2. This process can be propagated through all layers of the NN to obtain the bounds $\underline{v}, \overline{v} \in \mathbb{R}^{n_\phi}$ for the activation function input v_ϕ. The remainder of the dissertation will assume the local sector bounds have been computed as briefly summarized in the following property.

Property 1. *Let $v_* \in \mathbb{R}^{n_\phi}$ be an equilibrium value of the activation input and $v_*^1 \in \mathbb{R}^{n_1}$ be the corresponding value at the first layer. Let $\underline{v}^1$, $\bar{v}^1 \in \mathbb{R}^{n_1}$ with $v_*^1 \in [\underline{v}^1, \bar{v}^1]$ and their corresponding activation input bounds $\underline{v}, \overline{v}$ be given. There exist α_ϕ, $\beta_\phi \in \mathbb{R}^{n_\phi}$ such that ϕ satisfies the offset local sector around the point $(v_*, \phi(v_*))$ for all $v_\phi \in [\underline{v}, \bar{v}]$.*

Lyapunov Condition

This section uses a Lyapunov function and the offset local sector to compute an inner approximation for the ROA of the feedback system of G and π. To simplify notation, the interval bound on v^1 is assumed to be symmetrical about v_*^1, i.e. $\underline{v}^1 = 2v_*^1 - \bar{v}^1$ so that $\bar{v}^1 - v_*^1 = v_*^1 - \underline{v}^1$. This can be relaxed to handle non-symmetrical intervals with minor notational changes.

Theorem 10. *Consider the feedback system of plant G in (6.1) and NN π in (6.2) with equilibrium point (x_*, u_*, v_*, w_*) satisfying (6.11). Let $\bar{v}^1 \in \mathbb{R}^{n_1}$, $\underline{v}^1 := 2v_*^1 - \bar{v}^1$, and $\alpha_\phi, \beta_\phi \in \mathbb{R}^{n_\phi}$ be given vectors satisfying Property 1 for the NN. Denote the i^{th} row of the first weight W^1 by W_i^1 and define matrices*

$$R_V := \begin{bmatrix} I_{n_G} & 0_{n_G \times n_\phi} \\ N_{ux} & N_{uw} \end{bmatrix}, \text{ and } R_\phi := \begin{bmatrix} N_{vx} & N_{vw} \\ 0_{n_\phi \times n_G} & I_{n_\phi} \end{bmatrix}.$$

If there exists a matrix $P \in \mathbb{S}_{++}^{n_G}$, and vector $\lambda \in \mathbb{R}^{n_\phi}$ with $\lambda \geq 0$ such that

$$R_V^\top \begin{bmatrix} A_G^\top P A_G - P & A_G^\top P B_G \\ B_G^\top P A_G & B_G^\top P B_G \end{bmatrix} R_V + R_\phi^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi R_\phi < 0, \tag{6.19}$$

$$\begin{bmatrix} (\bar{v}_i^1 - v_{*,i}^1)^2 & W_i^1 \\ W_i^{1\top} & P \end{bmatrix} \geq 0, \;\; i = 1, \cdots, n_1, \tag{6.20}$$

then: (i) the feedback system consisting of G and π is locally stable around x_*, and (ii) the set $\mathcal{E}(P, x_*)$, defined by (2.3), is an inner-approximation to the ROA.

Proof. By Schur complements, (6.20) is equivalent to:

$$W_i^1 P^{-1} W_i^{1\top} \leq (\bar{v}_i^1 - v_{*,i}^1)^2, \;\; i = 1, \cdots, n_1. \tag{6.21}$$

It follows from Lemma 1 in [71] that:

$$\mathcal{E}(P, x_*) \subseteq \{x \in \mathbb{R}^{n_G} : \underline{v}^1 - v_*^1 \leq W^1(x - x_*) \leq \bar{v}^1 - v_*^1\}.$$

Finally, use $v^1 - v_*^1 = W^1(x - x_*)$ to rewrite this as:

$$\mathcal{E}(P, x_*) \subseteq \{x : \underline{v}^1 \leq v^1 \leq \bar{v}^1\}.$$

To summarize, feasibility of (6.20) verifies that if $x(k) \in \mathcal{E}(P, x_*)$ then $v^1(k) \in [\underline{v}^1, \bar{v}^1]$ and hence the offset local sector conditions are valid.

Next, since the LMI in (6.19) is strict, there exists $\epsilon > 0$ such that left / right multiplication of the LMI by $\begin{bmatrix} (x(k) - x_*)^\top & (w_\phi(k) - w_*)^\top \end{bmatrix}$ and its transpose yields

$$\begin{bmatrix} \star \end{bmatrix}^\top \begin{bmatrix} A_G^\top P A_G - P & A_G^\top P B_G \\ B_G^\top P A_G & B_G^\top P B_G \end{bmatrix} \begin{bmatrix} x(k) - x_* \\ u(k) - u_* \end{bmatrix} + \begin{bmatrix} \star \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi(k) - v_* \\ w_\phi(k) - w_* \end{bmatrix} \leq -\epsilon |x(k) - x_*|^2.$$

where the entries denoted by $\star$ can be inferred from symmetry. Define the Lyapunov function $V(x) := (x - x_*)^\top P (x - x_*)$ and use (6.1) and (6.11) to show:

$$V(x(k+1)) - V(x(k)) + \begin{bmatrix} \star \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi(k) - v_* \\ w_\phi(k) - w_* \end{bmatrix} \leq -\epsilon |x(k) - x_*|^2. \tag{6.22}$$

Assume $x(k) \in \mathcal{E}(P, x_*)$ for some $k \geq 0$, i.e., $V(x(k)) \leq 1$. As noted above, $x(k) \in \mathcal{E}(P, x_*)$ implies the offset local sector $[\alpha_\phi, \beta_\phi]$ around v_*. Then, by Lemma 2, the final term on the left side of (6.22) is ≥ 0, and thus from (6.22) we have $V(x(k+1)) \leq 1$, i.e., $x(k+1) \in \mathcal{E}(P, x_*)$. By induction, we have that $\mathcal{E}(P, x_*)$ is forward invariant, i.e., $x(0) \in \mathcal{E}(P, x_*) \implies x(k) \in \mathcal{E}(P, x_*) \; \forall k \geq 0$. As a result, if $x(0) \in \mathcal{E}(P, x_*)$, then the final term on the left side of (6.22) is ≥ 0 for all $k \geq 0$, and $V(x(k+1)) - V(x(k)) \leq -\epsilon |x(k) - x_*|^2$ for all $k \geq 0$. It follows from a Lyapunov argument, e.g. Theorem 4.1 in [72], that x_* is an asymptotically stable equilibrium point and $\mathcal{E}(P, x_*)$ is an inner approximation of the ROA. □

Remark 4. *Note that $\overline{v}^1$ should be chosen with care as it affects the size of ROA inner-approximations directly: decreasing $(\overline{v}^1 - v_*^1)$ gives rise to sharper local sector bounds, which is beneficial on ROA estimation, but also restricts the region where ROA inner-approximations lie in; increasing $(\overline{v}^1 - v_*^1)$ leads to a larger region that contains ROA inner-approximations, but also provides looser local sector bounds. A possible way of choosing $\overline{v}^1$ is to parameterize $(\overline{v}^1 - v_*^1)$ as $\overline{v}^1 - v_*^1 = \delta_v \times 1_{n_1 \times 1}$ with $\delta_v \in \mathbb{R}_{++}$, grid the interval $[0, \overline{\delta}_v]$[*] where δ_v lies in, inner-approximate the ROA on the grid, and choose δ_v that leads to the largest inner-approximation.*

Remark 5. *In the dissertation, the NN controller is assumed to be state-feedback. For the output-feedback case, i.e., $u = \pi(Cx)$, where $C \in \mathbb{R}^{n_y \times n_G}$, the stability analysis can be performed similarly, using a new N_{vx} defined as $N_{vx} := \begin{bmatrix} W^1 C \\ 0_{(n_2+\ldots+n_\ell)\times n_G} \end{bmatrix}$.*

6.2 Robust Stability Analysis

Problem Formulation

Consider the uncertain feedback system in Figure 6.5, consisting of an uncertain plant $F_u(G, \Delta)$ and a NN controller π as defined by (6.2). The uncertain plant $F_u(G, \Delta)$ is an interconnection of a nominal plant G and a perturbation Δ. The nominal plant G is defined by the following equations:

$$x(k+1) = A_G\, x(k) + B_{G1}\, q(k) + B_{G2}\, u(k), \tag{6.23a}$$

$$p(k) = C_G\, x(k) + D_{G1}\, q(k) + D_{G2}\, u(k), \tag{6.23b}$$

where $x(k) \in \mathbb{R}^{n_G}$ is the state, $u(k) \in \mathbb{R}^{n_u}$ is the control input, $p(k) \in \mathbb{R}^{n_p}$ and $q(k) \in \mathbb{R}^{n_q}$ are the input and output of Δ, $A_G \in \mathbb{R}^{n_G \times n_G}$, $B_{G1} \in \mathbb{R}^{n_G \times n_q}$, $B_{G2} \in \mathbb{R}^{n_G \times n_u}$, $C_G \in \mathbb{R}^{n_p \times n_G}$, $D_{G1} \in \mathbb{R}^{n_p \times n_q}$, and $D_{G2} \in \mathbb{R}^{n_p \times n_u}$. The perturbation is a bounded, causal operator $\Delta : \ell_{2e}^{n_p} \to \ell_{2e}^{n_q}$. The nominal plant G and perturbation Δ form the interconnection $F_u(G, \Delta)$ through the constraint

$$q(\cdot) = \Delta(p(\cdot)). \tag{6.24}$$

Denote the set of perturbations to be considered as $\mathcal{S}$.

Assumption 7. *In this section, we assume (i) the equilibrium point $(x_*, u_*, v_*, w_*, p_*, q_*)$ of the feedback system is at the origin, and (ii) $0 = \Delta(0)$ for all $\Delta \in \mathcal{S}$. Note that Δ is modeled as an operator mapping inputs to outputs. If Δ has an internal state then there is an implicit assumption that it has zero initial condition.*

[*] $\overline{\delta}_v$ is the largest value such that (6.19) and (6.20) stay feasible.

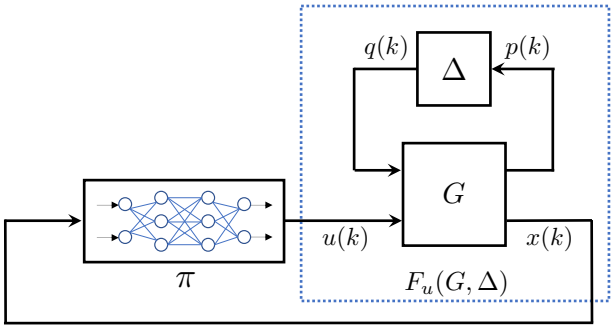

Figure 6.5: Feedback system with uncertain plant $F_u(G, \Delta)$ and NN controller π

Let $\chi(k; x_0, \Delta)$ denote the solution to the feedback system of $F_u(G, \Delta)$ and π with $\Delta \in \mathcal{S}$ at time k from the initial condition $x(0) = x_0$[†]. Define the robust ROA associated with x_* as follows.

Definition 19. *The robust ROA of the feedback system with the uncertain plant $F_u(G, \Delta)$ and NN π is defined as:*

$$\mathcal{R} := \{x_0 \in \mathbb{R}^{n_G} : \lim_{k\to\infty} \chi(k; x_0, \Delta) = x_* \ \forall \Delta \in \mathcal{S}\}. \tag{6.25}$$

The objective is to prove the uncertain feedback system is asymptotically stable and, if so, to find the largest estimate of the robust ROA using an ellipsoidal inner approximation.

Robust Lyapunov Condition

The perturbation can represent various types of uncertainty [13], [18], including saturation, unmodeled dynamics, and slope-restricted nonlinearities. The perturbation Δ is characterized with an integral quadratic constraint (IQC). In this chapter, we will focus on the use of time domain hard IQCs, which is defined in Definition 6. Note that it is possible to also incorporate time domain soft IQCs [14, 57, 68, 73].

The notation $\Delta \in \text{HardIQC}(\Psi_\Delta, M_\Delta)$ indicates that Δ satisfies the hard IQC defined by a filter Ψ_Δ and a matrix M_Δ. The filter Ψ_Δ is an LTI filter defined by (2.15) with zero initial condition $\psi(0) = 0$, where $\psi \in \mathbb{R}^{n_\psi}$ is the state. By $(p_*, q_*) = 0$ from Assumption 7, the equilibrium state ψ_* of Ψ_Δ is also zero. The filter Ψ_Δ is applied to the input p, and output q of Δ, and the input-output relationship of Δ is characterized by the IQC (2.16) imposed on r, the output of Ψ_Δ, where $r := \Psi_\Delta \left[\begin{smallmatrix} p \\ q \end{smallmatrix}\right]$. Therefore, the precise relation (6.24), for analysis, is replaced by the IQC on r associated with M_Δ:

$$\sum_{k=0}^{N} r(k)^\top M_\Delta r(k) \geq 0, \ \forall N \geq 0, \ p \in \ell_{2e}^{n_p}, \text{ and } q = \Delta(p). \tag{6.26}$$

[†]An input/output model is used for the perturbation Δ so that its internal state and initial condition is not explicitly considered.

The QC proposed in Lemma 2 is a special instance of a time-domain IQCs. Specifically, Lemma 2 defines a QC that holds at each time step k and hence the inequality also holds summing over any finite horizons. This is referred to as the offset local sector IQC.

Let $\zeta := \left[\begin{smallmatrix} x \\ \psi \end{smallmatrix}\right] \in \mathbb{R}^{n_\zeta}$ define the extended state vector, $n_\zeta = n_G + n_\psi$, whose dynamics are

$$\zeta(k+1) = \mathcal{A}\,\zeta(k) + \mathcal{B}\begin{bmatrix} q(k) \\ u(k) \end{bmatrix} \tag{6.27a}$$

$$r(k) = \mathcal{C}\,\zeta(k) + \mathcal{D}\begin{bmatrix} q(k) \\ u(k) \end{bmatrix} \tag{6.27b}$$

$$u(k) = \pi(x(k)) \tag{6.27c}$$

where the state-space matrices are

$$\mathcal{A} = \begin{bmatrix} A_G & 0 \\ B_{\psi 1} C_G & A_\psi \end{bmatrix},\ \mathcal{B} = \begin{bmatrix} B_{G1} & B_{G2} \\ B_{\psi 1} D_{G1} + B_{\psi 2} & B_{\psi 1} D_{G2} \end{bmatrix},$$
$$\mathcal{C} = \begin{bmatrix} D_{\psi 1} C_G & C_\psi \end{bmatrix},\ \mathcal{D} = \begin{bmatrix} D_{\psi 1} D_{G1} + D_{\psi 2} & D_{\psi 1} D_{G2} \end{bmatrix}.$$

Let $\zeta_* := \left[\begin{smallmatrix} x_* \\ \psi_* \end{smallmatrix}\right] = 0$ define the equilibrium point of the extended system (6.27). Since IQCs implicitly constrain the input p of the extended system (6.27), the response of the extended system subject to IQCs "covers" the behaviors of the original uncertain feedback system. The following theorem provides a method for inner-approximating the robust ROA by performing analysis on the extended system subject to IQCs.

Theorem 11. *Consider the feedback system of an uncertain plant $F_u(G, \Delta)$ in (6.23)–(6.24), and the NN π in (6.2) with zero equilibrium point $(\zeta_*, u_*, v_*, w_*, p_*, q_*)$. Assume $\Delta \in IQC(\Psi_\Delta, M_\Delta)$ with Ψ_Δ and M_Δ given. Let $\bar{v}^1 \in \mathbb{R}^{n_1}$, $\underline{v}^1 := -\bar{v}^1$, and $\alpha_\phi, \beta_\phi \in \mathbb{R}^{n_\phi}$ be given vectors satisfying Property 1 for the NN, and define matrices*

$$R_V = \left[\begin{array}{ccc} I_{n_\zeta} & 0 & 0 \\ \hline 0 & 0 & I_{n_q} \\ N_{u\zeta} & N_{uw} & 0 \end{array}\right],\ N_{u\zeta} = [N_{ux}, 0_{n_u \times n_\psi}],$$
$$R_\phi = \begin{bmatrix} N_{v\zeta} & N_{vw} & 0 \\ 0 & I_{n_\phi} & 0 \end{bmatrix},\ N_{v\zeta} = [N_{vx}, 0_{n_\phi \times n_\psi}],$$
$$\mathcal{W}_i^1 = \begin{bmatrix} W_i^1 & 0_{1 \times n_\psi} \end{bmatrix},\ W_i^1 \text{ is the } i^{th} \text{ row of } W^1.$$

If there exists a matrix $P \in \mathbb{S}_{++}^{n_\zeta}$, and vector $\lambda \in \mathbb{R}^{n_\phi}$ with $\lambda \geq 0$ such that

$$R_V^\top \begin{bmatrix} \mathcal{A}^\top P \mathcal{A} - P & \mathcal{A}^\top P \mathcal{B} \\ \mathcal{B}^\top P \mathcal{A} & \mathcal{B}^\top P \mathcal{B} \end{bmatrix} R_V + R_\phi^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi R_\phi$$
$$+ R_V^\top \begin{bmatrix} \mathcal{C} & \mathcal{D} \end{bmatrix}^\top M_\Delta \begin{bmatrix} \mathcal{C} & \mathcal{D} \end{bmatrix} R_V < 0, \tag{6.28a}$$

$$\begin{bmatrix} (\bar{v}_i^1)^2 & \mathcal{W}_i^1 \\ \mathcal{W}_i^{1\top} & P \end{bmatrix} \geq 0,\ i = 1, \cdots, n_1, \tag{6.28b}$$

then: (i) the feedback system comprising $F_u(G,\Delta)$ and π is locally stable around x_* for any $\Delta \in IQC(\Psi_\Delta, M_\Delta)$, and (ii) the intersection of $\mathcal{E}(P,\zeta_*)$ with the hyperplane $\psi = 0$, i.e. $\mathcal{E}(P_x, x_*)$ where $P_x \in \mathbb{R}^{n_G \times n_G}$ is the upper left block of P, is an inner-approximation to the robust ROA.

Proof. As in the proof of Theorem 10, feasibility of (6.28b) implies that if $\zeta(k) \in \mathcal{E}(P,\zeta_*)$ then $v^1(k) \in [\underline{v}^1, \bar{v}^1]$ and hence the offset local sectors conditions are valid. Since the LMI in (6.28a) is strict, there exists $\epsilon > 0$ such that left/right multiplication of the LMI by $[(\zeta(k)-\zeta_*)^\top, (w_\phi(k)-w_*)^\top, (q(k)-q_*)^\top]$ and its transpose yields:

$$\begin{bmatrix} \star \end{bmatrix}^\top \begin{bmatrix} \mathcal{A}^\top P \mathcal{A} - P & \mathcal{A}^\top P \mathcal{B} \\ \mathcal{B}^\top P \mathcal{A} & \mathcal{B}^\top P \mathcal{B} \end{bmatrix} \begin{bmatrix} \frac{\zeta(k)-\zeta_*}{q(k)-q_*} \\ u(k)-u_* \end{bmatrix} + \begin{bmatrix} \star \end{bmatrix}^\top \begin{bmatrix} \mathcal{C} & \mathcal{D} \end{bmatrix}^\top M_\Delta \begin{bmatrix} \mathcal{C} & \mathcal{D} \end{bmatrix} \begin{bmatrix} \frac{\zeta(k)-\zeta_*}{q(k)-q_*} \\ u(k)-u_* \end{bmatrix}$$
$$+ \begin{bmatrix} \star \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi(k)-v_* \\ w_\phi(k)-w_* \end{bmatrix} \leq -\epsilon |\zeta(k)-\zeta_*|^2.$$

Define the Lyapunov function $V(\zeta) := (\zeta-\zeta_*)^\top P(\zeta-\zeta_*)$, and use (6.27) to show:

$$V(\zeta(k+1)) - V(\zeta(k)) + r(k)^\top M_\Delta r(k) + \begin{bmatrix} \star \end{bmatrix}^\top \Psi_\phi^\top M_\phi(\lambda) \Psi_\phi \begin{bmatrix} v_\phi(k)-v_* \\ w_\phi(k)-w_* \end{bmatrix} \leq -\epsilon |\zeta(k)-\zeta_*|^2.$$

Sum this inequality from $k=0$ to any finite time $N \geq 0$. The third and fourth term on the left side will be ≥ 0 by the local sector conditions and the IQC. This yields:

$$V(\zeta(N+1)) - V(\zeta(0)) \leq -\sum_{k=0}^{N} \epsilon |\zeta(k)-\zeta_*|^2.$$

Thus if $\zeta(0) \in \mathcal{E}(P,\zeta_*)$ then $\zeta(k) \in \mathcal{E}(P,\zeta_*)$ for all $k \geq 0$. Moreover, this inequality implies that $\zeta(N) \to \zeta_*$ as $N \to \infty$. The initial condition for the virtual filter is $\psi(0) = 0$ so that $\zeta(0) \in \mathcal{E}(P,\zeta_*)$ is equivalent to $x(0) \in \mathcal{E}(P_x, x_*)$. Hence $\mathcal{E}(P_x, x_*)$ is an inner approximation for the ROA. □

For a particular perturbation Δ there is typically a class of valid time-domain IQCs defined by a fixed filter Ψ_Δ and a matrix M_Δ drawn from a constraint set $\mathcal{M}_\Delta$. Therefore when formulating an optimization problem, along with P and λ, we can treat $M_\Delta \in \mathcal{M}_\Delta$ as an additional decision variable to reduce conservatism. In this dissertation, the set $\mathcal{M}_\Delta$ is restricted to one that is described by LMIs [18]. Using $\text{trace}(P_x)$ as the cost function to minimize along with the LMIs developed before, we have the following optimization to compute the "largest" ROA inner-approximation:

$$\begin{aligned} \min_{P,\lambda,M_\Delta} \quad & \text{trace}(P_x) \\ \text{s.t. } & \text{(6.28a) and (6.28b) hold,} \end{aligned} \tag{6.29}$$

which is convex in (P,λ,M_Δ). The strict inequality in (6.28a) can be enforced by either replacing < 0 with $\leq -\epsilon I$ with $\epsilon = 1 \times 10^{-6}$, or solving (6.29) with a non-strict inequality ≤ 0, and checking if the constraint is active afterwards.

IQCs for Combined Activation Functions ϕ

Now that we have the general framework that merges Lyapunov theory with IQCs, we will revisit the problem of describing the activation functions ϕ using more general tools. Recall that offset local sector QCs have been used in Sections 7.2 and 6.2 to bound activation functions ϕ. However, these local sectors fail to incorporate slope bounds of ϕ. In this subsection, in addition to local sectors, we will use off-by-one IQCs [61] to capture the slope information of ϕ to achieve less conservative ROA inner-approximations.

Besides the local sector bound α_ϕ, β_ϕ, the bounds $\underline{v}, \overline{v}$ on activation input v_ϕ can also be used to compute the local slope bounds $[m_\phi, L_\phi]$ of ϕ, with $m_\phi, L_\phi \in \mathbb{R}^{n_\phi}$. For example, $\phi_i(v_{\phi,i}) := \tanh(v_{\phi,i})$ restricted to the interval $[\underline{v}_i, \overline{v}_i]$ for $i = 1, ..., n_\phi$ satisfies the local slope bound $[m_{\phi,i}, L_{\phi,i}]$ with $m_{\phi,i} := \min\left(\frac{d\tanh(v_i)}{dv_i}|_{v_i=\underline{v}_i}, \frac{d\tanh(v_i)}{dv_i}|_{v_i=\overline{v}_i}\right)$, and $L_{\phi,i} := 1$. If $w_\phi = \phi(v_\phi)$ with $v_\phi(k) \in [\underline{v}, \overline{v}]$, then ϕ also satisfies the hard IQC defined by $(\Psi_{\text{off}}, M_{\text{off}})$, where

$$\Psi_{\text{off}} := \left[\begin{array}{c|cc} 0_{n_\phi} & -\text{diag}(L_\phi) & I_{n_\phi} \\ \hline I_{n_\phi} & \text{diag}(L_\phi) & -I_{n_\phi} \\ 0_{n_\phi} & -\text{diag}(m_\phi) & I_{n_\phi} \end{array}\right],$$

$$M_{\text{off}}(\eta) := \begin{bmatrix} 0_{n_\phi} & \text{diag}(\eta) \\ \text{diag}(\eta) & 0_{n_\phi} \end{bmatrix}, \text{ for all } \eta \in \mathbb{R}^{n_\phi} \text{ with } \eta \geq 0.$$

This is the so-called "off-by-one" IQC [61], which is a special instance of the Zames-Falb IQC [74, 75]. It provides constraints that relate the activation at different time instances, e.g. between $\phi_i(v_{\phi,i}(k))$ and $\phi_i(v_{\phi,i}(k+1))$ for any $i = 1, \ldots, n_\phi$.

The analysis on the feedback system of $F_u(G, \Delta)$ and π can be instead performed on the extended system made up by G, Ψ_Δ and Ψ_{off} with additional constraints that $\Delta \in \text{IQC}(\Psi_\Delta, M_\Delta)$, and ϕ satsifies the offset local sector and $\phi \in \text{IQC}(\Psi_{\text{off}}, M_{\text{off}})$. However, since Ψ_{off} introduces a number of n_ϕ states to the extended system, the size of the corresponding Lyapunov matrix P will increase from $\mathbb{S}_{++}^{n_x+n_\psi}$ to $\mathbb{S}_{++}^{n_x+n_\psi+n_\phi}$, which leads to longer computation time. The effectiveness of the off-by-one IQC is demonstrated in Section 6.3.

6.3 Numerical Examples

In the following examples, the optimization (6.29) is solved using MOSEK with CVX. The code is available at `https://github.com/heyinUCB/Stability-Analysis-using-Quadratic-Constraints-for-Systems-with-Neural-Network-Controllers`.

Inverted pendulum

Consider the nonlinear inverted pendulum example with mass $m = 0.15$ kg, length $l = 0.5$ m, and friction coefficient $\mu = 0.5$ Nms/rad. The dynamics are:

$$\ddot{\theta}(t) = \frac{mgl\sin(\theta(t)) - \mu\dot{\theta}(t) + \text{sat}(u(t))}{ml^2}, \tag{6.30}$$

where θ is the angular position (rad) and u is the control input (Nm). The plant state is $x = [\theta, \dot{\theta}]$. The saturation function is defined as $\text{sat}(u) = \text{sgn}(u)\min(|u|, u_{\max})$, with $u_{\max} = 0.7$ Nm. The controller π is obtained through a reinforcement learning process using policy gradient [76–78]. During training, the agent decision making process is characterized by a probability: $u(k) \sim Pr(u(k) = u \mid x(k) = x)$ for all $u \in \mathbb{R}$ and $x \in \mathbb{R}^2$ where the probability is a Gaussian distribution with mean $\pi(x(k))$, and standard deviation σ. After training, the policy mean π is used as the deterministic controller $u(k) = \pi(x(k))$. The controller π is parameterized by a 2-layer, feedforward NN with $n_1 = n_2 = 32$ and tanh as the activation function for both layers. The biases in the NN are set to zero during training to ensure that the equilibrium point is $x_* = 0$ and $u_* = 0$. The dynamics used for training are the discretized version of (6.30) with sampling time $dt = 0.02$ s.

We rearrange (6.30) into the form:

$$\ddot{\theta}(t) = \frac{-mglq(t) + mgl\theta(t) - \mu\dot{\theta}(t) + \text{sat}(u(t))}{ml^2}, \tag{6.31a}$$

$$q(t) = \Delta(\theta(t)) := \theta(t) - \sin(\theta(t)). \tag{6.31b}$$

The static nonlinearity $\Delta(\theta) = \theta - \sin(\theta)$ is slope-restricted, and sector bounded. If we assume that $\theta(k) \in [\underline{\theta}, \bar{\theta}]$ with $\bar{\theta} = -\underline{\theta} = 0.73$, then the nonlinearity is slope-restricted in $[0, 0.2548]$, and sector bounded in $[0, 0.087]$. We also assume that $v^1 \in [\underline{v}^1, \bar{v}^1]$ with $\bar{v}^1 = -\underline{v}^1 = \delta_v \times 1_{32\times 1}$ using $\delta_v = 0.1$. Both assumptions are verified using the ROA inner-approximation. Two types of IQCs are used to characterize the nonlinearity $\Delta(\cdot)$: an off-by-one IQC to capture the slope information, and a local sector IQC to express the local sector bound. Only the local sector IQC is used to characterize the activation functions ϕ. The saturation function is static and can also be described using a local sector bound. Let $\bar{u}$ be the largest possible control command from π induced from the assumption that $v^1 \in [\underline{v}^1, \bar{v}^1]$. Then the saturation function satisfies the local sector $[\alpha, \beta]$, where $\alpha := \frac{u_{\max}}{\bar{u}}$ and $\beta := 1$.

Figure 6.6 shows the boundaries for the sets $\{x : \underline{v}^1 \le v^1 \le \bar{v}^1\}$ and $\{x : \underline{\theta} \le \theta \le \bar{\theta}\}$ with orange and brown lines, the ROA inner-approximation with a blue ellipsoid, and the phase portrait of the closed-loop system, with green and red curves representing trajectories inside and outside the ROA.

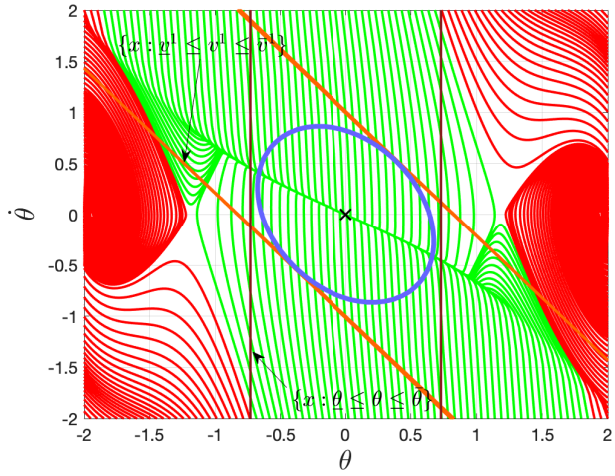

Figure 6.6: A ROA inner-approximation of the inverted pendulum

Vehicle lateral control

Consider the vehicle lateral dynamics from [79]:

$$\begin{bmatrix} \dot{e} \\ \ddot{e} \\ \dot{e}_\theta \\ \ddot{e}_\theta \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{C_{\alpha f}+C_{\alpha r}}{mU} & -\frac{C_{\alpha f}+C_{\alpha r}}{m} & \frac{aC_{\alpha f}-bC_{\alpha r}}{mU} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{aC_{\alpha f}-bC_{\alpha r}}{I_z U} & -\frac{aC_{\alpha f}-bC_{\alpha r}}{I_z} & \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z U} \end{bmatrix} \begin{bmatrix} e \\ \dot{e} \\ e_\theta \\ \dot{e}_\theta \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{C_{\alpha f}}{m} \\ 0 \\ -\frac{aC_{\alpha f}}{I_z} \end{bmatrix} u + \begin{bmatrix} 0 \\ \frac{aC_{\alpha f}-bC_{\alpha r}}{m} - U^2 \\ 0 \\ \frac{a^2C_{\alpha f}+b^2C_{\alpha r}}{I_z} \end{bmatrix} c \tag{6.32}$$

where e is the perpendicular distance to the lane edge (m), and e_θ is the angle between the tangent to the straight section of the road and the projection of the vehicle's longitudinal axis (rad). Let $x = [e, \dot{e}, e_\theta, \dot{e}_\theta]^\top$ denote the plant state. The control u is the steering angle of the front wheel (rad), the disturbance c is the road curvature (1/m), and the parameters are: longitudinal velocity $U = 28$ m/s, front cornering stiffness $C_{\alpha f} = -1.232 \times 10^5$ N/rad, rear cornering stiffness $C_{\alpha r} = -1.042 \times 10^5$ N/rad, mass $m = 1.67 \times 10^3$ kg, moment of inertia $I_z = 2.1 \times 10^3$ kg/m^2, distances from vehicle center of gravity to front axle $a = 0.99$ m and rear axle $b = 1.7$ m.

Again, the controller π is obtained using policy gradient, and is parameterized by a 2-layer, feedforward NN, with $n_1 = n_2 = 32$ and tanh as the activation function for both layers. The training process uses a discretized version of (6.32) with sampling time $dt = 0.02$ s and draws the curvature $c(k)$ at each time step from an interval $[-1/200, 1/200]$. The control command derived from $u(k) = \pi(x(k))$ enters the vehicle dynamics through a saturation function $\mathrm{sat}(\cdot)$ with $u_{\max} = \pi/6$. Let $u_{\mathrm{sat}} := \mathrm{sat}(\pi(x))$ define the saturated control signal.

The analysis is performed for a constant curvature $c \equiv 0$, resulting in a zero equilibrium state $x_* = 0$. In the analysis problem, on top of saturation, we also add a norm-bounded LTI uncertainty $\Delta_{\mathrm{LTI}} \in \mathbb{RH}_\infty$ with $\|\Delta_{\mathrm{LTI}}\|_\infty \le 0.1$ to the control input. This is used to assess the robustness of the NN controller against actuator uncertainty. As shown in Figure 6.7,

the actual input to the vehicle dynamics is

$$u_{\text{pert}}(k) = u_{\text{sat}}(k) + q(k), \text{ and } q(\cdot) = \Delta_{\text{LTI}}(u_{\text{sat}}(\cdot)).$$

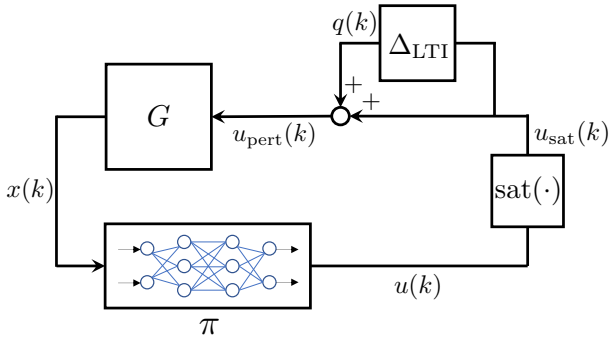

Figure 6.7: Uncertain vehicle system with actuator uncertainty

It is assumed that $v^1 \in [\underline{v}^1, \bar{v}^1]$, where $\bar{v}^1 = -\underline{v}^1 = \delta_v \times 1_{32\times 1}$ with $\delta_v = 0.6$. To show effectiveness of the off-by-one IQC, two experiments were carried out: one with only local sector IQC to describe ϕ, and one with both local sector and off-by-one IQCs. The achieved trace(P_x) for the two experiments are 4.4 and 2.9, respectively. Moreover, the achieved $\det(P_x^{-1})$ (proportional to the volume) for the experiments are 3.2×10^5, and 1.1×10^6, respectively. Therefore, with the help of off-by-one IQC to sharpen the description of ϕ, the second experiment achieves a larger ROA inner-approximation. It is also important to note that thanks to the off-by-one IQC, the SDP is able to tolerate looser local sector bounds. The largest value of δ_v such that the SDP is feasible is 0.67 for the first experiment, and 1.4 for the second experiment.

Figure 6.8 shows slices of the ROA inner-approximation from the second experiment on the e–$\dot{e}$ and e_θ–$\dot{e}_\theta$ spaces. Specifically, these are intersections of $\mathcal{E}(P_x, x_*)$ with the hyperplanes $(e_\theta, \dot{e}_\theta) = (e_{\theta*}, \dot{e}_{\theta*})$ and $(e, \dot{e}) = (e_*, \dot{e}_*)$, respectively, where $x_* = [e_*, \dot{e}_*, e_{\theta*}, \dot{e}_{\theta*}]^\top$. The slices are shown with blue ellipsoids. The boundary of the polytopic set $\{x : \underline{v}^1 \le v^1 \le \bar{v}^1\}$ is shown with the orange lines. The brown crosses represent the zero equilibrium state x_*.

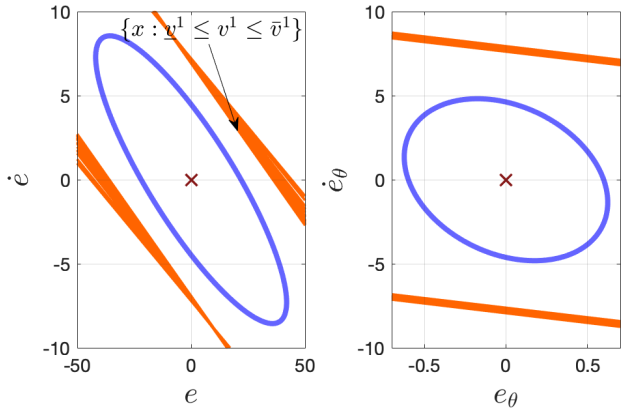

Figure 6.8: ROA inner-approximation on the e–$\dot{e}$ and e_θ–$\dot{e}_\theta$ spaces using both local sector and off-by-one IQCs with $\delta_v = 0.6$.

6.4 Chapter Summary

This chapter presents a method to analyze stability and to inner-approximate the region of attraction of equilibria in feedback systems with NN controllers. First, LTI plants were analyzed using Lyapunov theory and local sector QCs for bounding nonlinear activation functions. Second, the results were extended to uncertain plants with perturbations described by IQCs, such as nonlinearities and unmodeled dynamics. The extended result allows for the use of off-by-one IQCs to refine the description of activation functions. Finally, the method was illustrated on a nonlinear inverted pendulum example, and uncertain vehicle lateral dynamics with stabilizing NN controllers obtained using policy gradient.

Chapter 7

Imitation Learning With Stability and Safety Guarantees Using Quadratic Constraints

Imitation learning (IL) is a class of methods that learns a policy to attain a control goal consistent with expert demonstrations [80, 81]. Used in tandem with deep neural networks (NNs), IL presents unique advantages, including a substantial increase in sample efficiency compared to reinforcement learning (RL) [82], and wide applicability to domains where the reward model is not accessible or on-policy data is difficult/unsafe to acquire [80]. While IL is closely related to supervised learning as it trains a mapping from observations to actions [83], a key difference is the ensuing deployment of the learned policy under dynamics, which consequently raises the issue of closed-loop stability. This problem naturally falls within the realm of robust control, which provides rigorous analysis of stability for linear or nonlinear systems [1]; however, a major technical challenge is to derive nonconservative guarantees for highly nonlinear policies such as NNs that can be also tractably incorporated into the learning process.

This chapter tackles this issue and presents a method to learn NN controllers with stability and safety guarantees through IL. We first derive convex stability and safety conditions by merging Lyapunov theory with local sector quadratic constraints (QCs) to describe the activation functions in the NN. Then we incorporate these constraints in the IL process that minimizes the IL loss, and maximizes the volume of the region of attraction associated with the NN controller. Finally, we propose an alternating direction method of multipliers (ADMM) based method to solve the IL problem.

The method of using QCs to characterize the activation functions has been proposed in [62], and applied to Lipschitz constant estimation for NNs [63], reachability analysis of NN controlled systems [11], and stability analysis of NN controlled systems [84–86]. Reference [64] proposes an ADMM based robust NN training method that minimizes the training loss, and the Lipschitz constant of NNs simultaneously. Reference [87] formulates convex sets of recurrent NN with bounded incremental ℓ_2 gain using incremental QCs.

Compared to existing works, our approach makes the following contributions. First, to the best of our knowledge, this is the first attempt to ensure stability of IL based on deep NNs. The stability condition from Chapter 6 is nonconvex and thus computationally intractable for NN control synthesis; here we convexify this constraint (using loop transformation) for its efficient enforcement in the learning process. Notably, a well-known challenge in IL is the existence of suboptimal demonstrations. As demonstrated in the case studies, while the proposed approach can train a policy that imitates the expert demonstrations, it can potentially improve local stability over suboptimal expert policies, thus enhance the robustness of IL.

In this chapter, we first present a loop transformation based method to convexify the stability condition from Chapter 6. Then we formulate the safe IL problem with the convexified stability condition. Finally we illustrate the method on an inverted pendulum example, and an aircraft example.

7.1 Problem Formulation

In this chapter, we again consider the feedback system consisting of a plant G and state-feedback controller π as shown in Figure 6.1. Similar to Chapter 6, we assume the plant G is a linear, time-invariant (LTI) system defined by (6.1), and the controller $\pi : \mathbb{R}^{n_G} \to \mathbb{R}^{n_u}$ is an ℓ-layer, feedforward neural network (NN) defined by (6.2) associated with weight matrices $W^i \in \mathbb{R}^{n_i \times n_{i-1}}$, bias vectors $b^i \in \mathbb{R}^{n_i}$, and activation functions $\phi^i : \mathbb{R}^{n_i} \to \mathbb{R}^{n_i}$. We assume the activation φ is identical in all layers; this can be relaxed with minor changes to the notation. Finally, we assume $x(k)$ is constrained to a set $X \subset \mathbb{R}^{n_G}$, which is referred to as the "safety condition". This state constraint set is assumed to be a polytope symmetric around the origin:

$$X = \{x \in \mathbb{R}^{n_G} : -h \leq Hx \leq h,\ h \geq 0\}, \tag{7.1}$$

where $H \in \mathbb{R}^{n_X \times n_G}$, and $h \in \mathbb{R}^{n_X}$.

Remark 6. *Note that the control constraint is not considered in this dissertation. However, if the control constraint set is a hypercube, it can be addressed by scaling the dynamics* (6.1) *so that the control constraint set becomes* $[-1_{n_u \times 1}, 1_{n_u \times 1}]$*, and applying* tanh *elementwise to the output layer* (6.2c)*.*

Assumption 8. *We assume the equilibrium point of the closed-loop system is at the origin* $x_* = 0_{n_G \times 1}$ *with input* $u_* = 0_{n_u \times 1}$*. To ensure this assumption holds, we restrict ourselves to zero bias terms:* $b^i = 0_{n_i \times 1}$*, for* $i = 1, ..., \ell + 1$*, and we also assume that the activation* φ *satisfies* $\varphi(0) = 0$*.*

Given state and control data pairs from the expert demonstrations, our goal is to learn a NN controller from the data to reproduce the demonstrated behavior, while guaranteeing the system trajectories under the control of the learned NN controller satisfy the safety condition

($x(k) \in X \ \forall k \geq 0$), and are able to converge to the equilibrium point if they start from the ROA associated with the learned NN controller.

7.2 Stability Condition for NN Controlled Systems

In this section, we treat the parameters of the NN controller as fixed and derive the safety and local stability conditions of the NN-controlled LTI systems.

We first isolate the nonlinear activation functions from the linear operations of the NN. By Assumption 8 that $b^i = 0_{n_i \times 1}$, the NN controller π defined in (6.2) can be rewritten as:

$$\begin{bmatrix} u(k) \\ v_\phi(k) \end{bmatrix} = N \begin{bmatrix} x(k) \\ w_\phi(k) \end{bmatrix} \tag{7.2a}$$

$$w_\phi(k) = \phi(v_\phi(k)), \tag{7.2b}$$

where v_ϕ, and w_ϕ are defined by (6.7). The matrix N is redefined as follows, where the vertical and horizontal bars partition N compatibly with the inputs (x, w_ϕ) and outputs (u, v_ϕ):

$$N = \left[\begin{array}{c|cccc} 0 & 0 & 0 & \cdots & W^{\ell+1} \\ \hline W^1 & 0 & \cdots & 0 & 0 \\ 0 & W^2 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & W^\ell & 0 \end{array}\right] := \begin{bmatrix} N_{ux} & N_{uw} \\ N_{vx} & N_{vw} \end{bmatrix}.$$

This decomposition of the NN, depicted in Figure 7.1, isolates the nonlinearities ϕ in preparation for the stability analysis.

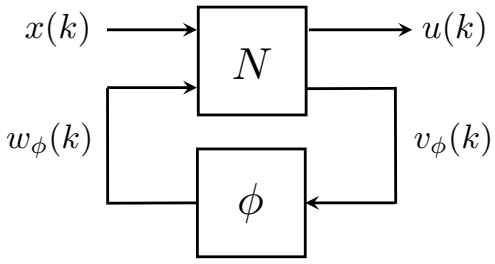

Figure 7.1: NN representation to isolate the nonlinearities ϕ.

The equilibrium state $x_* = 0_{n_G \times 1}$ can be propagated through the NN. This yields the equilibrium values $v_* = w_* = 0_{n_\phi \times 1}$ as well as the equilibrium control command $u_* = 0_{n_u \times 1}$. Therefore, the activation functions ϕ can be described by the following local QC centered around the origin $(v_*, w_*) = (0, 0)$, which is a special instance of the offset local QC in Lemma 2.

Lemma 3. *Let α_ϕ, β_ϕ, $\underline{v}$, $\bar{v} \in \mathbb{R}^{n_\phi}$ be given with $\alpha_\phi \leq \beta_\phi$, and $\underline{v} \leq 0 \leq \bar{v}$. Suppose that ϕ satisfies the local sector $[\alpha_\phi, \beta_\phi]$ element-wise for all $v_\phi \in [\underline{v}, \bar{v}]$. For any $\lambda \in \mathbb{R}^{n_\phi}$ with $\lambda \geq 0$, it holds that*

$$\begin{bmatrix} v_\phi \\ w_\phi \end{bmatrix}^\top \begin{bmatrix} -2A_\phi B_\phi \Lambda & (A_\phi + B_\phi)\Lambda \\ (A_\phi + B_\phi)\Lambda & -2\Lambda \end{bmatrix} \begin{bmatrix} v_\phi \\ w_\phi \end{bmatrix} \geq 0$$
$$\forall v_\phi \in [\underline{v}, \bar{v}],\ w_\phi = \phi(v_\phi), \tag{7.3}$$

where $A_\phi = diag(\alpha_\phi)$, $B_\phi = diag(\beta_\phi)$, and $\Lambda = diag(\lambda)$.

Proof. For any $v_\phi \in \mathbb{R}^{n_\phi}$ and $w_\phi = \phi(v_\phi)$: left-hand side of (7.3) $= \sum_{i=1}^{n_\phi} \lambda_i (w_i - \alpha_i v_i) \cdot (\beta_i v_i - w_i)$. If $v_\phi \in [\underline{v}, \bar{v}]$ then each term in the sum is non-negative by the local sector constraints and $\lambda \geq 0$. □

In order to apply the local sector bounds in the stability analysis, we must first compute the bounds $\underline{v}, \overline{v} \in \mathbb{R}^{n_\phi}$ on the activation input v_ϕ. The first step is to find the smallest hypercube that bounds the state constraint set: $X \subseteq \{x : \underline{x} \leq x \leq \overline{x}\}$. Therefore, w^0 (defined in (6.2a)) is bounded by $\underline{w}^0 = \underline{x}$ and $\overline{w}^0 = \overline{x}$. Define $c = \frac{1}{2}(\overline{w}^0 + \underline{w}^0)$, $r = \frac{1}{2}(\overline{w}^0 - \underline{w}^0)$, and denote $y^\top$ as the i^{th} row of W^1. Then the first activation input $v^1 = W^1 w^0$ is bounded by $[\underline{v}^1, \overline{v}^1]$, where $\overline{v}_i^1 = y^\top c + \sum_{j=1}^{n_0} |y_j r_j|$, and $\underline{v}_i^1 = y^\top c - \sum_{j=1}^{n_0} |y_j r_j|$. If the activation functions ϕ^1 are monotonically non-decreasing, then the first activation output w^1 is bounded by $\underline{w}^1 = \phi^1(\underline{v}^1)$ and $\overline{w}^1 = \phi^1(\overline{v}^1)$. This process can be propagated through all layers of the NN to obtain the bounds $\underline{v}, \overline{v} \in \mathbb{R}^{n_\phi}$ for the activation input v_ϕ. The remainder of the dissertation will assume the local sector bounds have been computed as briefly summarized in the following property.

Property 2. *Let the state constraint set X and its corresponding activation input bounds $\underline{v}, \overline{v}$ be given. There exist α_ϕ, β_ϕ such that ϕ satisfies the local sector for all $v_\phi \in [\underline{v}, \overline{v}]$.*

Lyapunov Condition

This section uses a Lyapunov function and the local sector to compute an inner approximation for the ROA of the feedback system of G and π.

Theorem 12. *Consider the feedback system of plant G in (6.1) and NN π in (6.2) with equilibrium point $x_* = 0_{n_G \times 1}$, and the state constraint set X. Let $\bar{v}, \underline{v}, \alpha_\phi, \beta_\phi \in \mathbb{R}^{n_\phi}$ be given vectors satisfying Property 1 for the NN and the set X. Denote the i^{th} row of the matrix H by $H_i^\top$ and define*

$$R_V := \begin{bmatrix} I_{n_G} & 0_{n_G \times n_\phi} \\ N_{ux} & N_{uw} \end{bmatrix}, \text{ and } R_\phi := \begin{bmatrix} N_{vx} & N_{vw} \\ 0_{n_\phi \times n_G} & I_{n_\phi} \end{bmatrix}.$$

If there exists a matrix $P \in \mathbb{S}_{++}^{n_G}$, and vector $\lambda \in \mathbb{R}^{n_\phi}$ with $\lambda \geq 0$ such that

$$R_V^\top \begin{bmatrix} A_G^\top P A_G - P & A_G^\top P B_G \\ B_G^\top P A_G & B_G^\top P B_G \end{bmatrix} R_V$$

$$+ R_\phi^\top \begin{bmatrix} -2A_\phi B_\phi \Lambda & (A_\phi + B_\phi)\Lambda \\ (A_\phi + B_\phi)\Lambda & -2\Lambda \end{bmatrix} R_\phi < 0, \tag{7.4a}$$

$$\begin{bmatrix} h_i^2 & H_i^\top \\ H_i & P \end{bmatrix} \geq 0, \;\; i = 1, \cdots, n_X, \tag{7.4b}$$

then: (i) the feedback system consisting of G and π is locally asymptotically stable around x_*, and (ii) the set $\mathcal{E}(P)$, defined by (2.4), is an inner-approximation to the ROA.

Proof. By Schur complements, (7.4b) is equivalent to:

$$H_i^\top P^{-1} H_i \leq h_i^2, \;\; i = 1, \cdots, n_X. \tag{7.5}$$

It follows from Lemma 1 in [71] that:

$$\mathcal{E}(P) \subseteq \{x \in \mathbb{R}^{n_G} : -h_i \leq H_i^\top x \leq h_i, \;\; i = 1, ..., n_X\} = X.$$

To summarize, feasibility of (7.4b) verifies that if $x(k) \in \mathcal{E}(P) \subseteq X$ then $v_\phi(k) \in [\underline{v}, \bar{v}]$ and hence the local sector conditions are valid.

Next, define $V(x) := x^\top P x$. Since the LMI in (7.4a) is strict, there exist $\epsilon > 0$ such that left / right multiplication of the LMI $\begin{bmatrix} x(k)^\top, \; w_\phi(k)^\top \end{bmatrix}$ and its transpose yields:

$$\begin{aligned} V(x(k+1)) - V(x(k)) + \begin{bmatrix} v_\phi(k) \\ w_\phi(k) \end{bmatrix}^\top \begin{bmatrix} -2A_\phi B_\phi \Lambda & (A_\phi + B_\phi)\Lambda \\ (A_\phi + B_\phi)\Lambda & -2 \end{bmatrix} \begin{bmatrix} v_\phi(k) \\ w_\phi(k) \end{bmatrix} \\ \leq -\epsilon |x(k) - x_*|^2, \end{aligned} \tag{7.6}$$

Using the proof by induction argument from [84, Theorem 1], we can show that $\mathcal{E}(P)$ is an invariant set: if $x(0) \in \mathcal{E}(P)$, then $x(k) \in \mathcal{E}(P) \; \forall k \geq 0$. As noted above, $x(k) \in \mathcal{E}(P)$ implies the local sector $[\alpha_\phi, \beta_\phi]$. Then, by Lemma 3, if $x(0) \in \mathcal{E}(P)$, the final term on the left side of (7.6) is ≥ 0 for all $k \geq 0$, and thus $V(x(k+1)) - V(x(k)) \leq -\epsilon |x(k) - x_*|^2$ $\forall k \geq 0$. Therefore, x_* is an asymptotically stable equilibrium point, and $\mathcal{E}(P)$ is a ROA inner approximation. □

The Lyapunov condition (7.4a) is convex in P and λ if the weight matrix N is given, and thus we can efficiently compute the ROA inner-estimates. However, this condition is computationally intractable for NN controller synthesis, as it is nonconvex if we search over N, P, and λ simultaneously.

7.3 Convex Stability and Safety Conditions

In [64, 87], α_ϕ is set to zero to formulate convex constraints. However, this restriction is too coarse for stability analysis of NN controlled systems. Instead, we perform a loop transformation [1] as shown in Fig. 7.2 to convexify the stability condition without having restrictions on α_ϕ and β_ϕ.

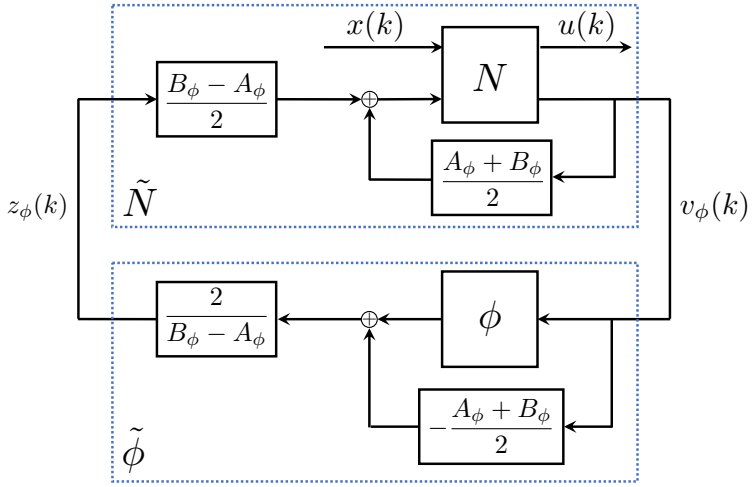

Figure 7.2: Loop transformation. If ϕ is in the sector $[\alpha_\phi, \beta_\phi]$, then $\tilde{\phi}$ is in the sector $[-1_{n_\phi \times 1}, 1_{n_\phi \times 1}]$.

Loop transformation

Through loop transformation, we obtain a new representation of the NN controller, which is equivalent to (7.2),

$$\begin{bmatrix} u(k) \\ v_\phi(k) \end{bmatrix} = \tilde{N} \begin{bmatrix} x(k) \\ z_\phi(k) \end{bmatrix}, \tag{7.7a}$$

$$z_\phi(k) = \tilde{\phi}(v_\phi(k)), \tag{7.7b}$$

where $\tilde{N}$ and $\tilde{\phi}$ are defined in Fig. 7.2. Here, we also partition $\tilde{N}$ compatibly with the inputs (x, z_ϕ) and outputs (u, v_ϕ)

$$\tilde{N} = \begin{bmatrix} \tilde{N}_{ux} & \tilde{N}_{uz} \\ \tilde{N}_{vx} & \tilde{N}_{vz} \end{bmatrix}. \tag{7.8}$$

The loop transformation normalizes the nonlinearity $\tilde{\phi}$ to lie in the sector $[-1_{n_\phi \times 1}, 1_{n_\phi \times 1}]$. As a result, $\tilde{\phi}$ satisfies the sector QC for any $\Lambda = diag(\lambda)$ with $\lambda \geq 0$:

$$\begin{bmatrix} v_\phi(k) \\ z_\phi(k) \end{bmatrix}^\top \begin{bmatrix} \Lambda & 0 \\ 0 & -\Lambda \end{bmatrix} \begin{bmatrix} v_\phi(k) \\ z_\phi(k) \end{bmatrix} \geq 0, \ \forall v_\phi \in [\underline{v}, \overline{v}]. \tag{7.9}$$

The input to N is transformed by the following equation:

$$w_\phi(k) = \frac{B_\phi - A_\phi}{2} z_\phi(k) + \frac{A_\phi + B_\phi}{2} v_\phi(k). \tag{7.10}$$

The transformed matrix $\tilde{N}$ can be computed by combining this relation with (7.2a). Substituting (7.10) into (7.2a) we obtain

$$u(k) = N_{ux}x(k) + C_1 z_\phi(k) + C_2 v_\phi(k), \tag{7.11}$$
$$v_\phi(k) = N_{vx}x(k) + C_3 z_\phi(k) + C_4 v_\phi(k), \tag{7.12}$$

where

$$C_1 = N_{uw}\frac{B_\phi - A_\phi}{2},\ C_2 = N_{uw}\frac{A_\phi + B_\phi}{2},$$
$$C_3 = N_{vw}\frac{B_\phi - A_\phi}{2},\ C_4 = N_{vw}\frac{A_\phi + B_\phi}{2}.$$

The expression for $v_\phi(k)$ can be solved from (7.12):

$$v_\phi(k) = (I - C_4)^{-1}N_{vx}x(k) + (I - C_4)^{-1}C_3 z_\phi(k). \tag{7.13}$$

Substituting (7.13) into (7.11) yields

$$u(k) = (N_{ux} + C_2(I - C_4)^{-1}N_{vx})x(k) + (C_1 + C_2(I - C_4)^{-1}C_3)z_\phi(k). \tag{7.14}$$

Matching (7.13) and (7.14) with (7.7a) we get

$$\tilde{N} = \begin{bmatrix} N_{ux} + C_2(I - C_4)^{-1}N_{vx} & C_1 + C_2(I - C_4)^{-1}C_3 \\ (I - C_4)^{-1}N_{vx} & (I - C_4)^{-1}C_3 \end{bmatrix}.$$

Thus, $\tilde{N}$ is a function of N denoted concisely as $\tilde{N} = f(N)$. It is important to note that $\tilde{N}$ depends on N both directly, and also indirectly through its dependence on (A_ϕ, B_ϕ). Specifically, suppose both N and a hypercube state bound $[\underline{x}, \bar{x}]$ are given. Then $\tilde{N}$ is constructed by: (i) propagating $[\underline{x}, \bar{x}]$ through the NN to compute bounds $[\overline{v}, \underline{v}]$ on the activation inputs, (ii) computing local sector bounds (A_ϕ, B_ϕ) consistent with $[\overline{v}, \underline{v}]$, and (iii) performing the steps in this section to compute $\tilde{N}$ from (N, A_ϕ, B_ϕ).

Stability condition after loop transformation

Similar to the original Lyapunov condition (7.4a), the new Lyapunov condition for the feedback system of G in (6.1) and NN in (7.7) can be written as

$$\tilde{R}_V^\top \begin{bmatrix} A_G^\top P A_G - P & A_G^\top P B_G \\ B_G^\top P A_G & B_G^\top P B_G \end{bmatrix} \tilde{R}_V + \tilde{R}_\phi^\top \begin{bmatrix} \Lambda & 0 \\ 0 & -\Lambda \end{bmatrix} \tilde{R}_\phi < 0, \tag{7.15}$$

where

$$\tilde{R}_V = \begin{bmatrix} I_{n_G} & 0 \\ \tilde{N}_{ux} & \tilde{N}_{uz} \end{bmatrix},\ \text{and}\ \tilde{R}_\phi = \begin{bmatrix} \tilde{N}_{vx} & \tilde{N}_{vz} \\ 0 & I_{n_\phi} \end{bmatrix}. \tag{7.16}$$

Lemma 4. *Consider the feedback system of G in (6.1) and NN in (7.2) with the state constraint set X. If there exist a matrix $P \in \mathbb{S}^{n_G}_{++}$, and vector $\lambda \in \mathbb{R}^{n_\phi}$ such that (7.15) (where $\tilde{N} = f(N)$) and (7.4b) hold, then: (i) the feedback system consisting of G in (6.1) and NN in (7.2) is locally asymptotically stable around x_*, and (ii) the set $\mathcal{E}(P)$ is a ROA inner-approximation for it.*

Proof. It follows from the assumption that (7.15) and (7.4b) hold that the feedback system of G in (6.1) and NN in (7.7) is locally asymptotically stable around x_*, and $\mathcal{E}(P)$ is its ROA inner-approximation. Since the representations (7.2) and (7.7) of NN are equivalent, the feedback system of G in (6.1) and NN in (7.2) is identical to the feedback system of G in (6.1) and NN in (7.7). As a result, the feedback system consisting of G in (6.1) and NN in (7.2) is locally asymptotically stable around x_*, and the set $\mathcal{E}(P)$ is a ROA inner-approximation for it. □

The new Lyapunov condition (7.15) is convex in P and Λ using $\tilde{N} = f(N)$, where N is given. To incorporate the stability condition in the IL process, we will proceed by treating $\tilde{N} \in \mathbb{R}^{(n_u+n_\phi)\times(n_G+n_\phi)}$ as a decision variable along with P and Λ, and try to derive a stability condition that is convex in $(P, \Lambda, \tilde{N})$. Substitute (7.16) into (7.15) to obtain

$$\begin{bmatrix} A_G + B_G\tilde{N}_{ux} & B_G\tilde{N}_{uz} \\ \tilde{N}_{vx} & \tilde{N}_{vz} \end{bmatrix}^\top \begin{bmatrix} P & 0 \\ 0 & \Lambda \end{bmatrix} \begin{bmatrix} A_G + B_G\tilde{N}_{ux} & B_G\tilde{N}_{uz} \\ \tilde{N}_{vx} & \tilde{N}_{vz} \end{bmatrix} - \begin{bmatrix} P & 0 \\ 0 & \Lambda \end{bmatrix} < 0.$$

Applying Schur complements yields the equivalent condition

$$\begin{bmatrix} P & 0 & A_G^\top + \tilde{N}_{ux}^\top B_G^\top & \tilde{N}_{vx}^\top \\ 0 & \Lambda & \tilde{N}_{uz}^\top B_G^\top & \tilde{N}_{vz}^\top \\ A_G + B_G\tilde{N}_{ux} & B_G\tilde{N}_{uz} & P^{-1} & 0 \\ \tilde{N}_{vx} & \tilde{N}_{vz} & 0 & \Lambda^{-1} \end{bmatrix} > 0, \tag{7.17}$$

and $P > 0$, $\Lambda > 0$. Now (7.17) is linear in $\tilde{N}$, but still nonconvex in P and Λ. Multiplying (7.17) on the left and right by $diag(P^{-1}, \Lambda^{-1}, I_{n_G}, I_{n_\phi})$ we obtain

$$\begin{bmatrix} Q_1 & 0 & Q_1A_G^\top + L_1^\top B_G^\top & L_3^\top \\ 0 & Q_2 & L_2^\top B_G^\top & L_4^\top \\ A_GQ_1 + B_GL_1 & B_GL_2 & Q_1 & 0 \\ L_3 & L_4 & 0 & Q_2 \end{bmatrix} > 0, \tag{7.18}$$

where $Q_1 = P^{-1} > 0$, $Q_2 = \Lambda^{-1} > 0$, $L_1 = \tilde{N}_{ux}Q_1$, $L_2 = \tilde{N}_{uz}Q_2$, $L_3 = \tilde{N}_{vx}Q_1$, and $L_4 = \tilde{N}_{vz}Q_2$.

The stability condition (7.18) is convex in the decision variables $(Q_1, Q_2, L_1, \ldots, L_4)$, where $Q_1 \in \mathbb{S}^{n_G}_{++}$, $Q_2 \in \mathbb{S}^{n_\phi}_{++}$ and Q_2 is a diagonal matrix, $L_1 \in \mathbb{R}^{n_u\times n_G}$, $L_2 \in \mathbb{R}^{n_u\times n_\phi}$, $L_3 \in \mathbb{R}^{n_\phi\times n_G}$, and $L_4 \in \mathbb{R}^{n_\phi\times n_\phi}$. Variables $(P, \Lambda, \tilde{N})$ that satisfy the Lyapunov condition (7.15)

can be recovered using the computed $(Q_1, Q_2, L_1, \ldots, L_4)$ through the following equations: $P = Q_1^{-1}, \Lambda = Q_2^{-1}$, and

$$\tilde{N} = LQ^{-1}, \tag{7.19}$$

where

$$Q := \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2 \end{bmatrix}, \text{ and } L := \begin{bmatrix} L_1 & L_2 \\ L_3 & L_4 \end{bmatrix}. \tag{7.20}$$

Thus, the convex stability condition (7.18) allows us to search over P, Λ, and $\tilde{N}$ simultaneously.

Moreover, to enforce the safety condition ($x(k) \in X \ \forall k \geq 0$) of the system, convex constraints on Q_1 are imposed:

$$H_i^\top Q_1 H_i \leq h_i^2, \ i = 1, \cdots, n_X, \tag{7.21}$$

which is derived directly from (7.4b) by Schur complements, and using $Q_1 = P^{-1}$. Again, this ensures $\mathcal{E}(Q_1^{-1}) \subseteq X$.

Denote the LMIs (7.18), (7.21) with $Q_1 > 0$ and $Q_2 > 0$ altogether as $\text{LMI}(Q, L) > 0$, which will later be incorporated in the IL process to learn robust NN controllers.

Remark 7. *Note that model uncertainties are not considered in the dissertation. They can be incorporated in the Lyapunov condition* (7.4a) *or* (7.15) *using integral quadratic constraints as in [84]. However, to derive convex stability conditions, only limited types of uncertainties may be incorporated.*

7.4 Safe Imitation Learning Algorithm

Given state and control data pairs from the expert demonstrations, we use a loss function $\mathcal{L}(N)$ to train NN controllers with weights N to match the data. Common choices of the loss function includes mean squared error, absolute error, and cross-entropy. In general, $\mathcal{L}(N)$ is non-convex in N. We propose the next optimization to solve the safe IL problem,

$$\min_{N,Q,L} \ \eta_1 \mathcal{L}(N) - \eta_2 \log\det(Q_1) \tag{7.22a}$$

$$\text{s.t. } \text{LMI}(Q, L) > 0 \tag{7.22b}$$

$$f(N)Q = L \tag{7.22c}$$

where Q and L are defined in (7.20). The optimization has separate objectives. The cost function (7.22a) combines the IL loss function with a term that attempts to increase the volume of $\mathcal{E}(Q_1^{-1})$ (which is proportional to $\det(Q_1)$). The parameters $\eta_1, \eta_2 > 0$ reflect the relative importance between imitation learning accuracy and size of the robustness margin. The optimization has two sets of decision variables: N and (Q, L). The former is involved

in mimicking the expert behaviour, and the latter are involved in the stability and safety constraints (7.22b). The two sets of variables are connected through the equality constraint (7.22c). Note that (7.22c) is equivalent to $f(N) = LQ^{-1}$, and the term on the right-hand side equals to $\tilde{N}$ from (7.19). Therefore, (7.22c) essentially means that the first set of decision variable N, after being transformed by the nonlinear function f, satisfies the stability and safety constraints.

Similar to [64], we use the alternating direction method of multipliers (ADMM) algorithm to solve this constrained learning problem. We first define an augmented loss function

$$\begin{aligned}\mathcal{L}_a(N, Q, L, Y) = \ & \eta_1 \mathcal{L}(N) - \eta_2 \log\det(Q_1) \\ & + \operatorname{tr}\left(Y^\top \cdot (f(N)Q - L)\right) + \frac{\rho}{2} \left\| f(N)Q - L \right\|_F^2, \end{aligned} \tag{7.23}$$

where $\|\cdot\|_F$ is the Frobenius norm, $Y \in \mathbb{R}^{(n_u+n_\phi)\times(n_G+n_\phi)}$ is the Lagrange multiplier, and $\rho > 0$ is the regularization parameter typically affecting the convergence rate of ADMM. The ADMM algorithm takes the following form:
1. N-update: $N^{k+1} = \arg\min_N \mathcal{L}_a(N, Q^k, L^k, Y^k)$.
2. (Q, L)-update:

$$\begin{aligned}(Q^{k+1}, L^{k+1}) = \arg\min_{Q,L} \ & \mathcal{L}_a(N^{k+1}, Q, L, Y^k) \\ \text{s.t. } & \mathrm{LMI}(Q, L) > 0\end{aligned}$$

3. Y-update: If $\left\| f(N^{k+1})Q^{k+1} - L^{k+1} \right\|_F \leq \sigma$, where $\sigma > 0$ is the stopping tolerance, then the algorithm has converged, and we have found a solution to (7.22), so terminate the algorithm. Otherwise, update Y and return to step 1.

$$Y^{k+1} = Y^k + \rho\left(f(N^{k+1})Q^{k+1} - L^{k+1}\right)$$

Step 1 can be solved using gradient based algorithm, e.g., stochastic gradient descent. The optimization in Step 2 is convex, and can be solved effectively using SDP solvers. The variable Y in Step 3 accumulates the deviation from the constraint (7.22c) as in integral control. Since the loss function $\mathcal{L}(N)$ is generally nonconvex, and the constraint (7.22c) is also nonconvex, the proposed ADMM does not have the guarantee to converge to the global optima. However, any converged solution (even local optima) provides a safe NN controller with stability and safety guarantees.

7.5 Numerical Examples

In the following example, Step 1 of ADMM algorithm is implemented on Tensorflow, and solved by ADAM [88]. Step 2 is formulated using CVX, and is solved by MOSEK. The mean squared error is chosen as the loss function $\mathcal{L}(N)$. The code is available at `https://github.com/heyinUCB/IQCbased_ImitationLearning`.

Inverted pendulum

Consider an inverted pendulum with mass $m = 0.15$ kg, length $l = 0.5$ m, and friction coefficient $\mu = 0.5$ Nms/rad. The discretized and linearized dynamics are:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ \frac{g\delta}{l} & 1 - \frac{\mu\delta}{ml^2} \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{\delta}{ml^2} \end{bmatrix} u(k),$$

where the states x_1, x_2 represent the angular position (rad) and velocity (rad/s), u is the control input (Nm), and $\delta = 0.02$ s is the sampling time. The state constraint set is $X = [-2.5, 2.5] \times [-6, 6]$. To generate state and control data pairs for IL, we design an explicit model predictive controller (MPC) to serve as the expert. By fitting a NN controller to the explicit MPC controller, we can expedite the evaluation of controllers during run-time [89–91]. In this example, besides a NN controller, we will also provide its associated ROA inner-approximation that guarantees stability and safety. The NN controller is parameterized by a 2-layer, feedforward NN with $n_1 = n_2 = 10$ and tanh as the activation function for both layers. Take $\rho = 1$, $\eta_1 = 100$, and $\eta_2 = 5$. The ADMM algorithm is terminated after 16 iterations, and $\|f(N) - LQ^{-1}\|_F = 0.17$.

In Fig. 7.3, the plot on the left shows the learned NN controller with a blue surface, and state and control data pairs from expert demonstrations with orange dots; the plot on the right shows the ROAs of the MPC controller and the NN controller with oranges dots and a blue ellipsoid, respectively. We can notice that the ROA of the NN controller is tightly contained by the state constraint set X (shown with a gray rectangle), which guarantees the safety of the system.

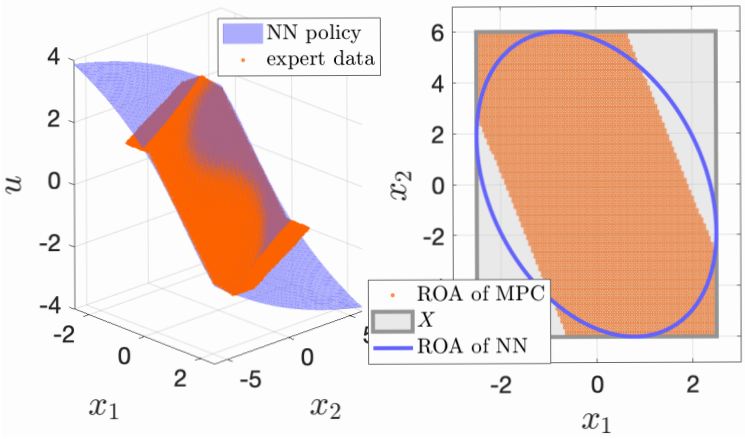

Figure 7.3: Left: NN controller vs. expert data from demonstrations; Right: ROAs of MPC controller and NN controller, and state constraint set X of the inverted pendulum

Generic Transport Model

The Generic Transport Model (GTM) is a 5.5% scale commercial aircraft. Linearizing and discretizing the longitudinal dynamics given in [37] with sampling time $\delta = 0.02$ s yields:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} 0.935 & 0.019 \\ -0.907 & 0.913 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \end{bmatrix} + \begin{bmatrix} -0.006 \\ -1.120 \end{bmatrix} u(k),$$

where the states x_1, x_2 represent angle of attack (rad), and pitch rate (rad/s), and the control u represents the elevator deflection (rad). Take the state constraint set as $X = [-2, 2] \times [-3, 3]$. In this example, we design an LQR controller to produce expert data. The NN controller is again parameterized by a 2-layer, feedforward NN with $n_1 = n_2 = 16$ and tanh as the activation function for both layers. In this example, we will show how the parameter η_2 affects the result. To do so, two experiments are carried out using two sets of parameters $(\rho = 1, \eta_1 = 100, \eta_2 = 5)$ and $(\rho = 1, \eta_1 = 100, \eta_2 = 20)$, meaning that we care more about the size of the ROA inner-approximation, and less about the IL accuracy in the second experiment than we do in the first experiment. In both experiments, the ADMM algorithm is terminated in 20 iterations.

The ROA inner-approximations of the NN controllers from the two experiments are shown in Figure 7.4. The one computed with $\eta_2 = 5$ is shown with a magenta ellipsoid, and the one computed with $\eta_2 = 20$ is shown with a blue ellipsoid. First, it is important to notice that both NN controllers' ROA inner-approximations are larger than that of the expert's LQR controller (shown with a dashed gray ellipsoid), thanks to the second term in the cost function (7.22a), which enhances the robustness of IL. Also, as expected, the ROA inner-approximation of the NN controller with $\eta_2 = 20$ is larger than that with $\eta_2 = 5$, since a larger η_2 leads to a larger ROA inner-approximation. However, the larger ROA inner-approximation comes at the cost of less accurate regression to the expert data. As shown in Figure 7.5, the mesh plot of the NN controller with $\eta_2 = 20$ (shown with a blue surface) is more off from the expert data (shown with orange stars) than that with $\eta_2 = 5$ (shown with a magenta surface).

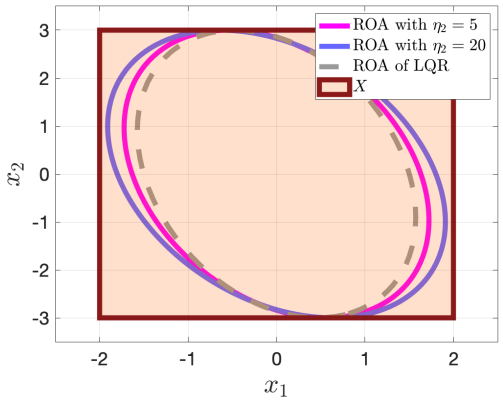

Figure 7.4: ROA inner-approximations and state constraint set X of GTM

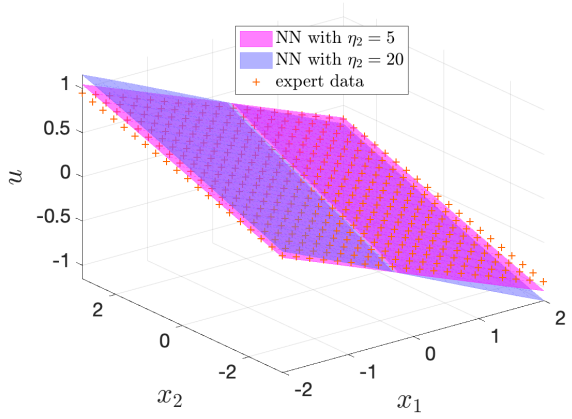

Figure 7.5: NN controllers vs. expert data of GTM

7.6 Chapter Summary

In this chapter, we present an IL algorithm with stability and safety guarantees for LTI systems. First, we convexify the stability and safety conditions for NN controlled systems from Chapter 6 using loop transformation. Then, these conditions are incorporated in the IL process, which trades off between the IL accuracy, and size of the stability margin. We propose an ADMM based algorithm to solve the IL problem. Finally, we illustrate the method on an inverted pendulum example, and an aircraft example.

Chapter 8

Conclusions and Future Work

In this dissertation we present an approach to control synthesis and analysis of nonlinear and NN controlled systems with robustness guarantees. The approach merges dissipation inequalities and IQCs that describe perturbations Δ and nonlinear activation functions in the NNs. In Chapter 3, we present a method for outer-approximating the FRS of uncertain nonlinear system on a finite time horizon. Both soft and hard factorizations of IQCs are considered to characterize perturbations. In Chapter 4, we switch from analysis problem to the control synthesis problem: given a target set, we present a method for computing a control law that steers the system to the target set, and its associated BRS inner-approximations. In Chapter 5, we propose a hierarchical planning and control framework, where a tracking controller is designed based on the missmatch between the planning and tracking models. The tracking controller guarantees that the error between the planning and tracking trajectories is bounded. In Chapter 6, stability conditions are derived to assess local stability and compute ROA inner-approximations of NN controlled LTI systems. Local IQCs are used to capture the sector and slope information of nonlinear activation functions. In Chapter 7, we convexify the stability conditions from Chapter 6 using loop transformation. These convex conditions are then incorporated in the IL process to robustify the learned NN controllers.

The key idea of this dissertation is to merge dissipation inequalities with IQCs to provide a unified framework for analysis and control synthesis of uncertain nonlinear and NN controlled systems. We briefly discuss several future directions along the path of this dissertation.

Disturbance accommodation and reachability analysis for NN controlled systems: In Chapter 6, we present a method for stability analysis of NN controlled LTI systems. This method can be extended to the infinite time horizon reachability analysis for NN controlled systems with ℓ_2 or polytopic bounded disturbances.

Compositional analysis of for interconnected NN controlled systems: In Chapter 6, the proposed algorithm might not be able to scale to large-scale systems like power systems, UAV swarms, or vehicle platoons that consist of many small interconnected subsystems with NN controllers. A compositional approach (e.g. [92]) can be applied here to certify the properties of the interconnected NN controlled systems, which decomposes the certification problem into parallelizable, local problems for individual subsystems and a global

problem that can be solved efficiently.

Stability analysis for NN controlled nonlinear systems: Although in Chapter 6 by including IQCs we consider a limited types of nonlinearities (e.g., saturation, trigonometric functions), we are not able to assess the stability of a general class of nonlinear system. It is possible to apply SOS techniques to the framework in order to accommodate nonlinear polynomial systems.

Reinforcement learning with stability and safety guarantees: In Chapter 7, we derive and incorporate convex stability and safety conditions for LTI systems with NN controllers in the imitation learning process, and these conditions are also possible to be included in the reinforcement learning process, including policy gradient and actor-critic algorithms.

Robust control of systems identified using neural state space models: Consider a neural state space model $\dot{x} = f_{\text{NN}}(x, u)$, where f_{NN} is a NN function obtained through system identification. We can design robust controllers for it by isolating the nonlinear activation functions from the linear operations of the NN, and incorporating those activation functions in the robust control framework using sector QCs. Other types of IQCs can be used to cover the identification error between the neural state space model and the real dynamics.

Robust tracking of planning trajectories generated from Neural Network policies: In the literature, many planning algorithms are based on reinforcement learning and imitation learning, and planning trajectories are generated using NNs. The tracking control design method proposed in Chapter 5 can be combined with the IQC framework to account for NN planning policies.

Improve the performance of robust controllers using reinforcement learning: For a large batch of products with the same specification, like autonomous vehicles, each individual product's system dynamics are different from others' due to the production error. A robust controller can be designed to account for the dynamics difference between products. Although this single robust controller guarantees to work on all the products, its performance might be overly conservative for running each individual product, as it fails to consider the specific dynamics of each product. Reinforcement learning can take the robust controller as a good starting point for control design, and running reinforcement learning algorithm on each product can further improve the control performance.

Appendix A

Iterative Algorithm

The algorithm for solving optimization (5.11) is summarized below, the (κ, γ)-step of which treats γ as a decision variable. By minimizing γ, the volume of $\Omega_\gamma^{V^{j-1}}$ can be shrunk. In the V-step, (A.1) enforces $\Omega_{\gamma^j}^{V^j} \subseteq \Omega_{\gamma^j}^{V^{j-1}}$.

Algorithm 4 Alternating direction method

Input: function V^0 such that constraints (5.11) are feasible by proper choice of s, l, κ, γ.
Output: κ, γ, V.

1: **for** $j = 1 : N_{\text{iter}}$ **do**
2: **$(\boldsymbol{\kappa}, \boldsymbol{\gamma})$-step**: decision variables (s, l, κ, γ). Minimize γ subject to (5.11) using $V = V^{j-1}$. This yields $(l^j, s_{4,i}^j, s_{7,i}^j, \kappa^j)$ and the cost γ^j.
3: ***V*-step**: decision variables $(s_{1\to 3}, s_{5\to 6,i}, s_{8\to 9,i}, V)$. Maximize the feasibility subject to (5.11) as well as $s_{10} - \epsilon \in \Sigma[e]$, and

$$-s_{10} \cdot (V^{j-1} - \gamma^j) + (V - \gamma^j) \in \Sigma[e], \tag{A.1}$$

using $(\gamma = \gamma^j, s_{4,i} = s_{4,i}^j, s_{7,i} = s_{7,i}^j, \kappa = \kappa^j, l = l^j)$. This yields V^j.
4: **end for**

The input to Algorithm 4 is a feasible initial guess V^0. One candidate might be a quadratic Lyapunov function $\bar{V}$ obtained by solving Lyapunov equations using the linearized error dynamics with LQR controllers. However, $\bar{V}$ might be too coarse to be feasible for the constraints (5.11). Here, we introduced a slack variable $\lambda > 0$ to the constraint (5.11c) to relax the constraint, and quantify how far $\bar{V}$ is away from a feasible candidate:

$$-\frac{\partial V}{\partial e} \cdot (f_e + g_e \cdot \kappa) + \lambda - \epsilon e^\top e + l \cdot (V - \gamma) + s_1 \cdot p_{\hat{x}} + s_2 \cdot p_{\hat{u}} + s_3 \cdot p_w \in \Sigma[(e, \hat{x}, \hat{u}, w)]. \tag{A.2}$$

By iteratively search over two bilinear groups of decision variables, we minimize λ until

$\lambda \leq 0$. Based on this idea, an algorithm to compute V^0 from $\bar{V}$ is proposed as Algorithm 5.

Algorithm 5 Computation of V^0

Input: function $\bar{V}$, and $\bar{\gamma} > 0$.
Output: V^0.

1: $V^{\text{pre}} \leftarrow \bar{V}$
2: **while** $\lambda > 0$ **do**
3: **$\boldsymbol{\kappa}$-step**: decision variables (s, l, κ). Minimize λ subject to (5.11a–5.11b, A.2, 5.11d–5.11e), using $V = V^{\text{pre}}$, $\gamma = \bar{\gamma}$.
$(l^{\text{pre}}, s_{4,i}^{\text{pre}}, s_{7,i}^{\text{pre}}, \kappa^{\text{pre}}) \leftarrow (l, s_{4,i}, s_{7,i}, \kappa)$
4: **V-step**: decision variables $(s_{1\to 3}, s_{5\to 6,i}, s_{8\to 9,i}, V)$. Minimize λ subject to (5.11a–5.11b, A.2, 5.11d–5.11e) using $(\gamma = \bar{\gamma}, s_{4,i} = s_{4,i}^{\text{pre}}, s_{7,i} = s_{7,i}^{\text{pre}}, \kappa = \kappa^{\text{pre}}, l = l^{\text{pre}})$.
$V^{\text{pre}} \leftarrow V$
5: **end while**
6: $V^0 \leftarrow V^{\text{pre}}$

Bibliography

[1] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. USA: Prentice-Hall, Inc., 1996.

[2] J. Buch and P. Seiler, "Finite horizon robust synthesis," in *2020 American Control Conference (ACC)*, pp. 1551–1556, IEEE, 2020.

[3] P. Seiler, R. Moore, C. Meissen, M. Arcak, and A. Packard, "Finite horizon robustness analysis of ltv systems using integral quadratic constraints," *Automatica*, vol. 100, pp. 135–143, 2019.

[4] S. Wang, H. Pfifer, and P. Seiler, "Robust synthesis for linear parameter varying systems using integral quadratic constraints," *Automatica*, vol. 68, pp. 111–118, 2016.

[5] A. Majumdar, A. A. Ahmadi, and R. Tedrake, "Control design along trajectories with sums of squares programming," in *Proceedings of International Conference on Robotics and Automation*, pp. 4054–4061, Karlsruhe, Germany: IEEE, 2013.

[6] M. Jones and M. M. Peet, "Relaxing the Hamilton Jacobi Bellman equation to construct inner and outer bounds on reachable sets," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 2397–2404, 2019.

[7] A. Majumdar and R. Tedrake, "Funnel libraries for real-time robust feedback motion planning," *The international Journal of Robotics Research*, vol. 36, pp. 947–982, 2017.

[8] I. Mitchell and C. Tomlin, "Level set methods for computation in hybrid systems," in *In Hybrid Systems: Computation and Control*, pp. 310–323, 2000.

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

[10] P. Holmes, S. Kousik, S. Mohan, and R. Vasudevan, "Convex estimation of the α–confidence reachable set for systems with parametric uncertainty," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, pp. 4097–4103, Dec 2016.

[11] H. Hu, M. Fazlyab, M. Morari, and G. J. Pappas, "Reach-SDP: Reachability Analysis of Closed-Loop Systems with Neural Network Controllers via Semidefinite Programming," p. arXiv:2004.07876, Apr. 2020.

[12] M. Jin and J. Lavaei, "Stability-certified reinforcement learning: A control-theoretic perspective," p. arXiv:1810.11505, Oct. 2018.

[13] A. Megretski and A. Rantzer, "System analysis via integral quadratic constraints," *IEEE Transactions on Automatic Control*, vol. 42, pp. 819–830, June 1997.

[14] M. Fetzer, C. W. Scherer, and J. Veenman, "Invariance with dynamic multipliers," *IEEE Transactions on Automatic Control*, vol. 63, no. 7, pp. 1929–1942, 2018.

[15] J. C. Willems, "Dissipative dynamical systems part i: General theory," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 321–351, 1972.

[16] J. C. Willems, "Dissipative dynamical systems part ii: Linear systems with quadratic supply rates," *Archive for Rational Mechanics and Analysis*, vol. 45, no. 5, pp. 352–393, 1972.

[17] G. Zames and P. L. Falb, "Stability conditions for systems with monotone and slope-restricted nonlinearities," *SIAM Journal on Control*, vol. 6, no. 1, pp. 89–108, 1968.

[18] J. Veenman, C. W. Scherer, and H. Köroğlu, "Robust stability and performance analysis based on integral quadratic constraints," *European Journal of Control*, vol. 31, pp. 1 – 32, 2016.

[19] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical Programming*, vol. 96, no. 2, pp. 293–320, 2003.

[20] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *Proc. of Hybrid Systems: Computation and Control*, pp. 477–492, 2004.

[21] A. Papachristodoulou and S. Prajna, "On the construction of lyapunov functions using the sum of squares decomposition," in *IEEE Conference on Decision and Control*, pp. 3482–3487, 2002.

[22] P. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," 2000.

[23] E. Summers, A. Chakraborty, W. Tan, U. Topcu, P. Seiler, G. Balas, and A. Packard, "Quantitative local L2-gain and reachability analysis for nonlinear systems," *International Journal of Robust and Nonlinear Control*, vol. 23, pp. 1115–1135, 2013.

[24] A. Chakraborty, P. Seiler, and G. Balas, "Local performance analysis of uncertain polynomial systems with applications to actuator saturation," in *IEEE Conference on Decision and Control*, pp. 8176–8181, 2011.

[25] M. Jones and M. M. Peet, "Using sos for optimal semialgebraic representation of sets: Finding minimal representations of limit cycles, chaotic attractors and unions," in *2019 American Control Conference (ACC)*, pp. 2084–2091, July 2019.

[26] Z. Jarvis-Wloszek, R. Feeley, W. Tan, K. Sun, and A. Packard, "Control applications of sum of squares programming," in *Positive Polynomials in Control*, vol. 312, pp. 3–22, Springer, Berlin, Heidelberg, 2005.

[27] A. Magnani, S. Lall, and S. Boyd, "Tractable fitting with convex polynomials via sum-of-squares," in *44th IEEE Conference on Decision and Control*, pp. 1672–1677, 2005.

[28] K. Zhou, J. C. Doyle, and K. Glover, *Robust and Optimal Control*. USA: Prentice-Hall, Inc., 1996.

[29] V. Balakrishnan, "Lyapunov functionals in complex μ analysis," *IEEE Transactions on Automatic Control*, vol. 47, pp. 1466–1479, Sep. 2002.

[30] A. Rantzer, "On the Kalman-Yakubovich-Popov lemma," *Systems & Control Letters*, vol. 28, no. 1, pp. 7 – 10, 1996.

[31] P. Seiler, "An iterative algorithm to estimate invariant sets for uncertain systems," in *2018 Annual American Control Conference (ACC)*, pp. 4249–4254, 2018.

[32] P. Seiler, "Stability analysis with dissipation inequalities and integral quadratic constraints," *IEEE Transactions on Automatic Control*, vol. 60, no. 6, pp. 1704–1709, 2015.

[33] J. Löfberg, "YALMIP : a toolbox for modeling and optimization in matlab," in *IEEE International Conference on Robotics and Automation*, pp. 284–289, 2004.

[34] MOSEK ApS, "The MOSEK optimization toolbox for MATLAB manual. version 8.1.," 2017.

[35] A. Murch and J. Foster, "Recent NASA research on aerodynamic modeling of post-stall and spin dynamics of large transport airplanes," in *45th AIAA Aerospace Sciences Meeting and Exhibit*, 2007.

[36] B. Stevens and F. Lewis, *Aircraft Control and Simulation*. Hoboken, NJ: John Wiley & Sons, 1992.

[37] A. Chakraborty, P. Seiler, and G. Balas, "Nonlinear region of attraction analysis for flight control verification and validation," *Control Engineering Practice*, vol. 19, pp. 335–345, 04 2011.

[38] A. Chakraborty, P. Seiler, and G. Balas, "Susceptibility of F/A-18 flight controllers to the falling-leaf mode: Nonlinear analysis," *Journal of Guidance, Control, and Dynamics*, vol. 34, pp. 73–85, 2011.

[39] J. Löfberg, "Pre-and post-processing sum-of-squares programs in practice," *IEEE Transactions on Automatic Control*, vol. 54, pp. 1007–1011, 2009.

[40] D. Henrion and M. Korda, "Convex computation of the region of attraction of polynomial control systems," *IEEE Transactions on Automatic Control*, vol. 59, pp. 297–312, 2014.

[41] A. Majumdar, R. Vasudevan, M. M. Tobenkin, and R. Tedrake, "Convex optimization of nonlinear feedback controllers via occupation measures," *The International Journal of Robotics Research*, vol. 33, pp. 1209–1230, 2014.

[42] I. M. Mitchell, A. M. Bayen, and C. J. Tomlin, "A time-dependent Hamilton-Jacobi formulation of reachable sets for continuous dynamic games," *IEEE Transactions on Automatic Control*, vol. 50, pp. 947–957, July 2005.

[43] D. Lee and C. J. Tomlin, "Iterative method using the generalized Hopf formula: Avoiding spatial discretization for computing solutions of Hamilton-Jacobi equations for nonlinear systems," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 1486–1493, 2019.

[44] B. Xue, M. Fränzle, and N. Zhan, "Inner-approximating reachable sets for polynomial systems with time-varying uncertainties," *IEEE Transactions on Automatic Control*, vol. 65, no. 4, pp. 1468–1483, 2020.

[45] H. Yin, A. Packard, M. Arcak, and P. Seiler, "Finite horizon backward reachability analysis and control synthesis for uncertain nonlinear systems," in *2019 American Control Conference (ACC)*, pp. 5020–5026, July 2019.

[46] H. Yin, M. Arcak, A. K. Packard, and P. Seiler, "Backward reachability for polynomial systems on a finite horizon," *IEEE Transactions on Automatic Control*, pp. 1–1, 2021.

[47] P. Seiler, "SOSOPT: A toolbox for polynomial optimization," *ArXiv e-prints*, Aug 2013. arXiv:1308.1889.

[48] I. M. Mitchell, J. Budzis, and A. Bolyachevets, "Invariant, viability and discriminating kernel under-approximation via zonotope scaling," *ArXiv e-prints*, Jan 2019. arxiv:1901.01006.

[49] C. Belta, B. Yordanov, and E. A. Gol, *Formal Methods for Discrete-Time Dynamical Systems*, vol. 89. Springer, 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)*, pp. 1517–1522, Dec. 2017.

[51] S. Singh, M. Chen, S. L. Herbert, C. J. Tomlin, and M. Pavone, "Robust Tracking with Model Mismatch for Fast and Safe Planning: an SOS Optimization Approach," *arXiv preprint arXiv:1808.00649*, 2018.

[52] S. Kousik, S. Vaskov, M. Johnson-Roberson, and R. Vasudevan, "Safe Trajectory Synthesis for Autonomous Driving in Unforeseen Environments," in *ASME 2017 Dynamic Systems and Control Conference*, pp. V001T44A005–V001T44A005, American Society of Mechanical Engineers, 2017.

[53] H. Yin, M. Bujarbaruah, M. Arcak, and A. Packard, "Optimization based planner–tracker design for safety guarantees," in *2020 American Control Conference (ACC)*, pp. 5194–5200, 2020.

[54] S. W. Smith, H. Yin, and M. Arcak, "Continuous abstraction of nonlinear systems using sum-of-squares programming," in *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 8093–8098, 2019.

[55] Y. V. Pant, H. Yin, M. Arcak, and S. A. Seshia, "Co-design of control and planning for multi-rotor uavs with signal temporal logic specifications," *arXiv preprint, arXiv: 2009.14363*, 2020.

[56] P.-J. Meyer, H. Yin, A. H. Brodtkorb, M. Arcak, and A. J. Sørensen, "Continuous and discrete abstractions for planning, applied to ship docking," in *21st IFAC World Congress*, 2020.

[57] H. Yin, P. Seiler, and M. Arcak, "Backward reachability using integral quadratic constraints for uncertain nonlinear systems," *IEEE Control Systems Letters*, vol. 5, no. 2, pp. 707–712, 2021.

[58] T. I. Fossen, *Handbook of Marine Craft Hydrodyamics and Motion Control*. Wiley, 2011.

[59] S. Singh, A. Majumdar, J. Slotine, and M. Pavone, "Robust online motion planning via contraction theory and convex optimization," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5883–5890, May 2017.

[60] P.-J. Meyer and D. V. Dimarogonas, "Hierarchical decomposition of ltl synthesis problem for nonlinear control systems," *IEEE Transactions on Automatic Control*, vol. 64, no. 11, pp. 4676–4683, 2019.

[61] L. Lessard, B. Recht, and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, 2016.

[62] M. Fazlyab, M. Morari, and G. J. Pappas, "Safety Verification and Robustness Analysis of Neural Networks via Quadratic Constraints and Semidefinite Programming," p. arXiv:1903.01287, Mar. 2019.

[63] M. Fazlyab, A. Robey, H. Hassani, M. Morari, and G. Pappas, "Efficient and accurate estimation of Lipschitz constants for deep neural networks," in *Advances in Neural Information Processing Systems 32*, pp. 11427–11438, Curran Associates, Inc., 2019.

[64] P. Pauli, A. Koch, J. Berberich, and F. Allgöwer, "Training robust neural networks using Lipschitz bounds," p. arXiv:2005.02929, May 2020.

[65] H. Fang, Z. Lin, and M. Rotea, "On iqc approach to the analysis and design of linear systems subject to actuator saturation," *Systems & Control Letters*, vol. 57, no. 8, pp. 611 – 619, 2008.

[66] S. Tarbouriech, *Stability and Stabilization of Linear Systems with Saturating Actuators*. Springer Publishing Company, Incorporated, 2014.

[67] E. Summers and A. Packard, "L2 gain verification for interconnections of locally stable systems using integral quadratic constraints," in *49th IEEE Conference on Decision and Control (CDC)*, pp. 1460–1465, 2010.

[68] A. Iannelli, P. Seiler, and A. Marcos, "Region of attraction analysis with integral quadratic constraints," *Automatica*, vol. 109, p. 108543, 2019.

[69] S. Gowal, K. Dvijotham, R. Stanforth, R. Bunel, C. Qin, J. Uesato, R. Arandjelovic, T. Mann, and P. Kohli, "On the Effectiveness of Interval Bound Propagation for Training Verifiably Robust Models," p. arXiv:1810.12715, Oct. 2018.

[70] K. K. Kim, E. R. Patrón, and R. D. Braatz, "Standard representation and unified stability analysis for dynamic artificial neural network models," *Neural Networks*, vol. 98, pp. 251–262, 2018.

[71] H. Hindi and S. Boyd, "Analysis of linear systems with saturation using convex optimization," in *Proceedings of the 37th IEEE Conference on Decision and Control*, vol. 1, pp. 903–908 vol.1, 1998.

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

[73] H. Yin, A. Packard, M. Arcak, and P. Seiler, "Reachability analysis using dissipation inequalities for uncertain nonlinear systems," *Systems & Control Letters*, vol. 142, p. 104736, 2020.

[74] G. Zames and P. L. Falb, "Stability conditions for systems with monotone and slope-restricted nonlinearities," *SIAM Journal on Control*, vol. 6, no. 1, pp. 89–108, 1968.

[75] V. V. Kulkarni and M. G. Safonov, "All multipliers for repeated monotone nonlinearities," *IEEE Transactions on Automatic Control*, vol. 47, no. 7, pp. 1209–1212, 2002.

[76] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural networks : the official journal of the International Neural Network Society*, vol. 21 4, pp. 682–97, 2008.

[77] J. Schulman, S. Levine, P. Moritz, M. I. Jordan, and P. Abbeel, "Trust Region Policy Optimization," p. arXiv:1502.05477, Feb. 2015.

[78] S. Kakade, "A natural policy gradient," in *Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, NIPS'01, (Cambridge, MA, USA), p. 1531–1538, MIT Press, 2001.

[79] A. Alleyne, "A comparison of alternative intervention strategies for unintended roadway departure (urd) control," *Vehicle System Dynamics*, vol. 27, no. 3, pp. 157–186, 1997.

[80] T. Osa, J. Pajarinen, G. Neumann, J. Bagnell, P. Abbeel, and J. Peters, "An algorithmic perspective on imitation learning," *Foundations and Trends in Robotics*, vol. 7, no. 1-2, pp. 1–179, 2018.

[81] D. A. Pomerleau, *ALVINN: An Autonomous Land Vehicle in a Neural Network*, p. 305–313. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1989.

[82] W. Sun, A. Venkatraman, G. J. Gordon, B. Boots, and J. A. Bagnell, "Deeply aggrevated: Differentiable imitation learning for sequential prediction," in *International Conference on Machine Learning*, pp. 3309–3318, 2017.

[83] H. Daumé, J. Langford, and D. Marcu, "Search-based structured prediction," *Machine learning*, vol. 75, no. 3, pp. 297–325, 2009.

[84] H. Yin, P. Seiler, and M. Arcak, "Stability Analysis using Quadratic Constraints for Systems with Neural Network Controllers," p. arXiv:2006.07579, June 2020.

[85] P. Pauli, J. Köhler, J. Berberich, A. Koch, and F. Allgöwer, "Offset-free setpoint tracking using neural network controllers," *arXiv preprint arXiv:2011.14006*, 2020.

[86] M. Jin and J. Lavaei, "Stability-certified reinforcement learning: A control-theoretic perspective," *arXiv preprint arXiv:1810.11505*, 2018.

[87] M. Revay, R. Wang, and I. R. Manchester, "Convex sets of robust recurrent neural networks," *arXiv preprint arXiv:2004.05290*, 2020.

[88] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[89] X. Zhang, M. Bujarbaruah, and F. Borrelli, "Safe and near-optimal policy learning for model predictive control using primal-dual neural networks," in *2019 American Control Conference (ACC)*, pp. 354–359, 2019.

[90] B. Karg and S. Lucia, "Efficient representation and approximation of model predictive control laws via deep learning," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 3866–3878, 2020.

[91] S. Chen, K. Saulnier, N. Atanasov, D. D. Lee, V. Kumar, G. J. Pappas, and M. Morari, "Approximating explicit model predictive control using constrained neural networks," in *2018 Annual American Control Conference (ACC)*, pp. 1520–1527, 2018.

[92] M. Arcak, C. Meissen, and A. Packard, *Networks of dissipative systems: compositional certification of stability, performance, and safety.* Springer, 2016.