Contributions to the Statistical Foundation of Data-Driven Control

Alex Devonport

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2023-207
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2023/EECS-2023-207.html

August 10, 2023

Contributions to the Statistical Foundation of Data-driven Control

By

Alex R. Devonport

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Murat Arcak, Chair
Professor Costas Spanos
Professor Somayeh Sojoudi
Professor Majid Zamani

Summer 2023

Contributions to the Statistical Foundation of Data-driven Control

Abstract

Contributions to the Statistical Foundation of Data-driven Control

by

Alex R. Devonport

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Murat Arcak, Chair

we demonstrate several techniques to prove safety guarantees for robust control problems with statistical structure; that is, for data-driven dynamical modeling or verification problems where uncertainty is modeled by probability. These guarantees are probabilistic in nature, in accordance with the statistical nature of the uncertainty, and can be derived with limited model assumptions. Indeed, some of the techniques require no more than measurability. We focus on two data-driven control problems: estimation of forward reachable sets from data, and robust control of time- and frequency-domain models defined by a Gaussian process regression model. In the former, we apply scenario optimization and statistical learning theory to obtain probabilistic guarantees of accuracy and confidence with minimal system knowledge. In the latter, we apply the theory of suprema of Gaussian processes to establish high-probability regions of attraction, L2 gain bounds, and integral quadratic constraints for the uncertain system.

Dedicated to George Devonport; once known as “The Gas-Man” to his friends; known as “Dad” to the sons he loved dearly; and “Grandad” to all who cherish his memory.

Contents

List of Figures

List of Tables

Acknowledgments

It's true what they say: there's no way to enumerate all of the people who have helped you on the way to a PhD. Even if I could remember them all,[1] to properly express my gratitude would require a document with a length of a considerable portion of this dissertation. Wit that in mind, let me just hit the key points.

Let's begin with my academic mentors; the principle of which being my PhD advisor, Murat Arcak. There is a very real sense, which I won't elaborate on here, in which I would not have had a successful graduate career without Murat. He has always put a great deal of trust and good faith in me, and has allowed me a degree of academic freedom that many PhD students do not enjoy. I only hope that I was worth the risk. I am also an honorary member of Costas Spanos's lab, the RAISE (Research in AI for Sustainable energy) group. I am not strictly a co-advisee, but a close friend of RAISE, acting as the "voice of control-theoric reason" on occasions where it might otherwise be forgotten. I've had as many collaborations with the RAISE group as I have within my own. I also consider myself an honorary student of Majid Zamani, having spent a summer research internship at TUM with him and remaining in close contact with him and his group ever since. I would be remiss to forget my research advisors during my undergrad research at Arizona State University, professor Nathan Newman and staff scientist Nicholas Rizzo. They rightly saw that I had a keen academic interest, and cultivated it with care and kindness, though they misjudged the precise field of research—another story I won't elaborate on here.

I've also had the pleasure of working with a lot of collaborators during my research. As I mentioned, there's the RAISE group; particularly Hari Prasanna Das, Wendy Lin, Lucas Spangher, and others. Working with the RAISE group taught me a lot about how to apply control theory. Within our group I worked closely at times with Pierre-Jean Meyer, Galaxy Yin, and Adnane Saoud; working with them, I learned about many techniques in control theory that I wouldn't have otherwise, many of which (particularly reachability analysis) have been instrumental in my research.

What can I be but a reflection of my family? I won't enter into particular names, but each part of my family has instilled in me some of the qualities I needed to succeed. The Devonports of England use patience, kindness, and sometimes even intelligence to overcome all conditions that confront them. The Marcheses of America value art and science in equal measure, and treasure the unique.

I also relied heavily on the support—academic and moral—of my friends during my research. My partner Alex and I have a strong support group in our home town of Phoenix, Arizona; a subset of whom satisfy Rule 4 of a good group chat [92], and all of whom we don't see often enough. Closer to home, and in no particular order, are: Ryan Ly, Kelsey & Ingrid Ockert, Jaimie Swartz, the RAISE group, Nick Harris, Mark Wetzlinger, Mahmoud Khaled, Valerie Chen and Jasmine Ma (jointly we form „Die Drei"), Rosalyn Gardencourt, and other folks online too numerous to name. Like I said, the list is incomplete: you know where to direct any complaints of omission.

[1] I will have omitted some on accident; if that's you, I apologize.

Lastly, I'd like to acknowledge the following grants, which have provided most of my funding during my PhD career: ONR N00014-18-1-2209, AFOSR FA9550-18-1-0253, NSF CNS-2111688.

Chapter 1

Introduction

If you'll permit a little poetic license, control theory is that arcane calculus of feedback and equilibrium that binds computers to reality. To the prosaic, control theory is the engineering-oriented study of dynamical systems, centered around the question of how to adjoin to a physical system a compensator—often but not always an information processing system—that renders a stable, robust, and self-regulating interconnection. Data are a collected set of measurements, often of a real system but just as often of a high-fidelity simulation. I have no poetic interpretation for data, but its importance is self-evident if we want our theory to bear any resemblance to reality. From these rudiments we come to a surprisingly difficult question.

1.1 What do we mean when we say "data-driven control?"

We have good reasons to want a precise answer. A California redwood's worth of papers and articles use the term; a conference is named after it; and this dissertation contributes to its statistical foundations. It seems like a question that should be simple to answer, but if we try, we quickly find ourselves repeating history.

> Since feedback control seeks to reduce the effects of disturbances and plant uncertainty, the question of the difference between feedback control and adaptive control immediately arises.
>
> A meaningful definition of adaptive control ... is still lacking. However, there appears to be a consensus that a constant-gain feedback system is not an adaptive system.
>
> Åström and Wittenmark, *Adaptive Control* (1995) [8]

This quote, from Åström and Wittenmark's seminal book on adaptive control, describes the struggle that adaptive control theorists faced when trying to answer the question of what, precisely, constitutes an adaptive control system. It seems like a question that should be simple to answer, especially by the people who are supposed to be the experts; on the contrary, Åström and Wittenmark cite disagreements going back to 1961. Evidently no satisfactory answer had been given in the three decades between 1961 and 1995, and I would venture that no satisfactory answer has been given in the three decades (nearly) between 1995 and today. The problem is that if you answer "a control system that adapts to a changing environment," you deem every control system that employs feedback to be an adaptive control system. This is why constant-gain feedback is excluded by fiat in the quote: while adaptive by any reasonable definition of the word, it's simply not what people mean when they say "adaptive control".

Now consider the question, "What is a data-driven control system?" The obvious response, something along the lines of "a control system that uses data", has essentially the same problems as the obvious response for adaptive control systems. In fact, when it comes to constant-gain feedback, it's *exactly* the same problem: it doesn't seem like constant-gain feedback should be a data-driven control system, but it would be untrue to say that data doesn't play an integral role in its decision-making procedure. Moreover, there are many subfields of control theory, now considered classical, that could compete for the title of data-driven control; how could system identification, or adaptive control for that matter, be considered anything *but* data-driven? Putting philosophy aside, the simple fact is that these methods are not what data-driven control, in its modern meaning, is about.

In the last few years, papers on data-driven control tend to fall into two distinct categories. In the first category, we have papers that use a result known as "Willems's Fundamental Lemma" and its extensions, a line of work summarized in the survey paper [31]. The fundamental lemma allows for an exact representation of a dynamical system– originally a behavioral model but in modern treatments typically a state-space representation– using the data from a sufficiently information-rich sequence of trajectory data. From the perspective of this category of data-driven control literature, a data-driven controller is one derived from a system representation derived from extension of Willems's fundamental lemma. This definition is refreshingly direct, but too limited for our purposes, as there are many more ways to apply data to control problems than just Willems's fundamental lemma. In the second category, we have papers that use a model for system behavior, often a dynamical model or a value function, that can be modified using trajectory data [81, 50]. More often than not, these models are borrowed from the literature of statistics and machine learning– neural nets, Gaussian processes– distinguishing this line of work in technique, if not in spirit, from the classical methods of system identification, adaptive control, and approximate dynamic programming. In this category of work, a data-driven control system is one that makes use of machine learning methods, either in its construction or in its analysis. This is closer in spirit to what this dissertation investigates, particularly when it comes to GPs, but we will just as often be interested in the methods that lie on the other side of the nebulous boundary that divides machine learning and modern statistics.

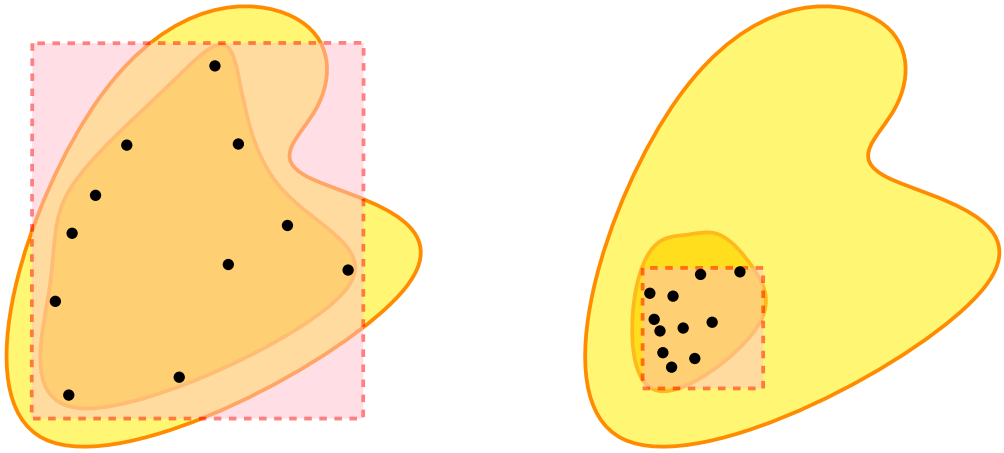

Figure 1.1: A reachable set (bright yellow) that we seek to estimate (red) from a set of independent simulation samples (black). Left: how do we ensure that our estimate will enclose a prescribed probability mass? Right: What is the likelihood of uninformative data?

Examples of Data-Driven Control

Choosing to learn from history, we will not endeavor to resolutely answer the question of what data-driven control is. Let us instead proceed by example with the following resume of problems that are popular both in the data-driven control literature and in practical control engineering: they are the core problems that motivate the work in this dissertation.

First is the problem of *data-driven reachability analysis*, shown in Figure 1.1, which is an important and widely-used heuristic in practical applications for verifying the safety of a control system. You can think of it as a *Monte Carlo* approach to safety verification. From the set of all possible system behaviors, we are are given a finite, random selection that we use to estimate the set of behaviors that we didn't see. We are not privy to any information about the dynamics that generate the behavior, as if the system is testable but inscrutable– as is all too often is in practice. Several questions immediately present themselves. To what degree is it possible at all to generalize from the finite point set to the infinite set of unseen behaviors? And if it is possible in some circumstances, what are the odds of getting an uninformative data set? From finite data alone, these question cannot be answered. In other words, there is some structural foundation that's missing from the problem as we currently have it.

Second is the problem of *safe exploration* of an unknown state-space system, shown in Figure 1.2. Here, we wish to ascertain the presence of, and extend, a *region of attraction*; a sphere of state space where we can always return to an equilibrium state. The objective is to use data collected inside the region of attraction to improve our knowledge of the system, make a new policy that extends the region of attraction, and then explore even further. But how do we construct a Lyapunov function from data alone? And without some kind of model

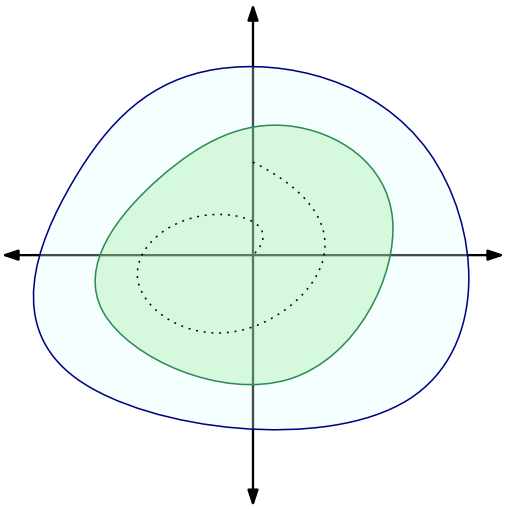

Figure 1.2: A region of attraction (blue) to be approximated (green) using a Lyapunov function derived from from trajectory data (grey). What is the likelihood that the estimate is really contained in the region of attraction?

structure, how do we assert its validity– in other words, that our estimate of the region of attraction is a good estimate?

Third is the problem of robustly estimating a transfer function, shown in Figure 1.3. Here, we are given frequency-domain data: a set of point evaluations of an unknown transfer function at a set of known frequencies. Constructing a transfer function that fits the data is a well known problem: a classical solution is Nevanlinna-Pick interpolation, which has multiple uses in control theory. The problem is that there are infinitely many transfer functions that interpolate a given set of frequency data: certainly some are more reasonable than others, but how do we make that decision? And supposing we have, and we design a controller to stabilize the system, how do we ensure that a large measure of that reasonable ensemble are stable– in other words, how do we ensure that we have robustly stabilized the system against our epistemic uncertainty?

These three examples share the following features:

1. The data "jump out of the system", playing a part not only in the control system, but in our design, and more crucially our *analysis*, of the system;

2. Information about the system, particularly quantitative model structure, is scarce;

3. As stated, there is insufficient structure to make a guarantee of safety.

The first two are likely inherent properties of data-driven control problems; but the third is unacceptable. However, the second seems to imply the third: if we have so little knowledge of the model, what other structure is to serve as the foundation for making safety guarantees?

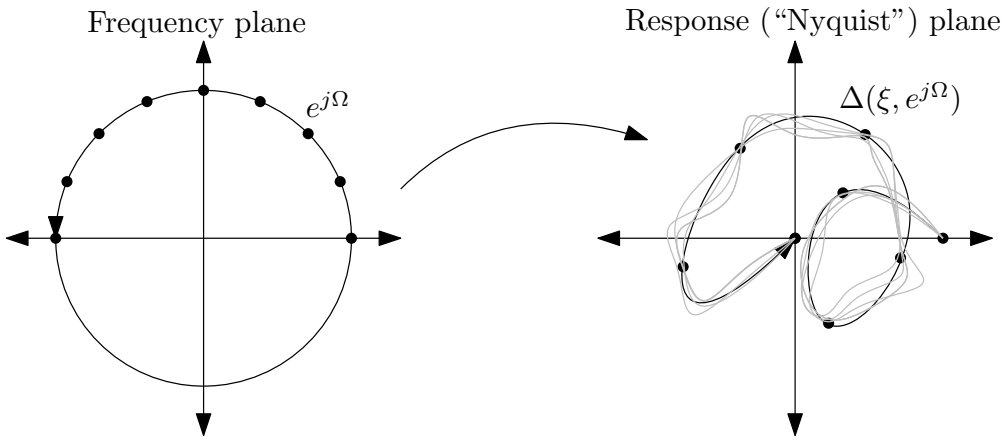

Figure 1.3: A set of frequency-domain data points (black) measured from an unknown transfer function (black line), and an ensemble of candidate transfer functions (grey lines) that fit the data. What is the likelihood that a fixed controller robustly stabilizes a reasonable ensemble?

Probabilistic Structure as the foundation for Probabilistic Guarantees

This dissertation explores how to use *probabilistic structure* as an addition to, or replacement of, model structure as a foundation for making safety guarantees in data-driven control. This amounts to treating data affected by the control system as random variables, or treating the entire system as a random function, or both. When the data or model becomes probabilistic, so do the safety certificates– reachable sets, regions of attraction, integral quadratic constraints– and so does the corresponding safety guarantee. Thus the problem of establishing safety guarantees becomes a statistical problem: the problem of managing risk under probabilistic uncertainty. As such, we say that these guarantees have a *statistical foundation*.

Generally, a probabilistic guarantee of safety makes two assertions: one about the accuracy of our claim of safety, and another about our confidence in the assertion of accuracy. The situation is similar to that of a meteorologist asserting a "95% chance of a 50% chance" of sunny skies: the latter percentage refers to the accuracy of the forecast, while the former percentage refers to the meteorologist's confidence that the forecast is valid in the first place. In the formal language of probability, "accuracy" is the assertion that the probability of an event that we deem to represent *safety* obtains a probability greater than a prescribed threshold: if S denotes this event, and P the probability measure corresponding to the problem's probabilistic structure, then the guarantee is an assertion that $P(S) \geq 1 - \epsilon$ for some prescribed $\epsilon \in (0, 1)$. The smaller ϵ is, the greater the probability mass of S, and therefore the more accurate our assertion of safety. On the other hand, "confidence" is the assertion that our data are sufficiently informative to make a valid claim about accuracy: if P_{data} is

the measure by which the data is distributed, and G is the event that the sampled data $x_1, \ldots, x_N$ are sufficiently rich to furnish our desired certificate of safety, we obtain a $1-\delta$ level of confidence if we can establish that $P^N_{\text{data}}(G) \geq 1-\delta$.

Depending on the information available to us, we may be able to eliminate uncertainty about the accuracy or confidence of our guarantee, effectively setting either ϵ or δ to zero.[1] Thus we encounter three possible forms for probabilistic guarantees:

- Accuracy-confidence guarantees, where $\epsilon, \delta > 0$. These are known in the literature of statistical learning theory as probably approximately correct (PAC) bounds, and we will generally refer to them by that name in this thesis. These bounds arise when we assume only qualitative knowledge about the control problem– such as measurability or analyticity– and acquire all quantitative knowledge through data.

- Accuracy-only guarantees, where $\delta = 0$. These bounds arise when we have quantitative knowledge about the in the control problem, e.g. knowing the pdf or having a bound on moments of a random variable. We will encounter these in Part II when we assume that the system dynamics function is distributed as a Gaussian process.

- Confidence-only guarantees, where $\epsilon = 0$. This type of bound asserts that, for sufficiently rich data, we obtain a non-stochastic safety guarantee; as such, they require stronger quantitative system knowledge than either of the other bounds. There is a body of literature that investigates confidence-only guarantees– particularly works that use results derived from [44] and its extensions– but confidence-only bounds do not appear in this dissertation, as our primary motivation for considering probabilistic structure is to minimize quantitative assumptions.

1.2 Outline

This dissertation comprises two parts, corresponding to two distinct data-driven control theory problems that can be endowed with probabilistic structure and probabilistic guarantees. Part I covers the probabilistic approach to data-driven reachability analysis. In the model-driven reachability, the standard safety guarantee is to assert that one's estimate contains the set of all possible evolutions: in part I, we develop methods that obtain analogous accuracy-confidence guarantees, where accuracy is determined with respect to a random variable supported on the set of all possible evolutions. There is a surprising amount of depth to this seemingly simple problem: the three parts of this chapter explore three distinct but synoptic ways to view the problem, each of which leads to different analytical techniques, different classes of estimator geometry, and different practical use cases. Chapter 3 takes the viewpoint of chance-constrained optimization, which leads to estimators and guarantees based on the theory of scenario optimization. Chapter 4 takes the viewpoint of classical

[1] Of course, with enough information we can set both to zero, but to do so is to recreate the condition of non-probabilistic, model-driven control.

frequentist statistical learning theory, which leads to estimators based on estimators and bounds constrained by Vapnik-Chervonenkis dimension. Chapter 5 takes the viewpoint of Bayesian PAC analysis, which leads to estimators based on prior and posterior distributions over estimator classes and guarantees based on the PAC-Bayes theorem.

Part II covers a data-driven approach to robustness analysis of control systems based on Gaussian Process dynamical models. Unlike data-driven reachability analysis, this thrust of data-driven control contains an explicit learning-based element. The goal of this thrust is to modify a standard Gaussian process regression model so as to be amenable to standard control-theoretical safety guarantees when used as a Bayesian model of a dynamical system. Chapter 8 explores the case of establishing a region of attraction for a nonlinear state-space system modeled by a vector-valued process. Here, we specialize to the case of a process whose realizations are polynomial functions that vanish at the equilibrium point: this polynomial structure allows us to use sum-of-squares programming to synthesize control policies and establish accuracy-confidence guarantees for a Lyapunov function that establishes an inner approximation of the region of attraction. Chapter 9 investigates Gaussian processes in the setting of robust control, namely as a transfer function model of an LTI uncertainty. This application requires that the realizations of the process must be complex-valued functions that inhabit the function space H_∞: we call this type of process an H_∞ process, and investigate some sufficient conditions for constructing them. We then apply H_∞ Gaussian processes to learning and robustness, showing how to estimate transfer functions from frequency-domain data and to prove accuracy-only guarantees that the uncertainty obeys an integral quadratic constraint.

The work in this dissertation is based on the papers [34, 39, 38, 37, 36], of all of which I am the primary author. This dissertation provides additional background, a unity of narrative that exists between the works but is not immediately clear from individual readings, and several significant technical advances not published elsewhere.

Part I

Data-Driven Reachability Analysis

Chapter 2

Background

The computation of reachable sets is an effective way to characterize and verify the behavior of safety-critical cyber-physical systems. However, many systems of practical interest possess high-dimensional, analytically intractable, and possibly unknown dynamics, which make the computation of safe sets with formal guarantees difficult or impossible. This applies in particular to cyber-physical systems that incorporate high-fidelity computational models. For example, suppose we wish to compute reachable sets for a system model that account for the possibility of missing deadlines. Computational models with enough fidelity to capture this phenomenon (such as a microarchitectural simulator) contain hybrid or discontinuous elements that are not amenable to the standard techniques of model-based reachability analysis. In cases like this we can employ a data-driven approach to reachability analysis instead. The data-driven approach uses a finite ensemble of sample trajectories to compute reachable set estimates which are guaranteed to achieve high accuracy in a probabilistic sense.

Unlike earlier results on data-driven reachability which incorporate data-driven elements into existing reachability approaches, e.g. [45, 71], we obtain an estimate directly from data, thus eliminating intermediate steps that may introduce conservatism, and reducing the number of assumptions imposed on the system. In addition to being applicable to any system which admits simulation, data-driven reachability is computationally straightforward: computation is typically dominated by an *a priori* known number of trajectory simulations which can be done in parallel.

We present three approaches to data-driven forward reachability analysis. The first approach, presented in Chapter 3, uses the tools of scenario optimization to construct reachable set estimates as approximate solutions to chance-constrained optimization problems. Chapter 3 is generalization of work begun in [34], with new examples and an examination of the approach from an algorithmic perspective. The second approach, presented in Chapter 4 and investigated in [39], uses the tools of statistical learning theory to derive probabilistic guarantees using combinatorial bounds on the range of estimator geometries permitted by a particular reachability algorithm. The third approach, presented in Chapter 5 and begun in [38], uses a Bayesian extension of PAC learning to derive probabilistic guarantees based on relative entropy arguments.

The Forward Reachability problem and its Probabilistic Extension

We consider a general dynamical system with state transition function $\Phi : \mathcal{X}_0 \times \mathcal{D} \to \mathbb{R}^{n_x}$ defined on an initial set $\mathcal{X}_0 \subseteq \mathbb{R}^{n_x}$ and a set of disturbances $\mathcal{D}$ comprising disturbance signals $d : [t_0, t_1] \to \mathbb{R}^{n_d}$. We assume, if the system has inputs, that a control policy has been selected, yielding an autonomous system subject to a disturbance made up of a combination of control deviation and exogenous disturbances.

In the forward reachability problem, we also consider a set $\mathcal{X}_0$ of initial states, and a set $\mathcal{D}$ of disturbances. The forward reachable set is then defined as

$$\mathcal{R} = \{\Phi(t_1; t_0, x_0, d) : x_0 \in \mathcal{X}_0, d \in \mathcal{D}\},$$

that is the set of all states to which the system can transition by time t_1 with initial states in $\mathcal{X}_0$ at initial time t_0 and disturbances in $\mathcal{D}$. The forward reachable set and the quantities used to define it are illustrated in Figure 2.1.

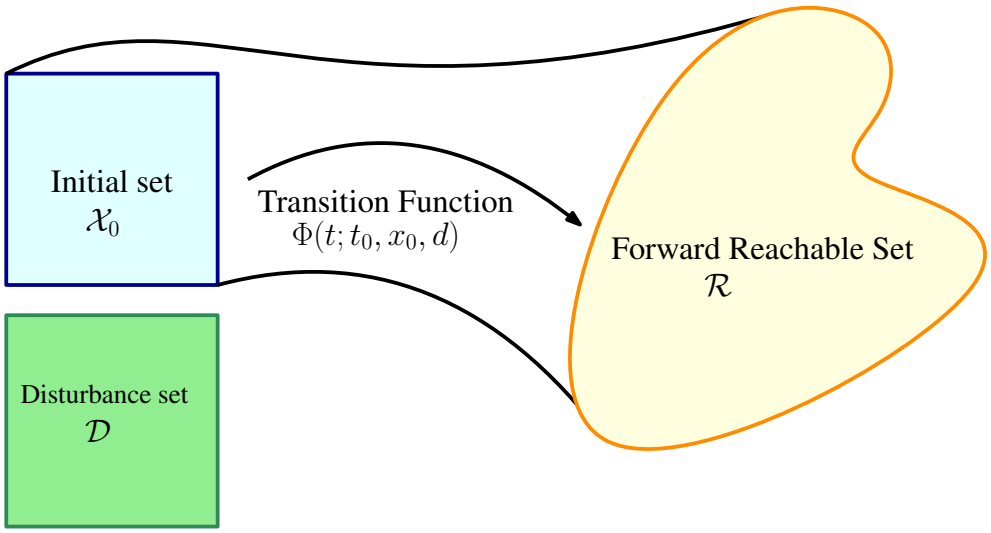

Figure 2.1: Illustration of a transition function, an initial set, a set of disturbances, and a time range giving rise to a forward reachable set.

We assume that our only access to Φ is through a black-box model: we can evaluate $\Phi(t_1; t_0, x_0, d)$ for a finite number of specific t_0, t_1, x_0, and d, but we have no other information about it. In particular, we do not have access to the side information typically required

for computationally efficient reachability analysis, such as Jacobian matrices, discrepancy functions, or mixed-monotone decomposition functions.

To estimate the reachable set by statistical means, we introduce probabilistic structure to the problem. Our approach is to define a random variable that takes values on the reachable set: by statistically estimating the values this random variable can take, we can estimate the reachable set. We first define auxiliary random variables on the initial states and disturbances. For the initial states, we introduce a random variable X_0, whose support[1] is $\mathcal{X}_0$. Similarly, we define a random variable D whose support is $\mathcal{D}$.[2] We may use X_0 and D to model genuinely random events in the problem at hand, but it is not necessary to do so: indeed, the problem need not possess probabilistic elements. The main purpose of X_0 and D is to provide a consistent rule for selecting sample points. Additionally, we require a mild technical assumption on the measurability of the transition function Φ: Specifically, Φ must be measurable in x_0 and d with respect to the probability measures defined by X_0 and D respectively.

Using these random variables, and the black-box model for Φ, we define the random variable $R = \Phi(t_1; t_0, x_0, d)$. This random variable is well-defined, since Φ is measurable with respect to X_0 and D. Additionally, since the support of X_0 and D are exactly $\mathcal{X}_0$ and $\mathcal{D}$ respectively, the support of R is exactly $\mathcal{R}$.

The Data-Driven Reachability Problem

Let P_R denote the probability measure with respect to R. Since $P_R(\mathcal{R}) = 1$, and $B \subseteq A$ implies $P_R(A) \geq P_R(B)$, the measure $P_R(A)$ of a set A acts as a probabilistic measure of how well A approximates $\mathcal{R}$ [33].[3] Since we cannot directly evaluate $P_R(\hat{\mathcal{R}})$ without more knowledge of Φ, we cannot directly verify that a candidate reachable set estimate attains high measure under P_R with complete certainty. However, under certain conditions we can use a finite number of evaluations of Φ to find an estimate that achieves a desired level of probabilistic accuracy with high confidence. This motivates the following variation of the reachability problem.

Problem 1 (Probabilistic Data-driven Reachability)**.** *Let X_0 and D be random variables supported on the set of initial states and set of disturbances respectively, and let $R = \Phi(t_1; t_0, X_0, D)$. Let $\epsilon \in (0, 1)$ denote an accuracy parameter, and $\delta \in (0, 1)$ a confidence parameter. Let $\mathcal{C} \subseteq 2^{\mathbb{R}^{n_x}}$ denote the class of admissible reachable set estimators.*

[1]The support of a random variable is the range of values it can assume. If a random variable has a probability density function, its support is the set of all points where the density function is nonzero.

[2]Since $\mathcal{D}$ is a set of functions, the random variable D is generally a function-valued random variable. For instance, D may be taken as a stationary Gaussian process such as n_d-dimensional Brownian motion. If the disturbance is assumed to be constant-valued, then D reduces an ordinary random vector on $\mathbb{R}^{n_d}$.

[3]The measure $P_R(A)$ has an intuitive interpretation: if we take a sample from X_0 and a sample from D and apply them to Φ, the probability that the output of Φ lies in A is $P_R(A)$.

Given a set of samples $r_i = \phi(t_1; t_0, x_{0i}, d_i)$, $i = 1, \ldots, N$, *where* $x_{01}, \ldots, x_{0N} \overset{i.i.d.}{\sim} X_0, d_1, \ldots, d_N \overset{i.i.d.}{\sim} D$, *compute a compact estimate* $\hat{\mathcal{R}} \in \mathcal{C}$ *of the reachable set such that*

$$P_R^N(P_R(\hat{\mathcal{R}}) \geq 1 - \epsilon) \geq 1 - \delta, \tag{2.1}$$

where P_R^N *denotes the product measure of* N *independent samples of the random variable* R.

The double probability inequality (2.1) is an accuracy-confidence guarantee that $\hat{\mathcal{R}}$ covers a certain amount of $\mathcal{R}$ measured in terms of probability. The guarantee makes two assertions:

- Accuracy: the inner inequality asserts that $\hat{\mathcal{R}}$ attains probability mass of at least $1-\epsilon$ under P_R.
- Confidence: The outer inequality asserts that $\hat{\mathcal{R}}$ attains $1-\epsilon$ accuracy with probability $1-\delta$ with respect to the samples $r_1, \ldots, r_N$. In other words, this inequality asserts the probability of observing an N-tuple of samples informative enough to satisfy the accuracy criterion is at least $1-\delta$.

A Reachable set estimate $\hat{\mathcal{R}}$ that provably solves Problem 1 is a probabilistically-certified estimate, suitable for probabilistic safety verification and controller synthesis [35]. The requirement that $\hat{\mathcal{R}}$ be compact precludes the trivial estimate $\hat{\mathcal{R}} = \mathbb{R}^{n_x}$; however, it does not stop the estimate from being conservative. To reduce conservatism, we will typically include some regularization on the volume of $\hat{\mathcal{R}}$. The algorithms presented in Part I can solve Problem 1 for arbitrary $\epsilon, \delta \in (0, 1)$, provided that N satisfies a known lower bound depending on ϵ, δ, and the complexity of $\mathcal{C}$ in a sense that will be made precise.

Symbols, Abbreviations, and notation used in Part I

symbol	definition
Reachability Analysis	
$\Phi(t_1; t_0, x_0, d)$	State transition function, evolving a state x_0 at time t_0 under disturbance d to a state at time t_1
$\mathcal{X}_0$	Set of initial states
$\mathcal{D}$	Set of disturbances
t_0, t_1	Initial and final times
$\mathcal{R}$	Forward reachable set
$\hat{\mathcal{R}}$	Approximation of forward reachable set
$\mathcal{C}$	class of admissible estimators; $\hat{\mathcal{R}} \in \mathcal{C}$. Also called "Concept class"
X_0	Random variable supported on the initial set
D	Random variable supported on the disturbance
R	Reachability random variable; $R = \Phi(t_1; t_0, X_0, D)$
$r_1, \ldots, r_N \overset{\text{i.i.d.}}{\sim} R$	N iid samples distributed according to R

P_R	Probability measure of the distribution of R
P_R^N	Probability measure of N iid samples from R
n_x	state dimension; $\mathcal{R} \subseteq \mathbb{R}^{n_x}$
Scenario Optimization	
g	Parameterized Estimator function: $\hat{\mathcal{R}}(\theta) = \{x : g(x,\theta) \leq 0\}$
Θ	Parameter space, $\theta \in \Theta \subseteq \mathbb{R}^{n_\theta}$
Vol	Volume proxy; Vol : $\Theta \to \mathbb{R}_+$
Probability	
$\mathbb{E}[\cdot]$	Expected value of a random variable
$\mathbb{P}(\cdot)$	Probability of an event defined in terms of random variables
$D_{KL}(P\|\|Q)$	Kullback-Leibler (KL) divergence from P to Q
$D_{ber}(p\|\|q)$	KL divergence between Bernoulli distributions with parameters p and q
F_1	CDF of the chi-square distribution with 1 degree of freedom
$\mathcal{X}$	Domain of X
PAC	Probably Approximately Correct
iid	Independent and Identically Distributed
ϵ,δ	accuracy and confidence parameters in PAC guarantees
P,Q	Prior and posterior probability measures on $\mathcal{C}$
W_P, W_Q	Parametric representations of P and Q
C_P, C_Q	Stochastic estimators: random variables on $\mathcal{C}$ distributed according to P, Q
$\bar{c}_Q$	"central concept" of the posterior measure Q
$\ell(c,x)$	statistical loss function comparing a concept c and a datum x
$r(c)$	risk: average of $\ell(c,x)$ for $x \sim X$
$\hat{r}(c)$	empirical estimate of $r(c)$ from data $x_1,\ldots,x_N$
r_Q	stochastic risk: average of $\ell(c,x)$ for $x \sim X$, $c \sim Q$
$\hat{r}_Q$	empirical estimate of r_Q from data $x_1,\ldots,x_N$
Christoffel Functions	
M_m, $\hat{M}_m$	Matrix of moments of degree $\leq m$ and its empirical estimate
$\hat{M}_{m,\sigma_0}$	Empirical moment matrix with diagonals modified by σ_0
$z_m(x)$	vector of monomials with degree $\leq m$ evaluated at point x
$\hat{\kappa}^{-1}(x)$	Polynomial empirical inverse Christoffel function evaluated at x
$\hat{\kappa}^{-1}(x)$	kernelized empirical inverse Christoffel function
$C(x)$	Christoffel-based support set estimator, output of Algorithms 4,5, and 6

Gaussian Processes	
m, k	prior mean and covariance functions
m_q, k_q	posterior mean and covariance functions
K	kernel Gramian matrix, $K_{ij} = k(x_i, x_j)$
k_D	vector of kernel evaluations on data, $(k_D(x))_i = k(x_i, x)$
$\mathcal{N}(\mu, \Sigma)$	Multivariate normal with mean μ and covariance Σ
$\mathcal{GP}(m, k)$	Gaussian process with mean and covariance functions m, k

Table 2.1: Important symbols and acronyms used in Part I.

Chapter 3

The Scenario Optimization Approach

A simple strategy for selecting a reachable set estimate $\hat{\mathcal{R}} \in \mathcal{C}$ from a class of admissible estimators $\mathcal{C}$ using data is to select the smallest set in $\mathcal{C}$ that contains all of the data points. Here, "small" can be with respect to a proxy for the volume, as directly computing the Lebesgue measure is not computationally feasible in general. This leads to the following heuristic:

1. Compute samples $r_1, \ldots, r_N \overset{\text{i.i.d.}}{\sim} R$ by sampling from X_0 and D and applying the black-box model for Φ.

2. Find the set $\hat{\mathcal{R}} \in \mathcal{C}$ that minimizes the volume proxy subject to the constraint that $r_1, \ldots, r_N \in \hat{\mathcal{R}}$.

Step 1 is parallelizable, since the samples r_i can be computed independently. Step 2 corresponds to a minimum-volume covering problem: special cases of this problem, such as the minimum-volume covering ellipsoid, have been studied extensively, and in many cases the problem can be solved by standard optimization packages. It sounds like a plausible strategy, and it has the advantage of being straightforward to compute, but is it enough to provably solve Problem 1? In other words, can this heuristic be endowed with a probabilistic guarantee of the accuracy and confidence of its output? Under the right conditions it can, and those conditions depend only on the nature of $\mathcal{C}$, and not on the nature of Φ outside of the measurability requirement mentioned above.

The step from a heuristic to an algorithm that provably solves Problem 1 is furnished by convex scenario optimization, a randomized approach to approximating feasible solutions of chance-constrained optimization problems. Convex scenario optimization replaces the probabilistic constraint of the chance-constrained problem with a set of sampled-data constraints. With a sufficient number of samples, the sampled-data solution satisfies the original probabilistic constraint with high confidence: we review this in Section 3.1. In Section 3.2, we show that the heuristic described above computes the solution to a scenario approximation of a convex chance-constrained problem under a suitable convexity assumption: the probabilistic guarantee of scenario optimization transfers to the heuristic, proving that it solves

Problem 1. The convexity assumption is on a parametric representation of $\mathcal{C}$: the reachable set estimates themselves are not required to be convex. In Section 3.3, we consider examples of scenario reachability for specific choices of $\mathcal{C}$ applied to a chaotic nonlinear system.

3.1 The Scenario Approach to Chance-Constrained Convex Optimization

Scenario optimization is a technique to approximately solve chance-constrained convex optimization problems. Specifically, we consider the problem

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & J(\theta) \\ \text{subject to} \quad & P_Z(g(Z,\theta) \leq 0) \geq 1-\epsilon \\ & \theta \in \Theta, \end{aligned} \tag{3.1}$$

where J and g are convex with respect to θ, P_Z is a probability measure corresponding to a random variable Z, $\epsilon \in (0,1)$, and $\Theta \in \mathbb{R}^{n_\theta}$ is convex and compact. The problem admits great freedom in Z: no specific assumptions are made on the nature of Z, and g does not need to be convex with respect to Z. Furthermore, we will ultimately not require specific knowledge about Z, just the ability to sample from it. Directly computing a solution to (3.1) is typically difficult due to the probabilistic constraint: it is difficult to verify outside of certain special cases, and its feasible set is generally not convex despite the given convexity assumptions.

To alleviate the difficulties caused by the probabilistic constraint, the technique of scenario optimization replaces it with a collection of N sampled constraints, called scenarios, where the random variable Z is replaced by a fixed sample $z \sim Z$. This leads to the scenario relaxation of the problem with N scenarios:

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & J(\theta) \\ \text{subject to} \quad & g(z_i,\theta) \leq 0,\ i = 1,\dots,N, \\ & \theta \in \Theta, \end{aligned} \tag{3.2}$$

where $z_1,\dots,z_N \overset{\text{i.i.d.}}{\sim} Z$. The scenario relaxation is easier to solve than the original problem for two important reasons. First, we no longer need to evaluate P_Z: we only need to be able to sample from Z to construct the scenario constraints before solving the problem. Second, the scenarios are convex constraints, meaning that the scenario relaxation is a convex optimization problem unlike the original problem.

In general we cannot be certain that a solution to the scenario relaxation is a feasible solution to the original problem, since we cannot evaluate P_Z. However, we can guarantee that the probability of satisfying the original constraint, with respect to the distribution of the samples $z_1,\dots,z_N$ used to construct the scenarios, can be made arbitrarily close to 1

when N is sufficiently large. The specific N depends on ϵ, n_θ, and the desired confidence of satisfying the constraint:

Theorem 1 ([83], Corollary 12.1)**.** *Let $\delta \in (0,1)$ denote a confidence parameter. If N is selected according to*

$$N \geq \frac{1}{\epsilon}\left(\frac{e}{e-1}\right)\left(\log\frac{1}{\delta} + n_\theta\right), \tag{3.3}$$

where e is the base of the natural logarithm, then a minimizer of (3.2), if it exists, is a feasible solution to (3.1) with probability $\geq 1-\delta$ with respect to the measure P_Z^N.

We can interpret the confidence parameter δ as the probability of selecting a sample $z_1, \ldots, z_N$ such that the solution of the scenario relaxation constructed from $z_1, \ldots, z_N$ does not satisfy the probabilistic constraint. For example, if we take $\delta = 10^{-9}$, then the event that we construct a scenario relaxation whose solution does not satisfy the original constraint is a "one in a billion" event. This theorem ensures that, for sufficiently large (but always finite) N, δ can be made arbitrarily small. Indeed, since δ only appears logarithmically in the sample bound (3.3), we can typically choose $\delta \leq 10^{-9}$ without greatly increasing the number of samples required to construct the scenario relaxation.

3.2 The Scenario Reachability Algorithm

The heuristic described in the beginning of this section, where we take samples from the reachable set and choose the smallest estimator in a set $\mathcal{C}$ that contains the samples, computes the solution of a scenario relaxation of a chance-constrained optimization problem for certain choices of $\mathcal{C}$. Specifically, we consider classes of sets of the form

$$\mathcal{C} = \{\ \{x \in \mathbb{R}^{n_x} : g(x,\theta) \leq 0\}\ \ \theta \in \Theta\}, \tag{3.4}$$

where $g : \mathbb{R}^{n_x} \times \Theta \to \mathbb{R}$ is convex in θ, and $\Theta \subseteq \mathbb{R}^{n_\theta}$ is convex and compact. The class $\mathcal{C}$ is a set of sublevel sets, one sublevel set $\{x \in \mathbb{R}^{n_x} : g(x,\theta) \leq 0\}$ for each $\theta \in \Theta$. We denote the sublevel set $\{x : g(x,\theta) \leq 0\}$ for a specific choice of θ as $\hat{\mathcal{R}}(\theta)$. We also suppose that we have a convex proxy $\mathrm{Vol} : \Theta \to \mathbb{R}$ for the volume of $\hat{\mathcal{R}}(\theta)$.

Now, consider the problem of finding the smallest set $\hat{\mathcal{R}} \in \mathcal{C}$, with respect to Vol, that satisfies the accuracy condition of the PAC bound (2.1), that is the smallest $\hat{\mathcal{R}}(\theta)$, with $\theta \in \Theta$, such that $P_R(\hat{\mathcal{R}}(\theta)) \geq 1-\epsilon$. The parameter corresponding to this set is the solution of the optimization problem

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \mathrm{Vol}(\theta) \\ \text{subject to} \quad & P_R(g(R,\theta) \leq 0) \geq 1-\epsilon \\ & \theta \in \Theta. \end{aligned} \tag{3.5}$$

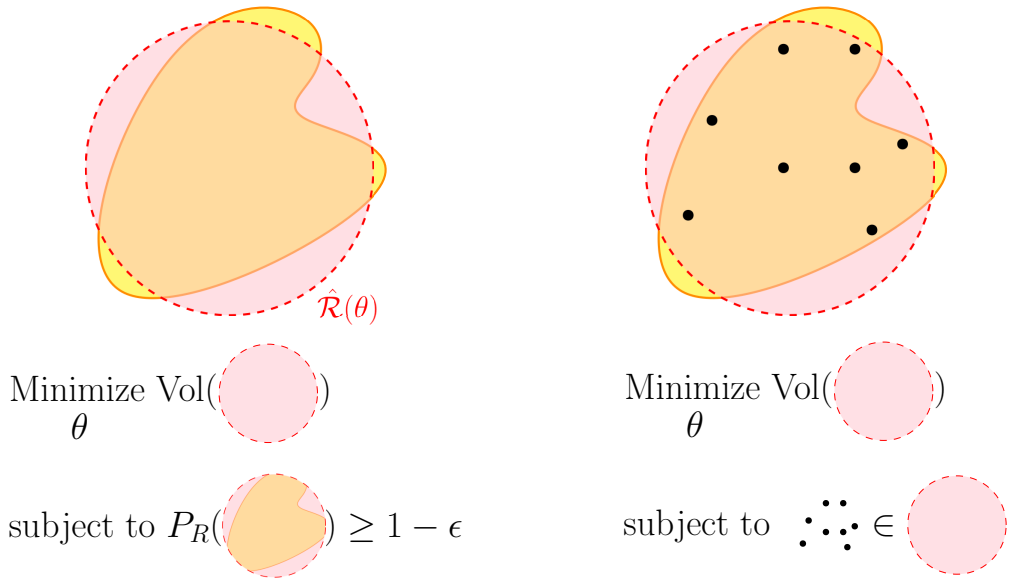

Figure 3.1: Illustrations of the probabilistic forward reachability problem as a chance-constrained optimization problem (left) and its scenario relaxation (right).

This is a chance-constrained convex optimization problem of the form (3.1), satisfying the same requirements on the objective and probabilistic constraint. In (3.5), the generic random variable Z is replaced by the random variable R defined on the reachable set: since no conditions were placed on the random variable in the probabilistic constraint, and we need only to be able to compute samples in order to construct the scenario relaxation, this approach is admissible for any measurable black-box model Φ. The scenario relaxation of the problem (3.5) with N scenarios is

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & \text{Vol}(\theta) \\ \text{subject to} \quad & g(r_i, \theta) \leq 0, \ i = 1, \ldots, N \\ & \theta \in \Theta, \end{aligned} \tag{3.6}$$

where $r_1, \ldots, r_N \overset{\text{i.i.d.}}{\sim} R$, that is $r_i = \Phi(t_1; t_0, x_0, d), \ i = 1, \ldots, N$ with $x_{01}, \ldots, x_{0N} \overset{\text{i.i.d.}}{\sim} X_0$, $d_1, \ldots, d_N \overset{\text{i.i.d.}}{\sim} D$. The solution to this problem is the minimum-volume set (with respect to the proxy Vol) that contains the sample points $r_1, \ldots, r_N$. The relationship between Problem 1, the chance-constrained problem (3.5), and its scenario relaxation (3.6) is illustrated in Figure 3.1.

Algorithm 1 describes the procedure of constructing samples $r_1, \ldots, r_N \overset{\text{i.i.d.}}{\sim} R$, and constructing and solving the scenario optimization problem (3.6). Since the sample size N used in Algorithm 1 satisfies the sample bound (3.3) for the ϵ and δ in the input, the conditions of Theorem 1 hold for the solution θ^* given as the output of the algorithm. This means that

Algorithm 1: Scenario Reachability.

Input: Black-box transition function model $\Phi(t_1; t_0, x_0, d)$; State dimension n_x; Parameter dimension n_θ; convex, compact parameter set $\Theta \subseteq \mathbb{R}^{n_\theta}$ Sublevel set function $g(x, \theta)$, convex for $\theta \in \Theta$; random variables X_0 and D supported on $\mathcal{X}_0$ and $\mathcal{D}$ respectively; time range $[t_0, t_1]$; accuracy and confidence parameters $\epsilon, \delta \in (0, 1)$.

Output: Parameter θ^* corresponding to a reachable set estimate $\hat{\mathcal{R}}(\theta^*)$ that solves Problem 1.

Set number of samples $N = \left\lceil \frac{1}{\epsilon} \frac{e}{e-1} \left(\log \frac{1}{\delta} + n_\theta \right) \right\rceil$;

for all $i \in \{1, \ldots, N\}$ **do**

  Take samples $x_{0i} \sim X_0$, $d_i \sim D$;

  Evaluate $r_i = \Phi(t_1; t_0, x_{0i}, d_i)$;

end

Solve the convex problem

$$\begin{aligned} \theta^* = \arg\min_{\theta} \quad & \mathrm{Vol}(\theta) \\ \text{subject to} \quad & g(r_i, \theta) \leq 0, \ i = 1, \ldots, N \\ & \theta \in \Theta, \end{aligned}$$

and return θ^*;

$P_R(\{x : g(x, \theta^*) \leq 0\}) \geq 1 - \epsilon$ with probability $1 - \delta$ with respect to the samples $r_1, \ldots, r_N$. If we take $\hat{\mathcal{R}}(\theta^*) = \{x : g(x, \theta^*) \leq 0\}$ as the estimate of the reachable set, then we have that

$$P_R^N(P_R(\hat{\mathcal{R}}(\theta^*)) \geq 1 - \epsilon) \geq 1 - \delta, \tag{3.7}$$

which is precisely the condition (2.1) required for $\hat{\mathcal{R}}(\theta^*)$ to solve Problem 1.

Proposition 1. *The reachable set estimate $\hat{\mathcal{R}}(\theta^*)$, where θ^* is the output of Algorithm 1, solves Problem 1.*

Informally, Algorithm 1 comprises two steps:

1. Compute samples $r_1, \ldots, r_N \overset{\text{i.i.d.}}{\sim} R$ by sampling from X_0 and D and applying the black-box model for Φ.
2. Find the set $\hat{\mathcal{R}}(\theta)$ that minimizes the volume proxy subject to the constraint that $r_1, \ldots, r_N \in \hat{\mathcal{R}}$.

This is precisely the heuristic described in the beginning of this section: Algorithm 1 and Proposition 1 show that this heuristic can in fact produce reachable set estimates with arbitrarily high probabilistic accuracy and confidence.

3.3 Examples of Scenario Reachability

The choice of the sublevel set function $g(x, \theta)$ determines the geometry of the reachable set estimates. In a sense, Algorithm 1 comprises a family of algorithms for estimating reachable sets from data, from which we fix a particular algorithm by choosing a particular g. The geometric properties of $\hat{\mathcal{R}}(\theta^*)$, the sample complexity of the algorithm, and the computational difficulty of solving the scenario relaxation (3.6) to find θ^* all depend strongly on the choice of g, so a meaningful computational analysis of scenario reachability is not possible without examining specific choices of g.

In addition to examining the computational details for specific g, we will demonstrate the performance of particular choices for g on a numerical example. The running example will be to compute a forward reachable set for the Lorenz system

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z \end{aligned} \tag{3.8}$$

where $x(t), y(t), z(t) \in \mathbb{R}$. We take $\sigma = 10$, $\beta = 8/3$, and $\rho = 28$. For these values, the system exhibits chaotic behavior: indeed, it is the classical Lorenz attractor. For the reachability problem, we take initial set $\mathcal{X}_0 = [0, 1]^3$, time range $[t_0, t_f] = [0, 100]$, and no disturbance. We will take the random variable X_0 to be the uniform distribution on $\mathcal{X}_0$, and use accuracy and confidence parameters $\epsilon = 0.05$, $\delta = 10^{-9}$. All reported computation times are with respect to a laptop with two 2.6 GHz physical cores running MATLAB. On this machine, one evaluation of the state transition function takes approximately 45 milliseconds.

Scenario Reachability with p-Norm Balls

In this example, we consider the estimation of reachable sets using p-norm balls. Specifically, we take

$$\hat{\mathcal{R}}(A, b) = \{x \in \mathbb{R}^{n_x} : \|Ax - b\|_p \leq 1\}, \tag{3.9}$$

where $A = A^\top \in \Theta_A \subseteq \mathbb{R}^{n_x \times n_x}$, $b \in \Theta_b \subseteq \mathbb{R}^{n_x}$, with Θ_A, Θ_b convex and compact,[1]

and $\|\cdot\|_p$ denotes the p-norm for $p \in [1, \infty) \cup \{\infty\}$. For a fixed p, the class of norm balls is of the form (3.4) for $\theta = (A, b)$, $\Theta = \Theta_A \times \Theta_b$, $g(x, A, b) = \|Ax - b\|_p - 1$. Since $\|Ax - b\|_p - 1$ is convex in (A, b), this class of sets is suitable for scenario reachability. For the volume proxy, we take $\mathrm{Vol}(A, b) = -\log\det A$. This function is directly proportional to the volume in the $p = 2$ case: by the equivalence of norms, it is proportional to an upper bound on the

[1]The compactness assumption is necessary to satisfy the compactness requirement of convex scenario optimization. However, Θ_A and Θ_b can typically be made large enough that its boundaries do not affect computations.

volume for general p. With this choice of g and Vol the scenario relaxation problem becomes

$$\begin{aligned} \underset{A,b}{\text{minimize}} \quad & -\log\det A \\ \text{subject to} \quad & \|Ar_i - b\| \leq 1, \ i = 1, \ldots, N \\ & A \in \Theta_A, b \in \Theta_b. \end{aligned}$$

This problem, which seeks to minimize a log determinant subject to norm constraints on A and b, is readily solved by standard optimization packages for any p. For $p = 2$ this reduces to the minimum-volume covering ellipsoid problem, for which there are more efficient specialized algorithms [84]. The parameter vector $\theta = (A, b)$ comprises $n_x(n_x + 1)/2$ parameters in A and n_x parameters in b, yielding $n_\theta = n_x(n_x + 3)/2$ and a sample size of

$$N = \left\lceil \frac{1}{\epsilon} \frac{e}{e-1} \left(\log \frac{1}{\delta} + \tfrac{1}{2}(n_x^2 + 3n_x) \right) \right\rceil \tag{3.10}$$

in Algorithm 1. This shows that the p-norm ball instance of the scenario reachability algorithm has quadratic sample complexity for fixed ϵ and δ.

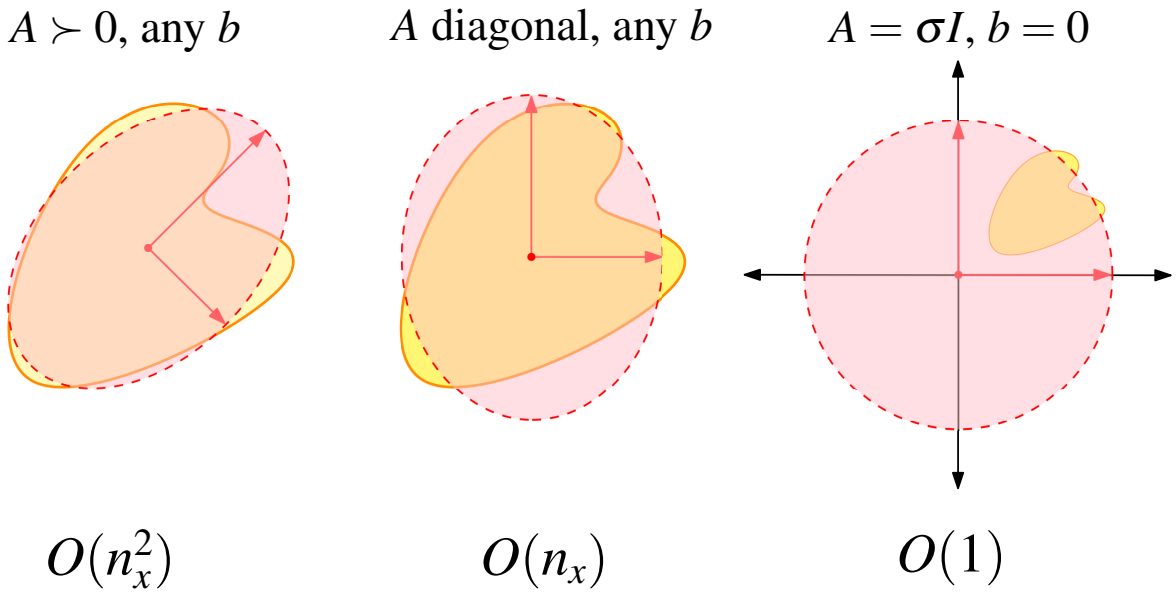

Figure 3.2: Illustration of the trade-off between orders of sample complexity and geometric restriction for three classes of sparsity for 2-norm ball scenario reachability.

Quadratic sample complexity does not always scale well enough to allow for practical computations. This can occur when the state dimension n_x is large, or when the Φ is computationally intensive model such as a high-fidelity simulation. For these cases, we can

reduce the order of the sample complexity by placing sparsity constraints on the parameters in A and b. For instance, restricting A to be diagonal reduces the number of parameters to $n_\theta = 2n_x$, yielding a sample complexity that scales linearly in n_x when applied to (3.3). A more severe example is to take $A = \sigma I$, $\sigma > 0$, and $b = 0$: with just a single scalar parameter regardless of state dimension, we attain constant sample complexity for fixed ϵ and δ. The price paid for this reduction in sample complexity is a restriction on the geometry of the norm balls. The diagonal A case leads to an "axis-aligned" geometry, where the semimajor axes (or their equivalents for $p \neq 2$) are parallel to the standard basis vectors of $\mathbb{R}^{n_x}$. The $A = \sigma I$, $b = 0$ case restricts the norm ball to be centered on the origin and have equal width in every direction. The trade-off between sample complexity and geometric restriction for these two cases, compared to the unconstrained case, is illustrated in Figure 3.2.

Adding sparsity constraints to A and b can also reduce the computational burden of solving the scenario relaxation, in some cases even allowing for an explicit solution. One such case occurs when $p = \infty$ and A is restricted to be diagonal: $\hat{\mathcal{R}}(A, b)$ takes the form of an axis-aligned hyperrectangle, whose center is the point b and whose side length along the i^{th} direction is $2/A_{ii}$. The solution to (3.6) corresponds to the smallest axis-aligned hyperrectangle containing $r_1, \ldots, r_N$, which can be computed directly by taking the componentwise minimum and maximum of $r_1, \ldots, r_N$. This instance of scenario reachability, which requires only $O(n_x)$ samples and directly computes the explicit solution to (3.6), is described in Algorithm 2.

Algorithm 2: Scenario reachability with axis-aligned hyperrectangles.

Input: Black-box transition function model $\Phi(t_1; t_0, x_0, d)$; State dimension n_x; random variables X_0 and D supported on $\mathcal{X}_0$ and $\mathcal{D}$ respectively; time range $[t_0, t_1]$; accuracy and confidence parameters $\epsilon, \delta \in (0, 1)$.

Output: Parameters A, b corresponding to an axis-aligned hyperrectangle (expressed as an ∞-norm ball) that solves Problem 1.

Set number of samples $N = \left\lceil \frac{1}{\epsilon} \frac{e}{e-1} \left(\log \frac{1}{\delta} + 2n_x \right) \right\rceil$;
for all $i \in \{1, \ldots, N\}$ **do**
  Take samples $x_{0i} \sim X_0$, $d_i \sim D$;
  Evaluate $r_i = \Phi(t_1; t_0, x_{0i}, d_i)$;
end
Let $\underline{r} = \min_i r_i$, $\overline{r} = \max_i r_i$, where min and max denote componentwise minimum and maximum;
Let $b = (\overline{r} + \underline{r})/2$, $a = (\overline{r} - \underline{r})/2$;
return $A = \mathrm{diag}(a)^{-1}$, b;

Turning to the numerical example, we apply the $p = 2$ norm ball instance of Algorithm 1 and to the forward reachability problem defined for the Lorenz system. For this instance, we consider three cases of sparsity: no constraint, A diagonal, and $A = \sigma I$, $b = 0$. For $\epsilon = 0.05$ and $\delta = 10^{-9}$, the unconstrained case uses $N = 941$ samples, the diagonal A uses $N = 846$

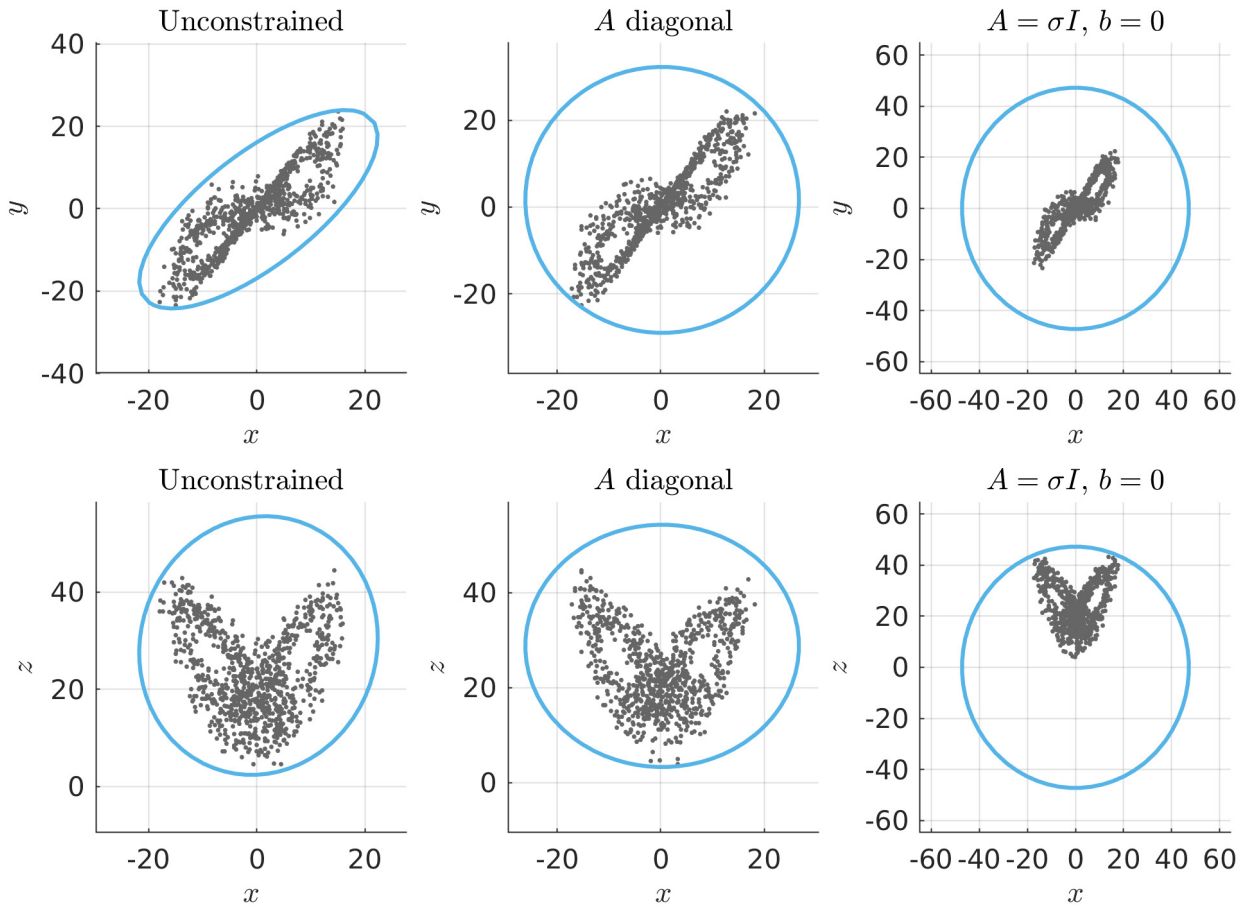

Figure 3.3: Reachable set estimates produced by 2-norm scenario reachability for the Lorenz system for three cases of sparsity in A and b, projected onto the (x, y) plane (top row) and (x, z) plane (bottom row).

samples, and the $A = \sigma I$, $b = 0$ case uses $N = 688$ samples. The reachable sets produced by these three cases of Algorithm 1, projected onto the (x, y) and (x, z) planes, are shown in Figure 3.3: they took 70 seconds, 59 seconds, and 45 seconds to compute, respectively.

Scenario Reachability with Basis Functions

Since $g(x, A, b) = \|Ax - b\|_p - 1$ is convex with respect to x as well as θ, the reachable set estimates made by scenario reachability with this choice of g are always convex. Since reachable sets are often nonconvex, this restriction can introduce unwanted conservatism. A simple way to construct a sublevel set function $g(x, \theta)$ that is convex in θ but not x is to select a finite set of basis functions $f_1(x), \ldots, f_m(x) : \mathbb{R}^{n_x} \to \mathbb{R}$ not convex in x and take g to be the weighted sum $g(x, \theta) = \sum_{i=1}^{m} \theta_i f_i(x)$. Since this g is convex – indeed, affine – in the parameter vector $\theta = (\theta_1, \ldots, \theta_m)$, it is a suitable function for scenario reachability, provided that we can also find a suitable convex volume proxy, and where we can ensure that the reachable set estimate is compact. In this section, we will use this approach to construct an instance of scenario reachability that estimates the reachable set using a union of cells taken from a partition of a compact region of the state space.

For this approach, we assume that $\mathcal{R}$ is known to be contained in a compact subset $A \subseteq \mathbb{R}^{n_x}$.[2] We then select a partition of A into m cells, that is a collection of sets $A_1, \ldots, A_m$ such that $\bigcup_{i=1}^{m} A_i = A$ and $A_i \cap A_j = \emptyset$ for $i \neq j$. Let $\mathbb{1}\{A_i\}$ denote the zero-one indicator function for the set A_i, so that $\mathbb{1}\{A_i\}(x) = 1$ if $x \in A_i$ and $\mathbb{1}\{A_i\}(x) = 0$ otherwise. Additionally, let $\mathbb{1}\{A^c\}$ denote the zero-one indicator function for A^c, the set-theoretic complement of A with respect to $\mathbb{R}^{n_x}$. We use these indicators as the basis functions for g, taking

$$g(x, \theta) = \mathbb{1}\{A^c\}(x) + \sum_{j=1}^{m} \theta_j \mathbb{1}\{A_j\}(x). \tag{3.11}$$

The reachable set estimates $\hat{\mathcal{R}}(\theta) = \{x \in \mathbb{R}^{n_x} : g(x, \theta) \leq 0\} \subseteq A$ generated by this g are unions of the partition cells: $A_j \in \hat{\mathcal{R}}(\theta)$ if and only if $\theta_j \leq 0$. We also set $\Theta = [0, 1]^m$: while respecting this constraint, $A_j \in \hat{\mathcal{R}}(\theta)$ if and only if $\theta_j = 0$.

The exact volume of $\hat{\mathcal{R}}(\theta)$ is $\sum_{j=1}^{m} \mathbb{1}\{\theta_j = 0\} V_{A_j}$, where V_{A_j} is the volume of the cell A_j. To derive a convex volume proxy from this expression, we remove the V_{A_j} factors and replace $\mathbb{1}\{\theta_j = 0\}$ with the convex approximation $1 - \theta_j$, yielding

$$\mathrm{Vol}(\theta) = \sum_{j=1}^{m} 1 - \theta_j. \tag{3.12}$$

We may further simplify Vol by removing the constant factors, since they do not affect the outcome of the optimization problem. With this choice of g and Vol, the scenario relaxation

[2] If such a region is not known in advance, we can use a data-driven estimate. For example, we could use the output of Algorithm 2 as the region A: the union-of-cells estimate for the reachable set would then act as a refinement of the hyperrectangle estimate.

is

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & -\sum_{j=1}^{m} \theta_j \\ \text{subject to} \quad & \mathbb{1}\left\{A^c\right\}(x) + \sum_{i=1}^{m} \theta_i \mathbb{1}\left\{A_i\right\}(x) \leq 0, \ i = 1, \ldots, N \\ & \theta \in [0,1]^m. \end{aligned} \tag{3.13}$$

The scenario relaxation (3.13) can be solved directly. To satisfy the scenario constraints, we set $\theta_j = 0$ for all j such that $r_i \in A_j$ for at least one $i \in \{1, \ldots, N\}$. To minimize the objective while respecting $\theta \in [0,1]^m$, we set $\theta_j = 1$ for all other j. The reachable set estimate $\hat{\mathcal{R}}(\theta^*)$ corresponding to the solution θ^* to (3.13) is the union of the cells A_j that contain one or more of the samples r_i. This is in fact the minimum-volume union of cells that contains $r_1, \ldots, r_N$, so the convex volume proxy does not introduce any conservatism. The scenario reachability algorithm based on the partition $A_1, \ldots, A_m$ is described in Algorithm 3.

Algorithm 3: Scenario reachability with a state-space partition.

Input: Black-box transition function model $\Phi(t_1; t_0, x_0, d)$; State dimension n_x; Partition $A_1, \ldots, A_m$ of a region $A \subseteq \mathbb{R}^{n_x}$ such that $\mathcal{R} \subseteq A$; random variables X_0 and D supported on $\mathcal{X}_0$ and $\mathcal{D}$ respectively; time range $[t_0, t_1]$; accuracy and confidence parameters $\epsilon, \delta \in (0,1)$.

Output: Parameters $\theta_1, \ldots, \theta_m$ corresponding a union of cells A_j such that $A_j \in \hat{\mathcal{R}}(\theta)$ if and only if $\theta = 0$.

```
Initialize θ_j = 1, j = 1, ..., m. Set number of samples N = ⌈(1/ε)(e/(e−1))(log(1/δ) + m)⌉;
for all i ∈ {1, ..., N} do
    Take samples x_0i ~ X_0, d_i ~ D;
    Evaluate r_i = Φ(t_1; t_0, x_0i, d_i);
    for all j ∈ {1, ..., M} do
        if r_i ∈ A_j, set θ_j = 0.
    end
end
Return θ_1, ..., θ_m;
```

To demonstrate partition-based scenario reachability, we apply it to the Lorenz reachability problem. We take A to be the hyperrectangle $A = [-30, 30] \times [-30, 30] \times [0, 50]$, and take the partition of A to be the grid with 10 sides along each dimension. This yields $m = 1000$, which for $\epsilon = 0.05$, $\delta = 10^{-9}$ yields a sample size of $N = 32296$ in Algorithm 3. The output of Algorithm 3 for this problem, which took 36 minutes to compute, is shown in Figure 3.4.

The partition-based instance of scenario reachability can produce arbitrarily fine estimates of the reachable set, depending on how refined the partition is. For example, when

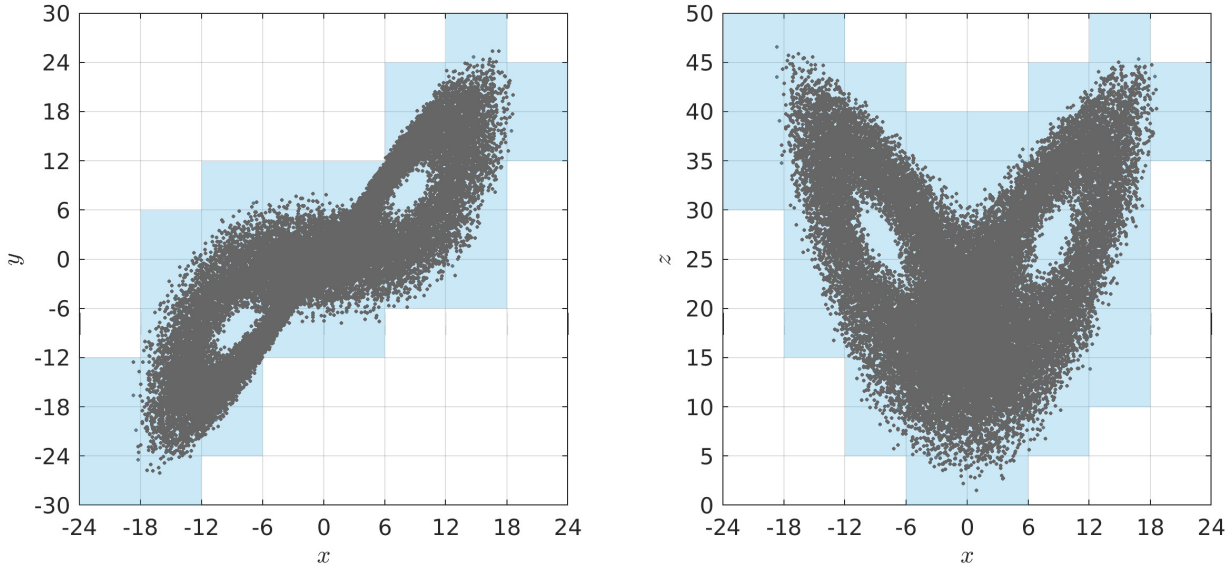

Figure 3.4: Reachable set (blue) computed by Algorithm 3 for the Lorenz system using a grid-based partition and $N = 32296$ samples (grey), projected onto the (x, y) and (x, z) planes.

A is a hyperrectangle as in the numerical example, we can use a regular grid with ℓ grid cells along each dimension as the partition. The accuracy of the partition will increase as ℓ increases, but the number of required number of samples increases at a polynomial rate in ℓ and at an exponential rate in n_x, as there are ℓ^{n_x} parameters.

3.4 Alternatives to Convex Scenario Optimization

From an algorithmic standpoint, the scenario approach can be applied to any chance-constrained optimization problem, whether it satisfies the convexity conditions or not: the difficulty is proving that solving the scenario relaxation is not just a heuristic, but indeed provides feasible solutions to the original problem with arbitrarily high probability. The theory of scenario optimization is concerned with proving this for various classes of chance-constrained problems, and for variants of the basic scenario relaxation described in Section 3.1.

The earliest and strongest result in this theory is the probabilistic guarantee for convex chance-constrained problems [18]. However, recent developments in the theory of scenario optimization allow for similar guarantees for chance-constrained problems that are not convex [23, 21]. However, the level of probabilistic accuracy guaranteed by the non-convex bound is *data-dependent* and cannot be computed until after after the scenario relaxation

has been solved. This means that a sample complexity bound cannot be established in advance, as it can be for convex scenario optimization: the only way to achieve a certain level of probabilistic accuracy is to solve the scenario relaxation multiple times for an increasing number of samples until the data-dependent bound attains the desired accuracy. While this "wait-and-judge" approach to scenario optimization does not have a pre-specified stopping time, it can work well in practice [22].

3.5 Limitation of the Scenario Approach

The scenario approach to reachability possesses a great deal of flexibility in the types of reachable set estimates it can use, which allows it to provide probabilistic guarantees of accuracy and confidence for many data-driven reachability algorithms. We have seen how it can provide such guarantees for methods based on minimum-volume covering ellipsoids, axis aligned rectangles, and grid-based state space partitions. However, there are several limitations to this approach.

The first limitation is that the scenario approach is essentially limited to data-driven algorithms of the minimum-volume covering type that admit a convex volume proxy. Any algorithm which is not based on the heuristic of finding a set that covers a collection of samples with minimal volume cannot be endowed with a probabilistic guarantee by the scenario approach. There are useful algorithms which are not based on this heuristic, one of which we will see in the next section.

The second limitation is that even if an algorithm is based on this heuristic, the class $\mathcal{C}$ of estimators it considers must:

1. be parameterized by a finite number of parameters;
2. admit a convex volume proxy $\mathrm{Vol}(\theta)$;
3. admit a representation as the zero-sublevel sets of a function $g(x, \theta)$ convex in θ.

These requirements rule out all nonparametric estimators, as well as several parametric classes of estimators that are popular in reachability analysis. The class of zonotopes is one example: they satisfy the first two requirements, but do not admit a convenient convex parameterization.

3.6 Conclusion

The core message of this chapter is an affirmation that the problem laid out in Chapter 2 is achievable: that computationally straightforward algorithms for estimating forward reachable sets from data can be endowed with guarantees of accuracy and confidence that hold with arbitrarily high probability. Our strategy so far has been to rely on the perspective of chance-constrained optimization and the scenario approach. This strategy allows us to prove

probabilistic safety guarantees for algorithms based on the heuristic of finding a minimum-volume cover for a set of points, but does not cover algorithms based on other principles. In the next two chapters we will meet a class of algorithms, based on various types of *Christoffel functions*, which are very effective at estimating reachable sets from data but nevertheless do not follow the volume-minimizing heuristic. In order to extend our ability to prove safety guarantees to a more general class of data-driven algorithms, and to methods based on Christoffel functions in particular, we require a technique with greater generality than the scenario approach.

Chapter 4

The PAC Approach

So far, our analysis of data-driven reachability has been based on a re-framing of the probabilistic variant of the forward reachability problem as a chance-constrained optimization problem, which has allowed us to leverage the tools of scenario optimization theory. In this chapter we consider a different framing, based on the notion of empirical risk minimization from statistical learning theory, that addresses some of the limitations of the scenario approach.

An empirical risk minimization problem, illustrated in Figure 4.1, comprises:

1. a collection of random observations $x_1 \ldots, x_N$ that are independent and identically distributed according to an unknown probability measure P_X;
2. a domain $\mathcal{X}$ and a class of sets $\mathcal{C} \subseteq 2^{\mathcal{X}}$, called the concept class;
3. a loss function $\ell : \mathcal{C} \times \mathcal{X} \to \mathbb{R}$.

In this framework, learning takes place by using the observations $x_1, \ldots, x_n$ to select a concept $c \in \mathcal{C}$. We would like to select the concept which minimizes the statistical risk $r(c) = \mathbb{E}\left[\ell(c, X)\right]$, where X is distributed according to P_X, but this is not possible since P_X is not known. We instead select a concept which minimizes empirical estimate $\hat{r}(c)$ of the risk. Under certain conditions on ℓ and $\mathcal{C}$, the difference between $r(c)$ and $\hat{r}(c)$ can be uniformly bounded with respect to C with high probability by any constant $\epsilon > 0$ for a sufficiently large N. A sufficient condition on ℓ is that it take on only the values zero and one, a condition which holds for the loss function appropriate for reachability analysis. A sufficient condition on $\mathcal{C}$ for this to be possible is that the Vapnik-Chervonenkis (VC) dimension[1] of $\mathcal{C}$ be finite.

The uniform bound on empirical risk described above is of the accuracy-confidence type described in Chapter 1, and is known in statistical learning theory as a Probably Approxi-

[1] The VC dimension is a combinatorial measure of the expressive power of a class of sets in terms of that class's ability to distinguish arbitrary points. We will not define the VC dimension here: for details, and rules to compute the VC dimension of several useful concept classes, we refer to [83, Chapter 10].

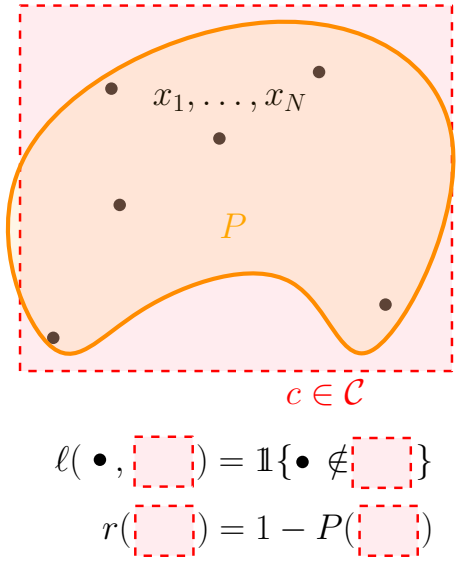

Figure 4.1: An empirical risk minimization problem: finding a concept c minimizing the empirical risk $\hat{r}(c)$ defined by data $x_1, \ldots, x_N$. Here, the loss is zero for points inside the concept, and one otherwise. P denotes the support of X.

mately Correct (PAC) bound.[2] For an appropriate choice of loss function, the PAC bound implies an accuracy-confidence bound on the probability mass covered by $\hat{\mathcal{R}}$: We call this approach to solving the Problem 1 the *PAC approach*. Unlike the scenario approach, the PAC approach places no restriction on how $\mathcal{C}$ is parameterized, nor on how $\hat{\mathcal{R}}$ is selected from $\mathcal{C}$. This means that the approach can be applied to any data-driven reachability technique that draws $\hat{\mathcal{R}}$ from a class of finite VC dimension.

To demonstrate how the PAC approach works, we will spend this chapter developing, and proving a probabilistic safety guarantee, for a data-driven reachability algorithm that uses sublevel sets of the inverses of *Christoffel functions* to estimate the reachable set. These sublevel sets make excellent estimators, but since the algorithm that derives them is not of the minimum-volume covering form, the scenario approach does not apply. However, the class of Christoffel function sublevel sets has finite VC dimension, so we are still able to apply the PAC approach. In Section 4.1 we review the family of Christoffel functions and their inverses; in Section 4.2 we introduce the algorithm to transform reachability data into a Christoffel-based estimate of the reachable set; and in Section 4.3 we give a more detailed review of PAC bounds and statistical learning theory, and prove the probabilistic

[2]The bound, and corresponding statistical framework, were invented and named by Leslie Valiant [85]. The name "Probably Approximately Correct" is an keen example of the British tendency to understate, for no stronger guarantee can be made under the assumptions of Valiant's framework.

guarantee for the Christoffel-based reachability algorithm. Finally, in Section 4.4 we apply the reachability algorithms to three numerical examples which demonstrates the ability of the algorithm to provide accurate estimations of complicated reachable set geometries.

4.1 Christoffel Functions

Given a finite measure μ on $\mathbb{R}^{n_x}$ and a positive integer m, the Christoffel function of order k is defined as the ratio

$$\kappa(x) = \frac{1}{z_m(x)^\top M^{-1} z_m(x)}, \tag{4.1}$$

where M is the matrix of moments

$$M = \int_{\mathbb{R}^{n_x}} z_m(x) z_m(x)^\top d\mu(x)$$

and $z_m(x)$ is the vector of monomials of degree $\leq m$. We assume throughout that M is positive definite, ensuring that M^{-1} exists. The Christoffel function has several important application in approximation theory, where its asymptotic properties are used to prove the regularity and consistency of Fourier series of orthogonal polynomials. For our purposes, it is more convenient to use the *inverse Christoffel function*

$$\kappa(x)^{-1} = z_m(x)^\top M^{-1} z_m(x), \tag{4.2}$$

which is a polynomial of degree $2m$. In Problem 1, and more generally in the problem of estimating a probability distribution from samples, μ is a probability measure which we do not *a priori* know– namely, the measure P_R over the reachable set. In this case, we instead use an empirical estimate of P_R constructed from a collection of independently and identically distributed (iid) samples r_i, $i = 1, \ldots, N$ samples from P_R, namely

$$\hat{P}_R = \frac{1}{N} \sum_{i=1}^{N} \delta_{r_i},$$

where δ_x is the *Dirac measure* satisfying $\int f(y) d\delta_x(y) = f(x)$. The measure $\hat{P}_R$ itself defines a Christoffel function, whose inverse

$$\begin{aligned} C(x) = \hat{\kappa}^{-1}(x) &= z_m(x)^\top \hat{M}^{-1} z_m(x) \\ &= z_m(x)^\top \left(\frac{1}{N} \sum_{i=1}^{N} z_m(r_i) z_m(r_i)^\top \right)^{-1} z_m(x), \end{aligned} \tag{4.3}$$

is called the *empirical inverse Christoffel function*. The matrix $\hat{M}$ is positive definite (and hence $\hat{M}^{-1}$ exists) if $N \geq \binom{n_x+m}{n_x}$ and the r_i do not all belong to the zero set of a single degree m polynomial.

4.2 Christoffel Function Level Sets as Reachable Set Approximations

Christoffel functions serve as approximations, to the degree possible by a polynomial, the measures that define them. This means that level sets of Christoffel functions provide tight approximations of the support of the measure, provided that the right level parameter can be found. This support-approximating quality has motivated the use of Christoffel functions in several statistical applications, such as density estimation [60, 61] and outlier detection [7].

Algorithm 4: Data-driven reachable set estimation by a sublevel set of an empirical inverse Christoffel function.

Input: Transition function Φ of a system with state dimension n; random variables X_0 and D defined on $\mathcal{X}_0$ and $\mathcal{D}$ respectively; time range $[t_0, t_1]$; probabilistic guarantee parameters ϵ and δ; Christoffel function order m.

Output: Set $\hat{\mathcal{R}}$ representing an ϵ-accurate reachable set estimate with confidence $1 - \delta$.

1 Set number of samples

$$N = \left\lceil \frac{5}{\epsilon} \left(\log \frac{4}{\delta} + \binom{n_x + 2m}{n_x} \log \frac{40}{\epsilon} \right) \right\rceil. \tag{4.4}$$

;

2 **for all** $i \in \{1, \ldots, N\}$ **do**

3 Take iid samples $x_{0,i}$ and d_i from X_0 and D respectively;;

4 evaluate $r_i = \Phi(t_1; t_0, x_{0,i}, d_i)$.;

5 **end**

6 Compute the matrix $\hat{M}^{-1}$ and level parameter α, where

$$\hat{M} = \frac{1}{N} \sum_{i=1}^{N} z_m(r_i) z_m(r_i)^\top,$$
$$\alpha = \max_{i=1,\ldots,N} z_m(r_i)^\top \hat{M}^{-1} z_m(r_i).$$

;

7 Record the set

$$\hat{\mathcal{R}} = \{x \in \mathbb{R}^n : z_m(x)^\top \hat{M}^{-1} z_m(x) \leq \alpha\}$$

as the reachable set estimate.;

Additionally, the level sets have been shown, using the plug-in approach [29], to converge exactly to the support of the distribution (in the sense of Hausdorff measure) when the

degree of the polynomial approaches infinity, and when the true probability distribution is available [61]. When the true probability distribution is *not* known, as is the case here, we can use the empirical Christoffel function instead. This is the essential idea behind Algorithm 4, which computes an empirical inverse Christoffel function $C(x)$ and a level parameter $\alpha \in \mathbb{R}$, and returns the sublevel set $\hat{\mathcal{R}} = \{x \in \mathbb{R}^{n_x} : C(x) \leq \alpha\}$ as a proposed solution to Problem 1. Of course, since we rely on the empirical inverse Christoffel function of a fixed degree, constructed from finite data, the convergence results cited above do not apply to Algorithm 4. This motivates us to look for an accuracy-confidence bound; while the scenario approach is not able to provide such a guarantee, the PAC approach is up to the task.

4.3 Classical PAC Analysis

PAC bounds originate in study of empirical risk minimization problems in statistical learning theory. Our strategy to prove a PAC bound for Algorithm 4 is to express Problem 1 as an empirical risk minimization problem and to then apply the tools of statistical learning theory.

In empirical risk minimization, the objective is to match a concept $c \subseteq \mathcal{X}$ from a pre-specified concept class $\mathcal{C} \subseteq 2^{\mathcal{X}}$ to an unknown random variable X supported on $\mathcal{X}$ using only a finite set of iid observations $x_1, \ldots, x_N$ of X. How well a concept matches X is quantified by the statistical risk $r(c) = \mathbb{E}[\ell(c, X)]$ defined by a loss function $\ell : \mathcal{C} \times \mathcal{X} \to \mathbb{R}_+$ and the unknown measure P_X: a lower risk indicates a better match. Since we do not know P_X, we cannot directly evaluate the statistical risk. However, we can use the empirical risk $\hat{r}(c) = \frac{1}{N}\sum_{i=1}^{N} \ell(c, x_i)$ as a proxy for the true risk, and select a concept to match the data on the basis of minimizing the empirical risk.

Whether empirical risk minimization actually selects a concept with low risk depends on how much $\hat{r}(c)$ differs from $r(c)$. A classical PAC bound provides a bound on the difference $r(c) - \hat{r}(c)$, or the absolute difference, that holds with high probability. We use the following result from [5], which gives a quantitative sample bound that depends on the Vapnik-Chervonenkis (VC) dimension [88] of the concept class. The VC dimension of a concept is a combinatorial measure of its complexity based on the expressiveness of its concepts.

Lemma 1 ([5], Corollary 4)**.** *Let $\mathcal{C}$ be a concept class of sets with VC dimension $\leq d$, and let $\ell : \mathcal{C} \times \mathcal{X} \to \{0, 1\}$ denote a $\{0, 1\}$-valued loss function. If*

$$N \geq \frac{5}{\epsilon}\left(\log\frac{4}{\delta} + d\log\frac{40}{\epsilon}\right), \tag{4.5}$$

and if $\hat{r}(c) = 0$, then $P_X^N(\{x_1, \ldots, x_N : r(c) \leq \epsilon\}) \geq 1 - \delta$.

A concept class with higher VC dimension generally provides greater-fidelity estimates than one with lower VC dimension, but is also more prone to overfitting: informally, this is

the reason why a concept class with higher VC dimension requires a larger sample bound for the same accuracy and confidence than one with lower VC dimension.

To apply Lemma 1, we must show that the sublevel sets of a polynomial empirical inverse Christoffel function belong to a concept class of bounded VC dimension. One such class is the class of superlevel sets of degree $2k$ polynomials: the following Lemma from [42], provides a bound on the VC dimension.

Lemma 2 ([42], Theorem 7.2)**.** *Let V be a vector space of functions $g : \mathbb{R}^n \to \mathbb{R}$ with dimension v. Then the class of sets $Pos(V) = \{ \{x : g(x) \geq 0\}, g \in V\}$ has VC dimension $\leq v$.*

The PAC bound, and hence the validity of Algorithm 4 follows from Lemmas 2 and 1 by framing the support estimation problem as one of empirical risk minimization.

Theorem 2. *The support set estimate produced by Algorithm 4, that is the set $\{x \in \mathcal{X} : C(x) \leq \alpha\}$ where $C(x) = z_m(x)^\top \hat{M}_{m,\sigma_0}^{-1} z_m(x)$, $\alpha = \max_i C(x_i)$, satisfies the PAC bound $P_R^N(\{x_1, \ldots, x_N : P_R(\{x \in \mathcal{X} : C(x) \leq \alpha\}) \geq 1 - \epsilon\}) \geq 1 - \delta$, and thereby solves Problem 1 with parameters ϵ, δ.*

Proof. Let $\mathcal{C} = \text{Pos}(\mathbb{R}[x]_{2m}^n)$, and $\ell(c, x) = \mathbb{1}\{x \notin c\}$. Note that the set $\{x \in \mathbb{R}^n : C(x) \leq \alpha\}$ is a member of $\text{Pos}(\mathbb{R}[x]_{2m}^n)$, since it can be expressed as $c = \{x \in \mathbb{R}^n : \alpha - C(x) \geq 0\}$. Since the dimension of $\mathbb{R}[x]_{2m}^{n_x}$ is $\binom{n_x+2m}{n_x}$, the VC dimension of $\text{Pos}(\mathbb{R}[x]_{2m}^{n_x})$ is $\leq \binom{n_x+2m}{n_x} = v$ by Lemma 2. For $\ell(c, x) = \mathbb{1}\{x \notin c\}$, the statistical risk is $r(c) = \mathbb{E}[\mathbb{1}\{x \notin c\}] = 1 - P_X(c)$, and its empirical counterpart is $\hat{r}(c) = \sum_{i=1}^N \mathbb{1}\{x_i \notin c\}$. The empirical risk is zero for any set c that encloses $x_1, \ldots, x_N$. The set $\{x \in \mathbb{R}^n : C(x) \leq \alpha\}$ encloses $x_1, \ldots, x_N$ by construction, meaning that $\hat{r}(\{x \in \mathbb{R}^n : C(x) \leq \alpha\}) = 0$. By applying Lemma 1 for this choice of $\mathcal{C}$, ℓ, and m, we find that if $N \geq \frac{5}{\epsilon}\left(\log\frac{4}{\delta} + \binom{n_x+2m}{n_x}\log\frac{40}{\epsilon}\right)$, then $P_R^N\left(\{x_1, \ldots, x_N\} : 1 - P_R(\{x \in \mathbb{R}^n : C(x) \leq \alpha\}) \leq \epsilon\}\right) \geq 1 - \delta$. Since Algorithm 4 selects N to be the smallest integer such that $N \geq \frac{5}{\epsilon}\left(\log\frac{4}{\delta} + \binom{n_x+2m}{n_x}\log\frac{40}{\epsilon}\right)$, it follows that the stated PAC bound holds for the output of Algorithm 4. □

Since Algorithm 4 isn't based on a minimum-volume covering heuristic, it isn't immediately clear whether or not its approximations will be conservative; in fact, as the sublevel set of a polynomial, it's not even clear that $\hat{\mathcal{R}}$ will be bounded. Fortunately, Algorithm 4 always produces bounded estimates, since $\hat{\mathcal{R}}$ is a sublevel set of the sum-of-squares polynomial $z(x)^\top \hat{M}^{-1} z(x)$. Furthermore, the level parameter α can equivalently be defined as the solution to the optimization problem

$$\begin{aligned} \arg\min_{\alpha > 0} \quad & \alpha \\ \text{subject to} \quad & z_k(x^{(i)})^\top M^{-1} z_k(x^{(i)}) \leq \alpha, \ i = 1, \ldots, N. \end{aligned}$$

In this problem, α acts as a penalty term for the volume of the sublevel set, since the volume increases monotonically with increasing α.

Remark 1. *In some reachability problems, we are only interested in computing a reachable set for a subset of the state variables. For example, suppose the state is* $(x_1, \ldots, x_n) \in \mathbb{R}^n$*, and we wish to verify a safety specification involving only the states* $x_1, \ldots, x_m$*, where* $m < n$*: a reachable set for the states* $x_1, \ldots, x_m$ *would suffice for this problem. In cases like this, Algorithm 4 can be modified to use only the first* m *elements of the samples* r_i*. The output of the algorithm is then an empirical inverse Christoffel function with domain* $\mathbb{R}^m$ *whose sublevel set* $\hat{\mathcal{R}}$ *estimates the reachable set for the reduced set of states. In the sequel, we refer to this application of Algorithm 4 as the* reduced-state variant *of Algorithm 4.*

4.4 Examples

This section demonstrates Algorithm 4's ability to make accurate estimates of forward reachable sets with three numerical examples. We demonstrate how the parallel nature of the algorithm can be leveraged to improve computation times by running all experiments on two computing platforms: (i) a laptop with 4 2.6 GHz cores; and (ii) an instance of the AWS EC2 computing platform `c5.24xlarge`, a virtual machine with 96 3.6 GHz cores.

Chaotic Nonlinear Oscillator

The first example is a reachable set estimation problem for the nonlinear, time-varying system with dynamics

$$\begin{aligned} \dot{x} &= y \\ \dot{y} &= -\alpha y + x - x^3 + \gamma \cos(\omega t), \end{aligned} \tag{4.6}$$

with states $x, y \in \mathbb{R}$ and parameters $\alpha, \gamma, \omega \in \mathbb{R}$. This system is known as the *Duffing oscillator*, a nonlinear oscillator which exhibits chaotic behavior for certain values of α, γ, and ω, for instance

$$\alpha = 0.05, \quad \gamma = 0.4, \quad \omega = 1.3. \tag{4.7}$$

The initial is the interval such that $x(0) \in [0.95, 1.05]$, $y(0) \in [-0.05, 0.05]$, and we take X_0 to be the uniform random variable over this interval. The time range is $[t_0, t_1] = [0, 100]$.

We use Algorithm 4 to compute a reachable set for (4.6) using an order $k = 10$ empirical inverse Christoffel function with accuracy and confidence parameters $\epsilon = 0.05$, $\delta = 10^{-9}$. With these parameters, (4.4) states that $N = 156,626$ samples are required to ensure that Theorem 2 holds for the reachable set estimate. Total computation times for this example were 39 minutes on the laptop, and 41 seconds on `c5.24xlarge`.

Figure 4.2 shows the reachable set estimate for the Duffing oscillator system with the problem data given above, and the point cloud of $156,626$ samples used to compute the empirical inverse Christoffel function and the level parameter α. The reachable set estimate is neither convex nor simply connected, closely following the boundaries of the cloud of points and excluding an empty region within the cloud of points.

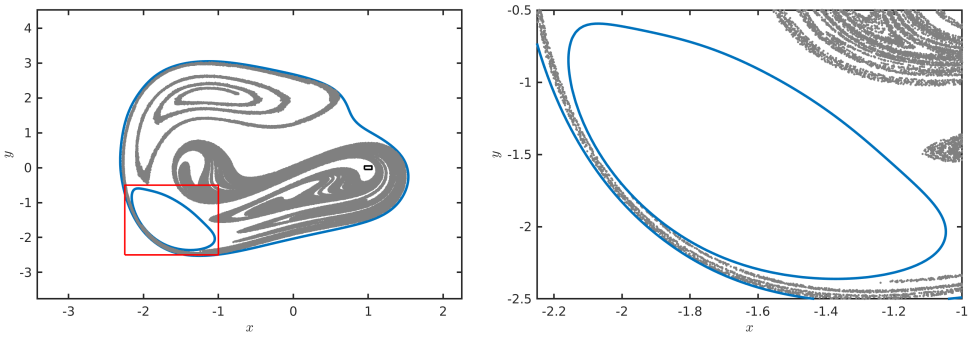

Figure 4.2: *Left*: reachable set estimate for the Duffing oscillator system (blue contour), the cloud of 156,626 samples used to compute the empirical inverse Christoffel function (grey points), and the initial set (black box). *Right*: enlarged version of the region in the left plot enclosed by the red box, showing the region excluded from the reachable set.

To experimentally verify that the assertion of Proposition 2 holds for the reachable set estimate, we compute an *a posteriori* estimate of the accuracy of the empirical inverse Christoffel function sublevel set. To do this, we first compute a new set of sample points of size N_{ap}. Denoting by N_{out} the number of new samples that lie outside of the reachable set estimate, we can compute the empirical accuracy of a reachable set approximation as $1 - N_{out}/N_{AP}$. We use $N_{AP} = 46{,}052$ sample points to make the *a posteriori* estimate. This sample size ensures that a one-sided Chernoff bound holds, which guarantees that empirical accuracy is within 1% of the true with 99.99% confidence. The *a posteriori* empirical accuracy computed with this sample is $1 - (2 \times 10^{-5})$, ensuring that the true accuracy of the reachable set estimate is at least $0.99 - 2 \times 10^{-5}$ with 99.99% confidence. This is well in excess of the 0.95 accuracy guaranteed by Theorem 2.

Planar Quadrotor Model

The next example is a reachable set estimation problem for horizontal position and altitude in a nonlinear model of the planar dynamics of a quadrotor used as an example in [66, 17]. The dynamics for this model are

$$\begin{aligned} \ddot{x} &= u_1 K \sin(\theta) \\ \ddot{h} &= -g + u_1 K \cos(\theta) \\ \ddot{\theta} &= -d_0\theta - d_1\dot{\theta} + n_0 u_2, \end{aligned} \tag{4.8}$$

where x and h denote the quadrotor's horizontal position and altitude in meters, respectively, and θ denotes its angular displacement (so that the quadrotor is level with the ground at $\theta = 0$) in radians. The system has 6 states, which we take to be x, h, θ, and their first

derivatives. The two system inputs u_1 and u_2 (treated as disturbances for this example) represent the motor thrust and the desired angle, respectively. The parameter values used (following [17]) are $g = 9.81$, $K = 0.89/1.4$, $d_0 = 70$, $d_1 = 17$, and $n_0 = 55$. The set of initial states is the interval such that

$$\begin{aligned} x(0) &\in [-1.7, 1.7], & \dot{x}(0) &\in [-0.8, 0.8], \\ h(0) &\in [0.3, 2.0], & \dot{h}(0) &\in [-1.0, 1.0], \\ \theta(0) &\in [-\pi/12, \pi/12], & \dot{\theta}(0) &\in [-\pi/2, \pi/2], \end{aligned}$$

the set of inputs is the set of constant functions $u_1(t) = u_1$, $u_2(t) = u_2$ $\forall t \in [t_0, t_1]$, whose values lie in the interval

$$u_1 \in [-1.5 + g/K, 1.5 + g/K], \quad u_2 \in [-\pi/4, \pi/4],$$

and we take X_0 and D to be the uniform random variables defined over these intervals. The time range is $[t_0, t_1] = [0, 5]$. We take probabilistic parameters $\epsilon = 0.05$, $\delta = 10^{-9}$. Since the goal of this example is to estimate a reachable set for the horizontal position and altitude only, we are interested in a reachable set for a subset of the state variables, namely x and h. As mentioned in Remark 2, Algorithm 4 can be used to estimate a reachable set for x and h in two ways: we can either compute a Christoffel function estimate for the reachable set and take the "shadow projection" of the estimate onto x and h, or we could compute a Christoffel function estimate for x and h directly using the reduced-state variant of Algorithm 4 with the (x, h) components of the reachable set data. To compare the relative accuracy and computational expense of these methods, we compute a reachable set estimate for (x, h) using both methods.

Figure 4.3 shows the reachable set estimates computed using both methods using order $k = 4$ inverse empirical Christoffel functions. Both reachable estimates turn out to be similar, though the estimate using the modification of Remark 2 is slightly tighter and significantly less computationally expensive. Running Algorithm 4 with the full state dimension $n = 6$ and order $k = 4$ with the ϵ and δ above requires $N = 2{,}009{,}600$ samples: using the reduced-state variant brings the effective state dimension to $n = 2$, and the sample size to $N = 32{,}292$. The computation times in the full-state case were 77 minutes on the laptop and 2 minutes on `c5.24xlarge`; in the reduced-state case, computation times were 78 seconds on the laptop and 2 seconds on `c5.24xlarge`. This shows that Algorithm 4's ability to work on subsets of the state space can speed up computations in cases where only a subset of state variables are of interest.

Monotone Traffic Model

The final example is a special case of a continuous-time road traffic analysis problem used as a reachability benchmark in [28, 65, 40]. This problem investigates the density of traffic on a single lane over a time range over four periods of duration T using a discretization of the cell

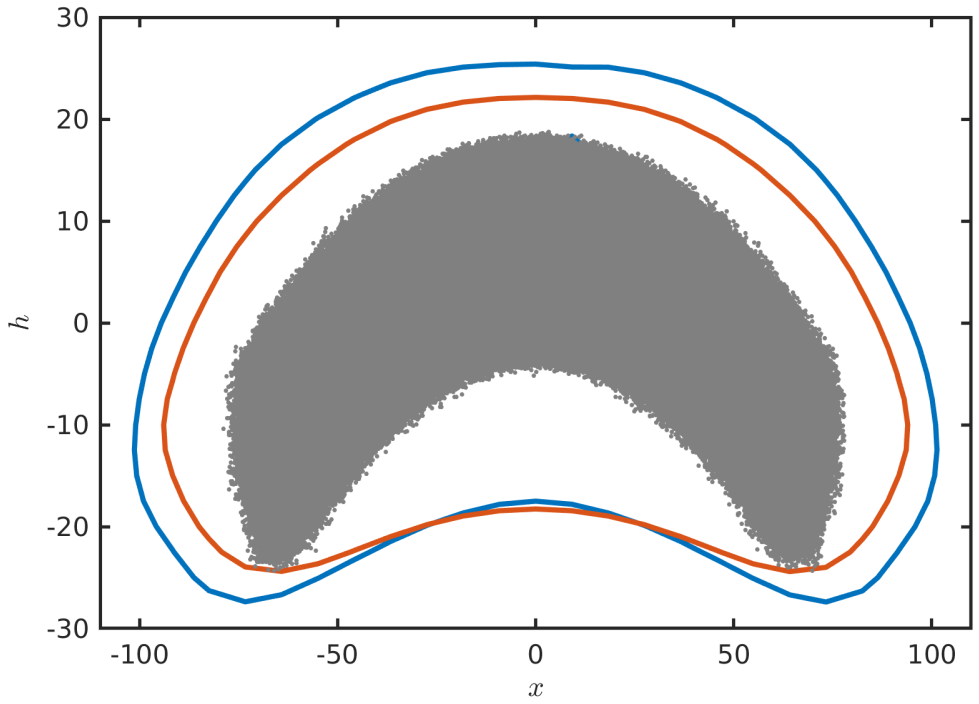

Figure 4.3: Reachable set estimates for the horizontal position and altitude of the planar quadrotor model, computed by projecting the output of Algorithm 4 onto (x, h) (blue) and using the modification of Algorithm 4 mentioned in Remark 2, where the algorithm is run using only the (x, h) components of the data (orange).

transmission model that divides the road into n equal segments. The spatially discretized model is an n-dimensional dynamical system with states $x_1, \ldots, x_n$, where x_i represents the density of traffic in the i^{th} segment. Traffic enters segment through x_1 and flows through each successive segment before leaving through segment n. The state dynamics are

$$\begin{aligned}
\dot{x}_1 &= \frac{1}{T}\left(d - \min(c, vx_1, w(\overline{x} - x_2))\right) \\
\dot{x}_i &= \frac{1}{T}\big(\min(c, vx_{i-1}, w(\overline{x} - x_i)) \\
&\quad - \min(c, vx_i, w(\overline{x} - x_{i+1}))\big), \quad (i = 2, \ldots, n-1) \\
\dot{x}_n &= \frac{1}{T}\left(\min(c, vx_{n-1}, w(\overline{x} - x_n)/\beta) - \min(c, vx_n)\right),
\end{aligned} \tag{4.9}$$

where v represents the free-flow speed of traffic, c the maximum flow between neighboring segments, $\bar{x}$ the maximum occupancy of a segment, and w the congestion wave speed. The input u represents the influx of traffic into the first node. For the reachable set estimation

problem, we use a model with $n = 6$ states, and take $T = 30$, $v = 0.5$, $w = 1/6$, and $\bar{x} = 320$. The initial set is the interval such that $x_i(0) \in [100, 200]$, $i = 1, \ldots, n$, the set of disturbances is the set of constant disturbances with values in the range range $d \in [40/T, 60/T]$, and X_0 and D are the uniform random variables over these sets. The time range is $[t_0, t_1] = [0, 4T]$.

The system dynamics (5.24) are *monotone*, or order-preserving, meaning that if two initial conditions $x^{(1)}(0)$, $x^{(2)}(0)$ and disturbances $d^{(1)}, d^{(2)}$ satisfy $x^{(1)}(0) \leq x^{(2)}(0)$ (where $\leq$ is the standard partial order) and $d^{(1)}(t) \leq d^{(2)}(t),\ t \in [0, T]$, then $x^{(1)}(T) \leq x^{(2)}(T)$. This monotonicity allows for a convenient interval over-approximation of the reachable set. If $\underline{x}$, $\overline{x}$ are the lower and upper bounds of the interval of initial states, and $\underline{d}$, $\overline{d}$ are the lower and upper bounds on the values admitted by the disturbance signal, then $[\Phi(t_1; t_0, \underline{x}, \underline{d}), \Phi(t_1; t_0, \overline{x}, \overline{d})]$ is the smallest interval that contains the entire reachable set. While this over-approximation is easy to compute, and the best possible over-approximation by an interval, it is in general a conservative over-approximation because reachable set may only occupy a small volume of the interval. Since the empirical Inverse Christoffel function method can accurately detect the geometry of the reachable set, we use this method to compare the shape of the reachable set to the best interval over-approximation. In particular, we use the reduced-state variant of Algorithm 4 to compute a reachable set for the traffic densities x_5 and x_6 at the end of the road, using an order $k = 10$ empirical inverse Christoffel function with accuracy and confidence parameters $\epsilon = 0.05$, $\delta = 10^{-9}$. Computation times for this example were 10 minutes on the laptop and 2 minutes on `c5.24xlarge`.

Figure 4.4 compares the reachable set estimate computed with Algorithm 4 to the projection of the tight interval over-approximation computed using the monotonicity property of the traffic system. The figure indicates that the tight interval over-approximation of the reachable set is a somewhat conservative over-approximation, since the reachable set has approximately the shape of a parallelotope whose sides are not axis-aligned.

4.5 Conclusion

Algorithm 4 demonstrates that Christoffel functions, in addition to being useful in data analysis, provide a powerful technique for data-driven reachability. Algorithm 4's probabilistic safety guarantee, presented in Theorem 2, demonstrates that statistical learning theory is a powerful technique for data-driven control theory in general, in this case presenting a significant generalization over the scenario approach to data-driven reachability. The only essential limit to the PAC approach is the requirement that $\mathcal{C}$ have a finite VC dimension. While this condition is sufficient for many classes of estimator geometry—certainly a greater number than are permitted by the scenario approach– but a surprising number of simple geometries have infinite VC dimension. Convex hulls of point clouds are a simple and particularly striking example of this phenomenon. More practically, we find that the sublevel sets of a kernelized variation of Christoffel functions exhibits this property. Although we have defined the Christoffel function using the standard monomial basis vector $z_k(x)$, the Christoffel function is in fact invariant to changes in polynomial coordinates. For instance,

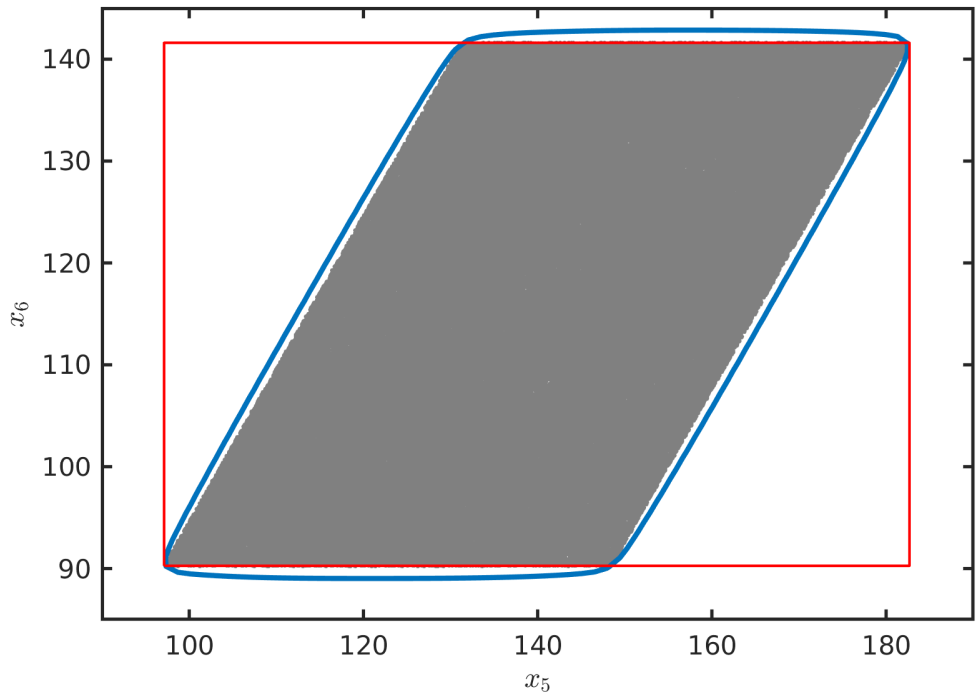

Figure 4.4: Reachable set estimate for the monotone traffic model with an order 10 empirical inverse Christoffel function (blue), compared to the tight interval over-approximation (red). The reachable set estimate was computed with Algorithm 4 using samples projected onto states x_5 and x_6.

$z_k(x)$ could be replaced with the feature vector $\phi_k(x)$ of the polynomial kernel $(1 + x^\top x)^k$, that is the monomial vector $\phi_k(x)$ such that $\phi(x)^\top \phi(x) = (1 + x^\top x)^k$. By an application of the kernel trick, this approach can be extended to kernels with infinite-dimensional feature spaces, as in [7]. However, the proof of Theorem 2 does not extend to the infinite-dimensional case, so we find ourselves once again on the search for more generality.

Chapter 5

The Bayesian PAC Approach

This chapter investigates how to apply Bayesian PAC analysis techniques to prove probabilistic safety guarantees for data-driven reachability algorithms meet neither the heuristic and parametric requirements of the scenario approach, nor the VC dimension condition required by the PAC approach of the PAC approach. Bayesian PAC obviates the need for VC dimension bounds by applying additional probabilistic structure to the class of reachable set estimators. The technique works by constructing prior and posterior measures P and Q over $\mathcal{C}$, where Q is allowed to depend on observations. The relative entropy between P and Q, rather than VC dimension, provides the geometric restriction required for the probabilistic safety guarantee to hold.

As in Chapter 4, we will focus on a single example in this chapter, namely a data-driven reachability algorithm that uses the kernel extension of empirical Christoffel functions alluded to in Section 4.5. The Bayesian PAC approach actually provides two benefits to the theory of Christoffel-based data-driven reachability. First, it allows for the construction of finite-sample guarantees for kernel inverse Christoffel functions, which have an infinite number of parameters: this would have been an impossible feat for the techniques of Chapters 3 and 4. Second, Bayesian PAC can improve matters in the finite-parameter case: when applied to polynomial empirical inverse Christoffel function estimators, the examples in Section 5.4 demonstrate that Bayesian PAC analysis can provide guarantees of probabilistic accuracy and confidence with much greater sample efficiency than the finite-sample bounds provided by classical VC dimension bound arguments.

Section 5.1 reviews how kernel Christoffel functions are derived from the polynomial Christoffel functions described in Section 4.1. Section 5.2 presents the kernel extension of Algorithm 4; unlike its predecessor, Algorithm 5 is sequential in nature, as Bayesian PAC does not provide an *a priori* sample bound. Section 5.3 reviews the theory of Bayesian PAC analysis and constructs the probabilistic safety guarantee for Algorithm 5 and its descendants. Finally, Section 5.4 tests the algorithms of this chapter on the same numerical examples as the last chapter, and compares their relative performance.

5.1 Kernel Christoffel Functions

Recall from Section 4.1 that the empirical inverse Christoffel function for data $r_1, \ldots, r_N$ is

$$\hat{\kappa}^{-1}(x) = z_m(x)^\top \hat{M}_{m,\sigma}^{-1} z_m(x), \tag{5.1}$$

where

$$\hat{M}_{m,\sigma} = \sigma^2 I + \frac{1}{N}\sum_{i=1}^{N} z_m(x_i) z_m(x_i)^\top. \tag{5.2}$$

The dyadic sum $\frac{1}{N}\sum_{i=1}^{N} z_m(x_i)z_m(x_i)^\top$ can be expressed as the matrix product $\frac{1}{N}ZZ^\top$, where $Z \in \mathbb{R}^{\binom{n+m}{n}\times N}$ is the matrix $Z = [z_m(x_i) \;\; \ldots \;\; z_m(x_N)]$ of polynomial features. By expressing the dyadic sum this way, we can apply the matrix inversion lemma to express the inverse of the empirical moment matrix as

$$\begin{aligned}\hat{M}_{m,\sigma} &= \left(\sigma^2 I + \tfrac{1}{N} ZZ^\top\right)^{-1} \\ &= \sigma^{-2}\left(I - Z\left(\sigma^2 N I + Z^\top Z\right)^{-1} Z^\top\right).\end{aligned} \tag{5.3}$$

This expression for $\hat{M}_{m\sigma}$ allows us to rewrite the empirical inverse Christoffel function as

$$\begin{aligned}\hat{\kappa}^{-1}(x) = {} & N\sigma_0^{-2} z_m(x)^\top z_m(x) \\ & - N\sigma_0^{-2} z_m(x)^\top Z\left(\sigma_0^2 I + Z^\top Z\right)^{-1} Z^\top z_m(x),\end{aligned} \tag{5.4}$$

where we have made the change of variables $\sigma^2 = \sigma_0^2/N$. The vector z_m enters (5.4) only through the inner products $z_m(x_i)^\top z_m(x_j)$: The matrix $Z^\top Z \in \mathbb{R}^{N\times N}$ has elements $(Z^\top Z)_{ij} = z_m(x_i)^\top z_m(x_j)$, and the matrix-vector product $Z^\top z_m(x)$ has elements

$$(Z^\top z_m(x))_i = z_m(x_i)^\top z_m(x).$$

By replacing the inner product $z_m(x_i)^\top z_m(x_j)$ with an arbitrary positive definite[1] function $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ and rescaling by a factor of σ_0^2/N, we obtain the kernelized variant of the empirical inverse Christoffel function,

$$\kappa^{-1}(x) = k(x,x) - k_D(x)^\top\left(\sigma_0^2 I + K\right)^{-1} k_D(x), \tag{5.5}$$

where $K \in \mathbb{R}^{N\times N}$ and $k_D(x) \in \mathbb{R}^N$ are defined as

$$K_{ij} = k(x_i, x_j), \quad (k_D(x))_i = k(x_i, x). \tag{5.6}$$

5.2 Kernel Christoffel Function as an Estimator of Support

[1]Here, and throughout the paper, we mean positive definite in the sense of reproducing kernel Hilbert spaces and kernel machines, which is that a square matrix K with elements $(K)_{ij} = k(x_i, x_j)$ is a positive definite matrix.

Algorithm 5: To estimate a support set by a kernelized empirical inverse Christoffel function satisfying a Bayesian PAC bound.

inputs: random variable X with support in $\mathcal{X}$; positive definite kernel function k; PAC parameters $\epsilon, \delta \in (0,1)$; noise parameter $\sigma_0^2 \in \mathbb{R}_{++}$; initial sample size N_0; batch size N_b; threshold η.
$N \leftarrow N_0$
$D \leftarrow (x_1, \ldots, x_N) \overset{\text{i.i.d.}}{\sim} X$
$i \leftarrow 0$
$\epsilon^0 \leftarrow 1$
while $\epsilon^i > \epsilon$ **do**
  $i \leftarrow i + 1$
  append
  $(r_{N+1}, \ldots, r_{N+N_b}) \overset{\text{i.i.d.}}{\sim} R$ to D
  $N \leftarrow N + N_b$
  $K_{\sigma_0} \leftarrow \sigma_0^2 I + K$
  define $C : \mathcal{X} \to \mathbb{R}_+$ to be $C(x) = k(x,x) - k_D(x) K_{\sigma_0}^{-1} k_D(x)$;
  Evaluate $\overline{r}$ as in (5.15)
  $\epsilon_i \leftarrow \frac{\overline{r} + \frac{2}{N} \log(\frac{\pi^2 i^2}{6\delta})}{1 - F_1(1)}$, F_1 as in (5.14)
end while
return $\mathbb{1}\{C(x) \leq \eta\}$

Algorithm 5 is a procedure to estimate the support of a random variable with a sublevel set of an kernel empirical inverse Christoffel function, where the only information needed from the random variable is a collection of iid samples. The algorithm is designed to satisfy a Bayesian PAC bound which is developed in Section 5.3. Unlike the classical PAC bound provided for Algorithm 4, this Bayesian PAC bound is applicable to all kernelized empirical inverse Christoffel functions, including those whose sublevel sets have infinite VC dimension. When applied to polynomial empirical inverse Christoffel functions as a special case, we find that it is more sample-efficient than the classical PAC bound: in some of the examples in Section 5.4, the Bayesian PAC bound requires an order of magnitude fewer samples to achieve the same accuracy and confidence as that guaranteed by the classical PAC bound. The disadvantages of the Bayesian PAC approach is that the required number of samples is not known *a priori*, since certain terms in the bound depend on the data. Algorithm 5 therefore takes an iterative approach, taking samples in batches and re-evaluating the Bayesian PAC bound after each batch until it reaches the desired level of accuracy.

Remark 2. *In some reachability problems, we are only interested in computing a reachable set for a subset of the state variables. For example, suppose the state is $(x_1, \ldots, x_n) \in \mathbb{R}^n$, and we wish to verify a safety specification involving only the states $x_1, \ldots, x_s$, where $s < n$: a reachable set for the states $x_1, \ldots, x_s$ would suffice for this problem. In cases like this, the*

algorithms presented in this section can be modified to use only the first s elements of the samples. The output of the algorithm is then an empirical inverse Christoffel function with domain $\mathbb{R}^s$ whose sublevel set $\hat{\mathcal{R}}$ estimates the reachable set for the reduced set of states. In the sequel, we refer to this variation of the algorithms in this section as their reduced-state *variations.*

5.3 Bayesian PAC Analysis

Much like the classical PAC analysis explored in Chapter 4, Bayesian PAC is a technique to bound the deviation between true risk and empirical risk for the problem of selecting a concept $c \in \mathcal{C}$ that minimizes a risk function defined by a random variable X. What sets Bayesian PAC analysis apart from its classical counterpart is that it bounds the deviation of the expected values of the true and empirical risks with respect to a data-dependent probability measure defined over $\mathcal{C}$, instead of bounding it uniformly. Given a prior measure P over $\mathcal{C}$ and a posterior measure Q derived from the prior and the observations, we define the expected risk $r_Q = \mathbb{E}\left[\ell(c, X)\right]$ and empirical expected risk $\hat{r}_Q = \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} \ell(c, x_i)\right]$ where $c \sim Q$. Equivalently, P and Q define random variables C_P, C_Q supported on $\mathcal{C}$, called the prior and posterior *stochastic estimators*: r_Q and $\hat{r}_Q$ are the true and empirical risks of C_Q. A Bayesian PAC bound is a bound on the deviation between r_Q and $\hat{r}_Q$. Bayesian PAC bounds can be used to provide an error bound for a single classifier which captures the central behavior of Q, which we call the *central concept* and denote as $\bar{c}_Q$. To verify that Algorithms 5 provides a valid solution to Problem 1, we show that its output is the central concept of a posterior stochastic estimator and use a Bayesian PAC bound (with $X = R$, of course) to show that a bound of the form (2.1) holds.

The most common tool to construct Bayesian PAC bounds is the PAC-Bayes theorem developed by McAllester [63], Seeger [77] and others [59]. We use the variation due to Seeger. This theorem assumes that the concept class admits a parameterization which can be infinite-dimensional.

Theorem 3 (PAC-Bayes Theorem, adapted from [77, 59])**.** *Consider a concept class $\mathcal{C}$ admitting a parametrization by $w \in \mathcal{W}$. Let the loss function be zero-one valued, that is $\ell : \mathcal{C} \times \mathcal{X} \to \{0, 1\}$. The following bound holds for all measures P, Q over the concept class $\mathcal{C}$ defined by measures W_P and W_Q over $\mathcal{W}$ such that W_Q is absolutely continuous with respect to W_P:*

$$P_X^N\left(\{x_1, \ldots, x_N : D_{ber}(\hat{r}_Q || r_Q) \leq \gamma\}\right) \geq 1 - \delta, \tag{5.7}$$

where $\gamma = (D_{KL}(W_Q||W_P) + \log\frac{N+1}{\delta})/N$. Here, $D_{KL}(W_P||W_Q)$ denotes the Kullback-Leibler (KL) divergence between W_P and W_Q, and $D_{ber}(q||p)$ denotes the KL divergence between two Bernoulli distributions with parameters q and p, given by the formula

$$D_{ber}(q||p) = q\log\frac{q}{p} + (1-q)\log\frac{1-q}{1-p}. \tag{5.8}$$

For a given set of data $x_1, \ldots, x_N$, confidence parameter δ, and a prior measure P chosen independently of the data, the inequality (5.7) provides a family of Bayesian PAC bounds, one for each posterior measure Q.

We use the PAC-Bayes theorem in the proof of Theorem 4, which asserts the validity of Algorithm 5. First, we construct prior and posterior stochastic estimators C_P and C_Q, corresponding to measures P, Q over a concept class, which admit a sublevel set of the empirical inverse Christoffel function as a central concept; namely $\bar{c}_Q = \{x : \kappa^{-1}(x) \leq \eta\}$ for a given positive η. Next, we express a formula to compute the empirical stochastic risk $\hat{r}_Q$ of C_Q from the data. Then, we establish a bound on the true stochastic r_Q in terms of $\hat{r}_Q$ using the PAC-Bayes theorem. Finally, we prove a bound on the true risk $r(\bar{c}_Q)$ of the central concept in terms of r_Q. This sequence of bounds combines to yield a bound of the form (2.1) computable in terms of known data.

Theorem 4. *Denote C^i as the inverse Christoffel function constructed during the ith iteration of Algorithm 3.2. We have the following PAC bound on all the inverse Christoffel functions constructed during the algorithm:*

$$\mathbb{P}(\forall i \geq 1,\, P_R(\{x : C^i(x) \leq \eta\}) \geq 1 - \epsilon^i) \geq 1 - \delta. \tag{5.9}$$

Thus, with confidence δ, upon the termination condition of Algorithm 3.2, we are left with a support set estimate of probability mass $\geq 1 - \epsilon$.

In addition to verifying the validity of the terminal output of Algorithm 5, Theorem 4 justifies the use of Algorithm 5 in an "any time algorithm" fashion, that is as an algorithm whose output is verified even if execution is stopped prematurely. The execution of Algorithm 5 will terminate as long as the growth of $D_{KL}(\mathcal{N}\left(0, (\sigma_0^{-1}I + K^{-1})^{-1}\right) || \mathcal{N}(0, K))$ is $o(N)$: determining the conditions under which this growth condition holds is a topic for future research.

We now develop the constructions used in the proof, starting with the prior and posterior stochastic estimators for the kernel case. We take

$$C_P = \{x : g_p(x)^2 \leq \eta\}, \quad C_Q = \{x : g_q(x)^2 \leq \eta\}, \tag{5.10}$$

where g_p and g_q are the prior and posterior of a general Gaussian process regression model with prior kernel k, conditioned on the observations $x_1, \ldots, x_N$, $y_1 = \ldots = y_N = 0$ with observation noise level σ_0^2. [2] The corresponding concept class is the class of η-sublevel sets of functions in the support of g_p, which depends on the choice of kernel. According to (5.25), g_q has posterior mean $m_q = 0$ and variance

$$\mathrm{Var}_{g_q}(x) = k(x,x) - k(r,x)^\top \left(\sigma^2 I_N + K\right)^{-1} k(r,x), \tag{5.11}$$

[2] Appendix 5.6 provides background on the theory of Gaussian process regression models.

where K denotes the kernel Gramian matrix of $x_1, \ldots, x_N$. We take the posterior central concept to be $\bar{c}_Q = \{x : \mathbb{E}\left[g_q(x)^2\right] \leq \eta\}$. Since $\mathbb{E}\left[g_q(x)\right] = m_q(x) = 0$ for all $x \in \mathcal{X}$, we know $\mathbb{E}\left[g_q(x)^2\right] = \mathrm{Var}_{g_q}(x)$. This means that the posterior central concept is

$$\begin{aligned} \bar{c}_Q &= \{x : k(x,x) \\ &\quad - k(r,x)^\top \left(\sigma^2 I_N + K\right)^{-1} k(r,x) \leq \eta\} \\ &= \{x : \kappa^{-1}(x) \leq \eta\} \end{aligned} \tag{5.12}$$

as desired.

Next, we construct the sequence of bounds, starting with the formula for the empirical stochastic risk of C_Q in terms of known data.

Lemma 3. *For the zero-one membership loss $\ell(c,x) = \mathbb{1}\{x \notin c\}$, the empirical stochastic risk of the posterior stochastic estimators C_Q defined in* (5.10) *is*

$$\hat{r}_Q = \frac{1}{N} \sum_{i=1}^{N} 1 - F_1\left(\frac{\eta}{\kappa^{-1}(x_i)}\right), \tag{5.13}$$

where F_1 is the CDF of the chi-square distribution with one degree of freedom, that is

$$F_1(x) = \mathbb{P}\left(Z^2 \leq x\right) \text{ where } Z \sim \mathcal{N}(0,1). \tag{5.14}$$

The proofs of this Lemma and the other Lemmas in this section are deferred to Appendix 5.7.

Next, we use the PAC-Bayes theorem to bound the stochastic risk r_Q by the empirical stochastic risk $\hat{r}_Q$.

Lemma 4. *Let $x_1, \ldots, x_N \overset{i.i.d.}{\sim} X$ denote a set of observations used to construct C_Q from C_P in* (5.10)*. The stochastic risk r_Q is bounded by $\overline{r} \in (0,1)$, where*

$$\overline{r} = \sup\{\beta : D_{ber}(\hat{r}_Q || \beta) \leq \gamma_k\} \tag{5.15}$$

with confidence $1-\delta$, where

$$\gamma_k = \frac{(D_{KL}(\mathcal{N}\left(0, (K^{-1} + \sigma_0^{-2} I)^{-1}\right) || \mathcal{N}(0,K)) + \log \frac{N+1}{\delta}}{N}.$$

Since $D_{ber}(q||p)$ is convex in (q,p) and equal to zero for $q = p$, the set in (5.15) is an interval containing $\hat{r}_Q$. Once $\hat{r}_Q$ and the right-hand side of the inequality in (5.15) are evaluated, the supremum $\overline{r}$ can be computed using a scalar root-finding procedure to solve $D_{\mathrm{ber}}(\hat{r}_Q||\beta) - (D_{KL}(\mathcal{N}\left(0, (K^{-1} + \sigma_0^{-2} I)^{-1}\right) || \mathcal{N}(0,K)) + \log \frac{N+1}{\delta})/N = 0$ over the interval $\beta \in [\hat{r}_Q, 1)$.

Finally, we relate the statistical risk of $r(\bar{c}_Q)$ to r_Q.

Algorithm 6: To estimate a support set by a polynomial empirical inverse Christoffel function satisfying a Bayesian PAC bound.

inputs: random variable R with support in $\mathcal{X}$; Christoffel function order m; PAC parameters $\epsilon, \delta \in (0,1)$; noise parameter $\sigma_0^2 \in \mathbb{R}_{++}$; initial sample size N_0; batch size N_b.
$N \leftarrow N_0$
$D \leftarrow (x_1, \ldots, x_N) \overset{\text{i.i.d.}}{\sim} R$
$i \leftarrow 0$
$\epsilon^0 \leftarrow 1$
while $\epsilon^i > \epsilon$ **do**
  $i \leftarrow i + 1$
  append $(x_{N+1}, \ldots, x_{N+N_b}) \overset{\text{i.i.d.}}{\sim} R$ to D
  $N \leftarrow N + N_b$
  define $C : \mathcal{X} \to \mathbb{R}_+$ to be $C(x) = z_m(x)^\top \hat{M}_{m,\sigma_0}^{-1} z_m(x)$;
  evaluate $\bar{r}$ as in (5.18)
  $\epsilon_i \leftarrow \frac{\bar{r} + \frac{2}{N}\log(\frac{\pi^2 i^2}{6\delta})}{1 - F_1(1)}$, F_1 as in (5.14)
end while
return $\mathbb{1}\{C(x) \le \eta\}$

Lemma 5. *The statistical risk $r(\bar{c}_\eta)$ of the posterior central concept and the stochastic risk r_Q of the posterior stochastic estimator satisfy the bound $r(\bar{c}_Q) \le \frac{1}{1-F_1(1)} r_Q \approx 3.15 r_Q$.*

When combined, the sequence of bounds, the sequence of bounds above provide a bound of the form (2.1) that holds independently for each iteration of Algorithm 4. Applying a union bound argument to provide a guarantee that holds uniformly over iterations forms the central argument of the proof of Theorem 4.

Proof (of Theorem 4). The bound is trivially satisfied at the beginning of execution, since $\epsilon^0 \leftarrow 1$. Next, let $i > 0$, and let C_Q^i denote the stochastic classifier $\{g_Q^i(x)^2 \le \eta\}$, where $g_Q^i(x) \sim \mathcal{N}(0, k(x,x) - k_{D^i}(x)^\top(\sigma_0^2 I + K^i)k_{D^i}(x))$, with the i superscripts signifying using the dataset accumulated so far at iteration i. Let r_Q^i denote the risk of C_Q^i. By Lemma 3.7, we have $\forall i \ge 1$, $\mathbb{P}(r_Q^i > (1 - F_1(1))\epsilon^i) \le \frac{6\delta}{\pi^2 i^2}$. By a union bound, $\mathbb{P}(\exists i,\, r_Q^i > (1 - F_1(1))\epsilon^i) \le \sum_{i \ge 1} \frac{6\delta}{\pi^2 i^2} = \delta$. Thus, with probability at least $1 - \delta$, every $r_Q^i \le \epsilon^i$. On this event, by Lemma 3.8, we have $\forall i \ge 1$, $P_X(\{x : C^i(x) > \eta\}) \le \frac{r_Q^i}{1 - F_1(1)} = \epsilon^i$ as desired. □

Bayesian PAC Analysis: the Polynomial Case

With the general kernel case settled, we now consider the polynomial case in particular. Since the kernel case reduces to the polynomial case by the kernel $k(x,y) = z_m(x)^\top z_m(y)$, we have in a sense already provided a bound for the polynomial empirical inverse Christoffel function by means of Bayesian PAC analysis. However, we can construct a prior and posterior stochastic estimator for the polynomial case which avoids direct use of the $N \times N$ kernel Gramian, which can be computationally advantageous. The special prior and posterior stochastic estimators are

$$\begin{aligned} C_P &= \{x : (W_P^\top z_m(x))^2 \leq \eta\}, \\ C_Q &= \{x : (W_Q^\top z_m(x))^2 \leq \eta\}, \end{aligned} \tag{5.16}$$

where $W_P \sim \mathcal{N}\left(0, \sigma_0^{-2} I\right)$, $W_Q \sim \mathcal{N}\left(0, \hat{M}_{m,\sigma_0}^{-1}\right)$.

Notice that $W_P^\top z_m$ and $W_Q^\top z_m$ are Gaussian processes: indeed, they correspond to the prior and posterior of a general Gaussian process regression model with prior kernel $k(x,y) = z_m(x)^\top z_m(y)$, conditioned on the observations $x_1, \ldots, x_N$, $y_1 = \ldots = y_N = 0$ with observation noise level σ_0^2. We take the central concept $\bar{c}_Q$ of C_Q to be the η-sublevel set

$$\begin{aligned} \bar{c}_Q &= \{x : \mathbb{E}\left[(W_Q^\top z_m(x))^2\right] \leq \eta\} \\ &= \{x : z_m(x)^\top \hat{M}_{m,\sigma_0}^{-1} z_m(x) \leq \eta\}, \end{aligned} \tag{5.17}$$

that is the η-sublevel set of the polynomial empirical inverse Christoffel function. Applying the PAC-Bayes theorem to this construction yields the following alternative to Lemma 4.

Lemma 6. *Let $x_1, \ldots, x_N \overset{i.i.d.}{\sim} X$ denote a set of observations used to construct C_Q from C_P in* (5.16)*. The stochastic risk r_Q is bounded by $\overline{r} \in (0,1)$, where*

$$\overline{r} = \sup\left\{\beta : D_{ber}(\hat{r}_Q || \beta) \leq \gamma_p\right\}, \tag{5.18}$$

where

$$\begin{aligned} \gamma_p = \frac{1}{N}\Bigg(& D_{KL}(\mathcal{N}\left(0, (\sigma_0^2 I + \hat{M}_{m,\sigma_0})^{-1}\right) || \mathcal{N}\left(0, \sigma_0^{-2} I\right)) \\ & + \log\frac{N+1}{\delta}\Bigg) \end{aligned}$$

Using this alternative lemma, we obtain a validation for Algorithm 6.

Corollary 1. *At each stage i of execution, the empirical inverse Christoffel function constructed in Algorithms 6 satisfies the PAC bound* (5.9)*.*

Proof. The argument to verify Algorithm 4 is identical to that used in the proof of Theorem 4, except that Lemma 6 is used instead of Lemma 4. □

Remark 3. *Algorithms 5 and 6 require that a threshold parameter η be selected* a priori *based on the kernel. For instance, if a squared exponential kernel $k(x,y) = \exp(-\|x-y\|^2/(2\ell)^2)$ is used in Algorithm 5, the resulting empirical inverse Christoffel function will always have*

values in $[0,1]$*, with values generally smaller close to data points: thus choosing a value between* 0 *and* 1 *is a suitable choice, with smaller values yielding finer approximations of the support set. For Algorithm 6, a reasonable heuristic is to select* $\eta = \binom{n+2m}{n}/\epsilon$*: one can show that the expected value of the true inverse Christoffel function of order* m *is* $\binom{n+2m}{n}$ *when the input is distributed according to* X*, so by Markov's inequality the probability mass of the* $\binom{n+2m}{n}/\epsilon$*-sublevel set of the true inverse Christoffel function is at least* $1-\epsilon$*.*

Numerical Considerations for Large Datasets

As the sample size N grows, the calculations in Algorithm 5 involving the kernel matrix K can become computation- and memory-intensive. In particular, evaluating $\kappa^{-1}(x)$ to compute the support set estimate and computing the KL divergence that appears in (5.15) both require the construction of an $N \times N$ matrix and an $O(N^3)$ matrix inversion. Computational difficulties related to the size of the K matrix are well known in the field of kernel machines; in response, a wealth of approximation techniques have been developed to reduce compute and memory requirements at the cost of fidelity. These approximation techniques can be used to improve the efficiency of evaluating the kernelized empirical inverse Christoffel function and its construction via Algorithm 5.

For example, to reduce the speed and memory requirements of evaluating $\kappa^{-1}(x)$, we can replace the kernel matrix K with its rank-r Nyström approximation [91]. The Nyström approximation is a method to construct low-rank approximations of Gramian matrices, such as the kernel matrix K, which has a simple expression in terms of block submatrices of the original matrix. Specifically, the rank-r Nyström approximation of the kernel matrix K has the form

$$\tilde{K} = K_{Nr} K_{rr}^{-1} K_{Nr}, \tag{5.19}$$

where $K_{Nr} \in \mathbb{R}^{N \times r}$, $K_{rr} \in \mathbb{R}^{r \times r}$ are submatrices of K whose i,j elements are $k(x_i, x_j)$. Making the substitution $K \mapsto \tilde{K}$ and applying the matrix inversion lemma to $\kappa^{-1}(x)$ yields

$$\begin{aligned} \tilde{\kappa}^{-1}(x) &= k(x,x) \\ &\quad - k_D(x)^\top (\sigma_0^{-2} I + K_{Nr} K_{rr}^{-1} K_{Nr})^{-1} k_D(x) \\ &= k(x,x) \\ &\quad - \sigma_0^{-2} k_D(x)^\top k_D(x) - k_D(x,x)^\top V k_X(x), \end{aligned} \tag{5.20}$$

where

$$V = K_{Nr}(\sigma_0^2 K_{rr} + K_{rN} K_{Nr})^{-1} K_{rN}$$

To numerically compute the final expression, we need only invert an $r \times r$ matrix instead of an $N \times N$ one; indeed, we do not need to explicitly construct an $N \times N$ matrix at all.

Next, we consider a method to over-approximate the KL divergence based on the r largest eigenvalues of K. Since the KL divergence $D_{KL}(Z_0 || Z_1)$ between N-dimensional

normal random variables $Z_0 \sim \mathcal{N}(\mu_0, \Sigma_0)$ and $Z_1 \sim \mathcal{N}(\mu_1, \Sigma_1)$ has the expression

$$\begin{aligned} D_{KL}(Z_0||Z_1) = &\tfrac{1}{2}\log\det\Sigma_1\Sigma_0^{-1} \\ &+ \tfrac{1}{2}\mathrm{tr}\Sigma_1^{-1}\left((\mu_0-\mu_1)(\mu_0-\mu_1)^\top + \Sigma_0\right) \\ &- \tfrac{N}{2}. \end{aligned} \tag{5.21}$$

For $\Sigma_0 = (\sigma_0^{-2}I + K^{-1})^{-1}$, $\Sigma_1 = K$, $\mu_0 = \mu_1 = 0$, (5.21) reduces to

$$\tfrac{1}{2}\log\det(I + \sigma_0^{-2}K) + \tfrac{1}{2}\mathrm{tr}\left((I + \sigma_0^{-2}K)^{-1}\right) - \tfrac{N}{2}. \tag{5.22}$$

Since $\log(1 + \sigma_0^{-2}x)$ and $1/(1 + \sigma_0^{-2}x)$ are analytic for $x \geq 0$, we can apply the spectral mapping theorem [20, Sec. 4.7] to (5.22) to obtain an expression for the KL divergence in terms of the eigenvalues $\lambda_1, \ldots, \lambda_N$ of K, namely

$$= \frac{1}{2}\sum_{i=1}^{N}\left(\log(1 + \sigma_0^{-2}\lambda_i) + \frac{1}{1 + \sigma_0^{-2}\lambda_i} - 1\right). \tag{5.23}$$

Numerically computing the KL divergence with the expression 5.22 requires an explicit construction of the K matrix, and the inverse of an $N \times N$ matrix: this requires $O(N^3)$ operations and $O(N^2)$ memory. Using (5.23) instead of (5.22) to compute the KL divergence with the full set of eigenvalues does not generally yield an improvement, since computing the eigenvalues of K is also $O(N^3)$. However, since K is a symmetric positive definite matrix, the eigenvalues are all positive, and the m largest eigenvalues can be computed in less than $O(N^3)$ time, for instance by a Lanczos-type algorithm [86, ch. 9]. Let λ_p denote the p^{th} largest eigenvalue: Since (5.23) is a nondecreasing function in each λ_i, the approximation $\lambda_i \approx \lambda_p$ for λ_i such that $\lambda_i < \lambda_p$ yields an upper bound on the KL divergence that can be computed in less than $O(N^3)$ time.

5.4 Examples

This section demonstrates how Algorithms 5 and 6 can be used to make accurate estimates of forward reachable sets, and how they compare to Algorithm 4 of Chapter 4. These examples were run on Savio, a high-performance computing cluster managed by the University of California at Berkeley. Specifically, each experiment used a single `savio2_bigmem` node comprising 20 CPUs running at 2.3 GHz and 128 GB of memory. In all experiments, we use the parameters $\epsilon = 0.1$, $\delta = 10^{-9}$ for all three algorithms, and in Algorithm 5, we use the squared exponential kernel $k(x, y) = \exp(-\|x - y\|^2/(2\ell)^2)$. The values for m and ℓ used in experiments is listed in Table 5.1. To select thresholds in Algorithms 5 and 6, we follow the advice of Remark 3, using $\eta = 0.15$ for Algorithm 5 and $\eta = \binom{n+2m}{n}/\epsilon$ for Algorithm 6. For Algorithm 4, we use an initial sample size of $20,000$ and a batch size of $5,000$ samples. For Algorithm 6, we use an initial sample size and batch size of $1,000$ samples.

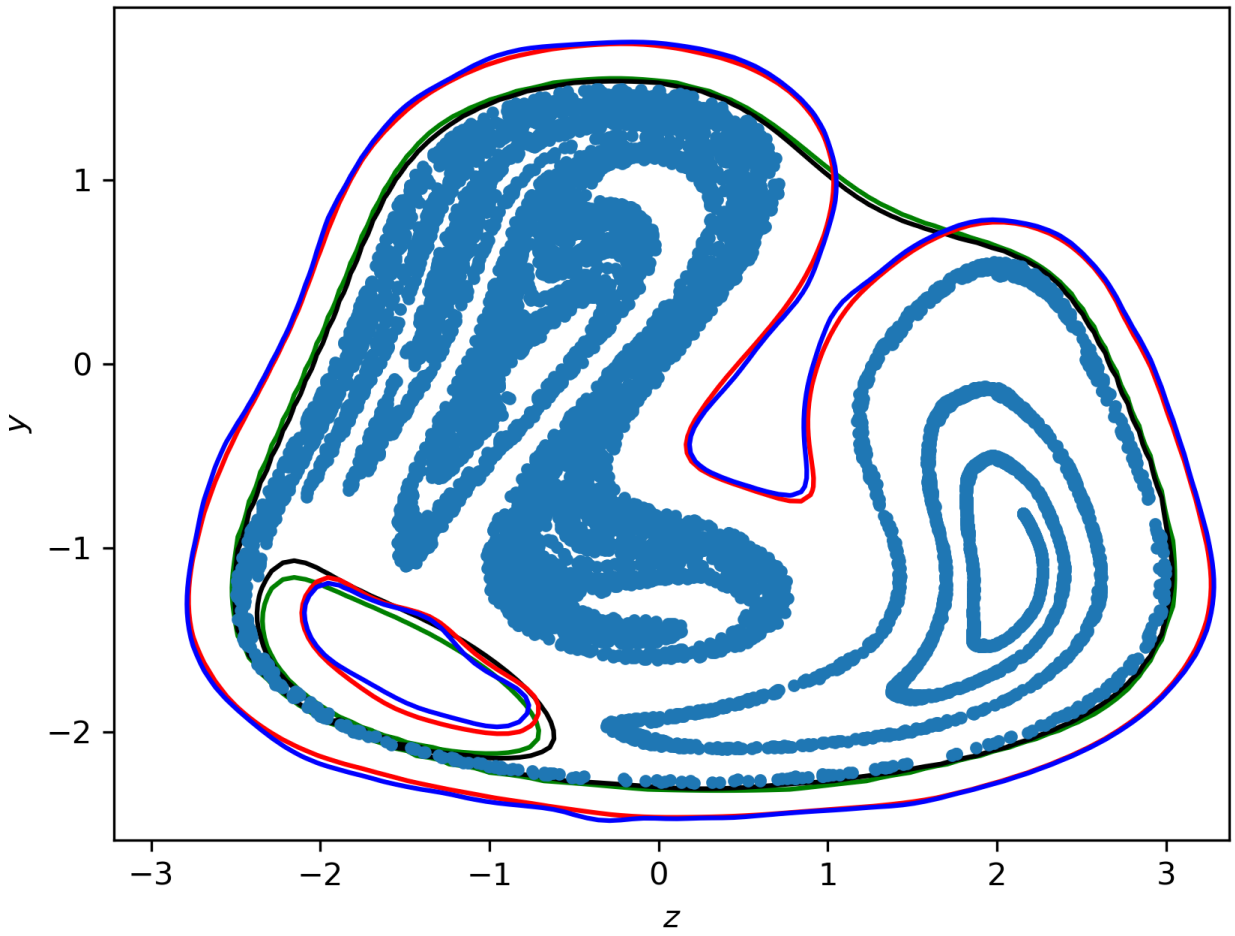

Figure 5.1: Results of Algorithms 4, 5 and 6 on the Duffing oscillator reachability problem. Black contour: output of Algorithm 4. Green contour: output of Algorithm 6. Red contour: output of Algorithm 5. Blue contour: output of Algorithm 5, over-approximated using the Nyström approximation with 1,000 samples. Blue dots: samples used in Algorithm 6.

Example	Alg. 4			Alg. 6			Alg. 5		
	m	time	N	m	time	N	ℓ	time	N
Oscillator	10	39	70307	10	13	11000	1/4	506	30000
Quadrotor	4	3	14587	4	4	6000	1/4	488	35000
Traffic	10	16	70307	10	11	10000	1/4	325	30000

Table 5.1: Computation times, sample sizes, and Christoffel function parameters for numerical experiments. All times in seconds. Algorithms 4 and 6 used polynomial order m, and Algorithm 5 used $k(x, y) = \exp(-\|x - y\|^2/(2\ell)^2)$, with m, ℓ as given in the table. All experiments use $\epsilon = 0.1$, $\delta = 10^{-9}$.

Chaotic Nonlinear Oscillator

The first example is a reachable set estimation problem for the nonlinear, time-varying system with dynamics $\dot{z} = y, \dot{y} = -\alpha y + z - z^3 + \gamma \cos(\omega t)$, with states $x = (z, y) \in \mathbb{R}^2$ and parameters $\alpha, \gamma, \omega \in \mathbb{R}$. This system is known as the *Duffing oscillator*, a nonlinear oscillator which exhibits chaotic behavior for certain values of α, γ, and ω, for instance $\alpha = 0.05$, $\gamma = 0.4$, $\omega = 1.3$. The initial set is the interval such that $z(0) \in [0.95, 1.05]$, $y(0) \in [-0.05, 0.05]$, and we take X_0 to be uniform over this interval. The time range is $[t_0, t_1] = [0, 100]$.

We use Algorithms 4 and 6 to compute reachable set estimates using an order $k = 10$ empirical inverse Christoffel function with accuracy and confidence parameters $\epsilon = 0.10$, $\delta = 10^{-9}$. Additionally, we use Algorithm 5 to compute a kernelized empirical inverse Christoffel function using the squared exponential kernel $k(x, y) = \exp(\|x - y\|^2/(2\ell^2))$ with $\ell = 0.25$. Figure 5.1 shows the reachable set estimate for the Duffing oscillator system with the problem data given above produced by all three algorithms: for Algorithm 5, both the full kernelized Christoffel function estimator and its Nyström approximation with $r = 2000$. The cloud of points are the $11,000$ samples used in Algorithm 6. The reachable set estimate is neither convex nor simply connected, closely following the boundaries of the cloud of points and excluding an empty region. In particular, all estimates exhibit a hole in a region of the state space devoid of samples.

Planar Quadrotor

The next example is a reachable set estimation problem for horizontal position and altitude in a nonlinear model of the planar dynamics of a quadrotor used as an example in [66, 17]. The dynamics for this model are $\ddot{p}_x = u_1 K \sin(\theta), \ddot{p}_h = -g + u_1 L \cos(\theta), \ddot{\theta} = -d_0\theta - d_1\dot{\theta} + n_0 u_2$, where p_x and p_h denote the quadrotor's horizontal position and altitude in meters, respectively, and θ denotes its angular displacement (so that the quadrotor is level with the ground at $\theta = 0$) in radians. The system has 6 states, which we take to be x, h, θ, and their

first derivatives. The two system inputs u_1 and u_2 (treated as disturbances for this example) represent the motor thrust and the desired angle, respectively. The parameter values used (following [17]) are $g = 9.81$, $L = 0.64$, $d_0 = 70$, $d_1 = 17$, and $n_0 = 55$. The set of initial states is the interval such that $p_x(0) \in [-1.7, 1.7]$, $\dot{p}_x(0) \in [-0.8, 0.8]$, $p_h(0) \in [0.3, 2.0]$, $\dot{p}_h(0) \in [-1.0, 1.0]$, $\theta(0) \in [-\pi/12, \pi/12]$, $\dot{\theta}(0) \in [-\pi/2, \pi/2]$, the set of inputs is the set of constant functions $u_1(t) = u_1$, $u_2(t) = u_2$ $\forall t \in [t_0, t_1]$, whose values lie in the interval $u_1 \in [-1.5 + g/L, 1.5 + g/L], u_2 \in [-\pi/4, \pi/4]$, and we take X_0 and D to be the uniform random variables defined over these intervals. The time range is $[t_0, t_1] = [0, 5]$. We take probabilistic parameters $\epsilon = 0.10$, $\delta = 10^{-9}$. Since the goal of this example is to estimate a reachable set for the horizontal position and altitude only, we are interested in a reachable set for a subset of the state variables, namely p_x and p_h. Following Remark 2, we use the reduced-state variations of Algorithms 4, 5, to compute reachable set estimates using only data for the (p_x, p_h) states, effectively reducing the dimension of the problem from 6 to 2. Figure 5.2 shows the reachable set estimate for the planar quadrotor system with the problem data given above produced by all three algorithms and the Nyström-approximated Algorithm 5 with $r = 2000$. The reachable set estimates displayed in Figure 5.2, and the computation times reported in Table 5.1, use the reduced-state variation.

Monotone Traffic

This example is a special case of a continuous-time road traffic analysis problem used as a reachability benchmark in [28]. This problem investigates the density of traffic on a single lane over a time range over four periods of duration T using the Cell Transmission Model [30] that divides the road into n equal segments. The spatially discretized model is an n-dimensional dynamical system with states $x_1, \ldots, x_n$, where x_i represents the density of traffic in the i^{th} segment. Traffic enters segment through x_1 and flows through each successive segment before leaving through segment n. The system dynamics (5.24) are monotone, i.e. order-preserving: this property allows us to compute an interval containing the reachable set by evaluating the dynamics at the extreme points of the intervals defining the initial set and the set of disturbances. While this interval over-approximation is easy to compute, and is the best possible over-approximation by an interval, it is in general a conservative over-approximation because the reachable set may only occupy a small volume of the interval. Since the empirical Inverse Christoffel function method can accurately detect the geometry of the reachable set, we use this method to compare the shape of the reachable set to the best interval over-approximation.

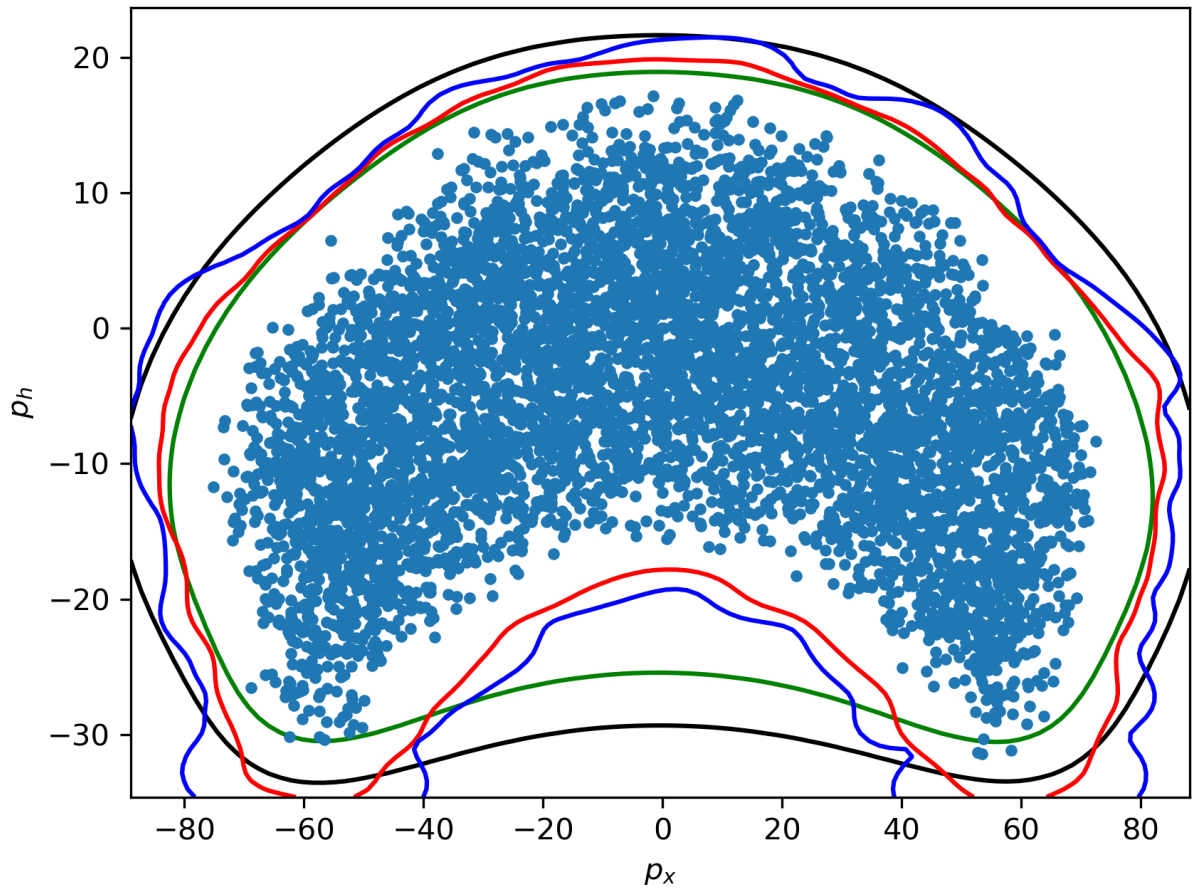

Figure 5.2: Results of Algorithms 4, 5, and 6 on the planar quadrotor reachability problem, restricting the reachability analysis to the (p_x, p_h) plane. Green contour: polynomial Christoffel function of order $k = 10$. Blue contour: kernelized inverse Christoffel function with squared exponential kernel. Red contour: Nyström approximation ($m = 10,000$) of the kernelized inverse Christoffel function with squared exponential kernel.

The state dynamics are

$$\begin{aligned}
\dot{x}_1 &= \frac{1}{T}\left(d - \min(c, vx_1, w(\overline{x} - x_2))\right) \\
\dot{x}_i &= \frac{1}{T}\big(\min(c, vx_{i-1}, w(\overline{x} - x_i)) \\
&\quad - \min(c, vx_i, w(\overline{x} - x_{i+1}))\big), \quad (i = 2, \ldots, n-1) \\
\dot{x}_n &= \frac{1}{T}\left(\min(c, vx_{n-1}, w(\overline{x} - x_n)/\beta) - \min(c, vx_n)\right),
\end{aligned} \tag{5.24}$$

where v represents the free-flow speed of traffic, c the maximum flow between neighboring segments, $\bar{x}$ the maximum occupancy of a segment, and w the congestion wave speed. The input u represents the influx of traffic into the first node. For the reachable set estimation problem, we use a model with $n = 6$ states, and take $T = 30$, $v = 0.5$, $w = 1/6$, and $\bar{x} = 320$. The initial set is the interval such that $x_i(0) \in [100, 200]$, $i = 1, \ldots, n$, the set of disturbances is the set of constant disturbances with values in the range $d \in [40/T, 60/T]$, and X_0 and D are the uniform random variables over these sets. The time range is $[t_0, t_1] = [0, 4T]$.

We use the reduced-state variant of Algorithms 4, 5, and 6 to compute a reachable set for the traffic densities x_5 and x_6 at the end of the road, using an order $k = 10$ empirical inverse Christoffel function with accuracy and confidence parameters $\epsilon = 0.10$, $\delta = 10^{-9}$. Figure 5.2 compares the reachable set estimates for the traffic system produced by all three algorithms, and the Nyström-approximated Algorithm 5 with $r = 2000$, with the projection of the tight interval over-approximation computed using the monotonicity property of the traffic system. The figure indicates that the tight interval over-approximation of the reachable set is a somewhat conservative over-approximation, since the reachable set has approximately the shape of a parallelotope whose sides are not axis-aligned.

5.5 Conclusion

The Bayesian PAC approach is the most general approach to proving safety bounds for data-driven reachability that we will cover in this thesis. This approach has no inherent restrictions on either the magnitude of the class of sets from which $\hat{\mathcal{R}}$ may be selected, nor on how $\hat{\mathcal{R}}$ is selected. However, this generality comes at a significant burden to the analyst: Bayesian PAC may yield a bound, but only if one can find a suitable prior and posterior measure over $\mathcal{C}$, compute the relative entropy and the associated stochastic risks, and can apply a procedure to pass from the posterior to a central concept. The technical developments in Section 5.3 demonstrate that this analysis can be carried out, even for very sophisticated estimators like empirical kernel Christoffel functions, thanks to the formal connection between kernel Christoffel functions and Gaussian process regression models. The examples in Section 5.4 demonstrate that our Bayesian PAC is worth the effort: not only does it provide bounds where other methods cannot, it yields an improvement in sample efficiency over classical PAC bounds.

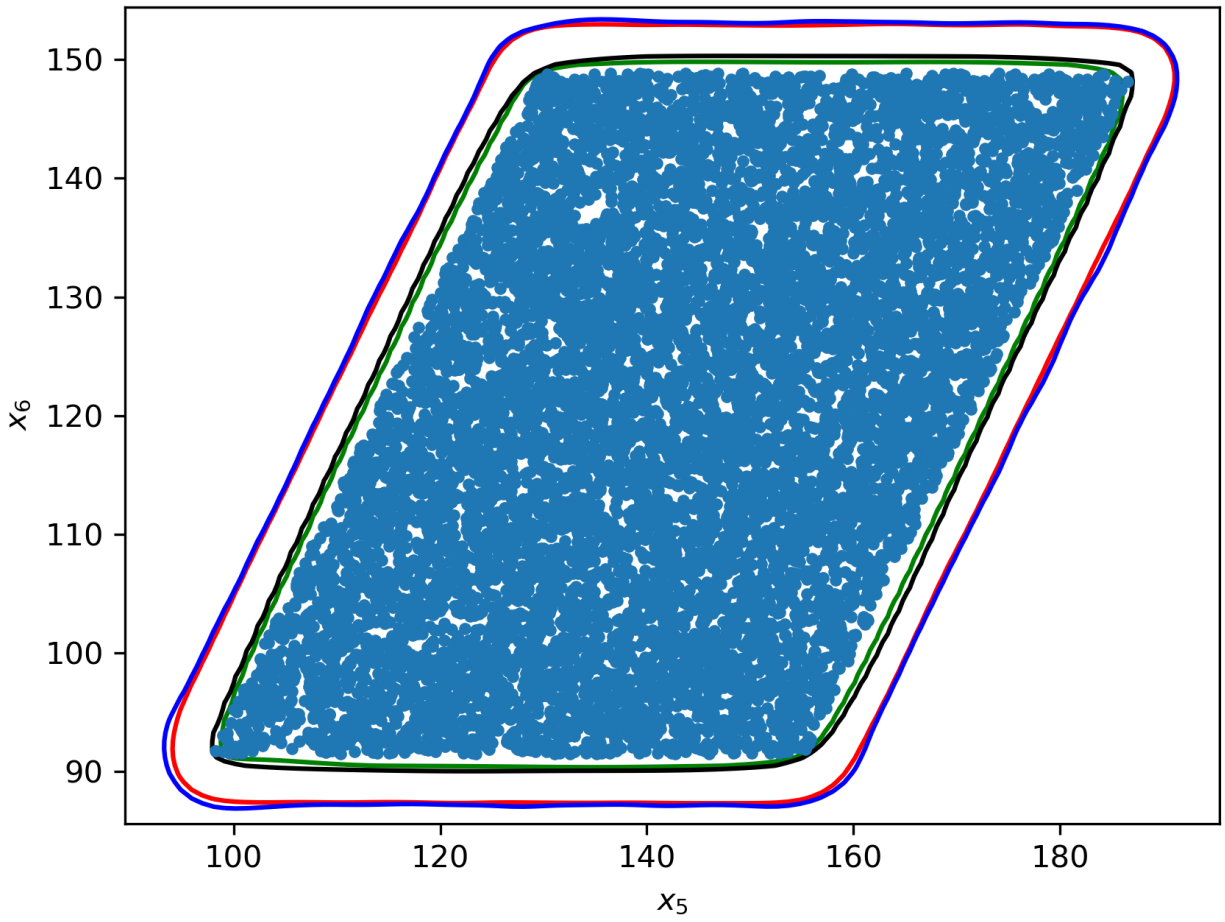

Figure 5.3: Results of Algorithms 4, 5, and 6 on the six-state monotone traffic reachability problem, restricting the reachability analysis to the (x_5, x_6) plane. Green contour: polynomial Christoffel function of order $k = 10$. Blue contour: kernelized inverse Christoffel function with squared exponential kernel. Red contour: Nyström approximation ($m = 10,000$) of the kernelized inverse Christoffel function with squared exponential kernel.

Improvements to the general theory can advance in step with advances in Bayesian PAC analysis. For instance, there are new results in theory of *derandomizing* Bayesian PAC bounds, which could offer sample efficiency improvements over the argument used in Lemma 5 to apply the Bayesian PAC bound to the central concept. Furthermore, domain-specific knowledge could be applied to the GP prior used to construct the Christoffel functions. For instance, in reachability problems and estimate of the system sensitivity matrix could be used to intelligently select length-scales in the kernel, along with other algorithm hyper-parameters such as the initial sample size and batch size.

5.6 Appendix: Background on Gaussian Process Models

A Gaussian process g is a stochastic process such that vectors $(g(x_1), \ldots, g(x_m))$ of point evaluations are multivariate Gaussian distributions. Similar to how a Gaussian random variable is completely characterized by its mean and variance, a Gaussian process is completely characterized by a mean function m, defined pointwise as $m(x) = \mathbb{E}[g(x)]$, and a positive semidefinite covariance function k, defined on all pairs of points $x, y \in \mathcal{X}$ as $k(x, y) = \mathbb{E}[g(x)g(y)]$.

Gaussian processes can also be defined according to a finite set of basis functions, admitting a direct construction as a finite weighted sum. For an m-dimensional space of functions with basis $b_1, \ldots, b_m : \mathcal{X} \to \mathbb{R}$, we form the stochastic weighted average $\sum_{i=1}^{m} w_i b_i$, where $w = (w_1, \ldots, w_m) \sim \mathcal{N}(0, \Sigma)$. This weighted average is a Gaussian process whose support is the span of $b_1, \ldots, b_m$, with mean $m(x) = 0$ and covariance $k(x, y) = \sum_{i=1}^{m} b(x)^\top \Sigma b(y)$, where $b(\cdot) = (b_1(\cdot), \ldots, b_m(\cdot))^\top$.

The Gaussian process regression model is Bayesian regression model that uses a Gaussian process as the prior over regression functions. In our case, we take the mean of the prior process to be zero. The data is assumed to be of the form $g(x_i) = h_i + \varepsilon$, where ε is a Gaussian noise term with variance σ^2. Under these conditions, the posterior for the unknown function is also a Gaussian process, whose mean and covariance are given by the formulas

$$m_q(x) = k_D(x)^\top \left(\sigma^2 I_N + K\right)^{-1} h, \tag{5.25}$$

$$k_q(x, y) = k(x, y) - k_D(x)^\top \left(\sigma^2 I_N + K\right)^{-1} k_D(y). \tag{5.26}$$

From the expression for the posterior covariance, we get the posterior variance

$$\begin{aligned} \mathrm{Var}_{g_q}(x) &= k_q(x, x) \\ &= k_p(x, x) \\ &\quad - k_D(x)^\top \left(\sigma^2 I_N + K\right)^{-1} k_D(X, x), \end{aligned} \tag{5.27}$$

which is precisely the kernelized empirical inverse Christoffel function with kernel k for the data $D = (x_1, \ldots, x_N)$ evaluated at the point x. In the finite-dimensional case, the posterior

process has mean and covariance functions

$$m_q(x) = \sigma^{-2} b(x)^\top \left(\Sigma^{-1} + \sigma^2 BB^\top\right)^{-1} By \tag{5.28}$$

$$k_q(x,y) = b(x)^\top (\Sigma^{-1} + \sigma^{-2} BB^\top)^{-1} b(x), \tag{5.29}$$

where $B \in \mathbb{R}m \times N$ is the matrix formed by evaluating the basis functions on the data, that is $B = [b(x_1) \quad \cdots \quad b(x_N)]$. Taking $b = z_k$, $\Sigma = \sigma_0^{-2} I$, $\sigma = N^{-1/2}$, yields the posterior variance $\mathrm{Var}_{g_q}(x) = z_m(x)^\top \left(\sigma_0^2 I + \frac{1}{N}\sum_{i=1}^n z_m(x_i) z_m(x_i)\top\right)^{-1} z_m(x)$, which is precisely the polynomial empirical inverse Christoffel function of order k for the data $x_1, \ldots, x_n$ evaluated at the point x.

5.7 Appendix: Proofs of Lemmas in Section 5.3

Proof of some Lemma 3

We consider the kernel case, since the polynomial case follows by the appropriate choice of kernel function. Recall that $\kappa^{-1}(x)$ is the variance of g_p by construction. Evaluating g_p at a single point x yields the normal random variable $g_p(x) \sim \mathcal{N}(0, \kappa^{-1}(x))$. It follows that $g_p(x)/\sqrt{\kappa^{-1}(x)} \sim \mathcal{N}(0,1)$, and that $g_p(x)^2/\kappa^{-1}(x) \sim \chi_1^2$, that is that $g_p(x)^2/\kappa^{-1}(x)$, is a chi-square random variable with one degree of freedom. The average loss over C_P for a fixed point x is then

$$\begin{aligned}
\mathbb{E}\left[\ell(C_Q, x)\right] &= \mathbb{E}\left[\mathbb{1}\left\{x \in C_Q\right\}\right] \\
&= 1 - \mathbb{P}\left(g_p(x)^2 \leq \eta\right) \\
&= 1 - \mathbb{P}\left(\frac{g_p(x)^2}{\kappa^{-1}(x)} \leq \frac{\eta}{\kappa^{-1}(x)}\right) \\
&= 1 - F_1\left(\frac{\eta}{\kappa^{-1}(x)}\right).
\end{aligned} \tag{5.30}$$

Averaging this expression over the data points yields (5.13).

Proof of Lemma 6

We apply the Seeger PAC-Bayes Theorem 3 to the prior and posterior measures P and Q induced by C_P and C_Q as defined in (5.16). Recall that these prior and posterior measures are defined by the random vectors $W_P \sim \mathcal{N}\left(0, \sigma_0^{-2} I\right)$, $W_Q \sim \mathcal{N}\left(0, (\sigma_0^{-2} I + \hat{M}_{m,\sigma_0})^{-1}\right)$, which act as parameters. Applying this choice of W_p and W_q to equation (5.7) of Theorem 3 yields the inequality

$$P_X^N\left(\{x_1, \ldots, x_N : D_{\mathrm{ber}}(\hat{r}_Q || r_Q) \leq \gamma\}\right) \geq 1 - \delta, \tag{5.31}$$

where

$$\gamma = \frac{D_{KL}(\mathcal{N}\left(0,(\sigma_0^2 I + \hat{M}_{m,\sigma_0})^{-1}\right) || \mathcal{N}\left(0, \sigma_0^{-2} I\right)) + \log \frac{N+1}{\delta}}{N}$$

Suppose the data set $x_1, \ldots, x_N$ is one such that the inner inequality $D_{\text{ber}}(\hat{r}_Q || r_Q) \leq \gamma$ holds: then r_Q, the true stochastic risk, lies in the set $\{\beta : D_{ber}(\hat{r}_Q||\beta) \leq \gamma\}$. The function $D_{ber}(\hat{r}_Q||\beta)$ is convex in β and covers the range $[0, \infty)$, attaining 0 for $\beta = \hat{r}_Q$ and approaching ∞ for $\beta \to 0$ and $\beta \to 1$. By these properties, $\{\beta : D_{ber}(\hat{r}_Q||\beta) \leq \gamma\}$ is a closed convex subset of $(0, 1)$ for any positive γ. As such, it attains a supremum, meaning that $\overline{r}$ as defined in (5.18) is well-defined. Thus we have, with confidence $1 - \delta$, that $\overline{r}$ is an upper bound on the stochastic risk r_Q.

Proof of Lemma 4

As in the proof of Lemma 6 we apply the Seeger PAC-Bayes Theorem 3, this time to the prior and posterior measures P and Q induced by C_P and C_Q as defined in (5.10). These measures are defined by the Gaussian processes g_p and g_q which act as the concept class parameters W_P and W_Q respectively in the statement of Theorem 3. To compute the KL divergence between W_P and W_Q, we use another result due to Seeger, described in Section 2.2 of [77], which states that the KL divergence between a prior Gaussian process g_p and the posterior Gaussian processes g_q obtained after conditioning on data $x_1, \ldots, x_N$ is equal to the KL divergence between the restriction of the two Gaussian processes to the data points, that is the KL divergence between the multivariate normal random vectors $(g_p(x_1), \ldots, g_p(x_N))$ and $(g_q(x_1), \ldots, g_q(x_N))$. The mean and covariance of these random variables are simply the restrictions of the mean and covariance functions of their defining processes to $(x_1, \ldots, x_N)$. Both random vectors have mean zero. The covariance matrix of the prior random vector $(g_p(x_1), \ldots, g_p(x_N))$ is $K_p(X, X) = K(X, X)$ as discussed in Section 5.6. By (5.25) and an application of the matrix inversion lemma, the covariance of the posterior random vector $(g_q(x_1), \ldots, g_q(x_N))$ is

$$\begin{aligned} K_q(X, X) &= K(X, X) \\ &\quad - K(X, X)\left(\sigma_0^2 I + K(X, X)\right)^{-1} K(X, X) \\ &= \left(K(X, X)^{-1} + \sigma_0^{-2} I\right)^{-1}. \end{aligned} \tag{5.32}$$

Proof of Lemma 5

Consider a point $x \in \mathcal{X}$ outside of the central concept, that is such that $\bar{c}_\eta(x) = \mathbb{E}\left[(g(x)^2\right] > \eta$. The probability that $W_Q^\top z_m(x)$ also exceeds η is bounded as

$$\mathbb{P}\left((g(x)^2 \geq \eta\right) \leq \mathbb{P}\left((g(x)^2 \geq \mathbb{E}\left[(g(x)^2\right]\right) \tag{5.33}$$

$$= \mathbb{P}\left(\frac{(g(x)^2}{\mathbb{E}\left[(g(x)^2\right]} > 1\right) \tag{5.34}$$

$$= 1 - F_1(1). \tag{5.35}$$

Next, let us consider the risk of the stochastic estimator, that is $r_Q = \mathbb{P}\left((g(X)^2 > \eta\right)$. Applying the law of total probability with respect to the random variable X, we divide r_Q into two integrals according to whether the central concept exceeds η:

$$\mathbb{P}\left((g(X)^2 > \eta\right) = \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) dP_x(x) \tag{5.36}$$

$$= \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) \mathbb{1}\left\{\mathbb{E}\left[(g(x)^2\right] > \eta\right\} dP_x(x) \tag{5.37}$$

$$+ \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) \mathbb{1}\left\{\mathbb{E}\left[(g(x)^2\right] \leq \eta\right\} dP_x(x). \tag{5.38}$$

We have that $\mathbb{P}\left((g(X)^2 > \eta\right) \geq \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) \mathbb{1}\left\{\mathbb{E}\left[(g(x)^2\right] > \eta\right\} dP_x(x)$, since all three integrands are nonnegative. To find an upper bound on this probability in terms of the empirical classifier, we combine the two inequalities above to find

$$\mathbb{P}\left((g(X)^2 > \eta\right) = \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) dP_x(x) \tag{5.39}$$

$$\geq \int_{\mathcal{X}} \mathbb{P}\left((g(x)^2 > \eta\right) \mathbb{1}\left\{\mathbb{E}\left[(g(x)^2\right] > \eta\right\} dP_x(x) \tag{5.40}$$

$$\geq (1 - F_1(1)) \int_{\mathcal{X}} \mathbb{1}\left\{\mathbb{E}\left[(g(x)^2\right] > \eta\right\} dP_x(x) \tag{5.41}$$

$$= (1 - F_1(1))\mathbb{P}\left(\mathbb{E}\left[(g(x)^2\right] > \eta\right) \tag{5.42}$$

$$= (1 - F_1(1))r(\hat{c}_\eta), \tag{5.43}$$

which we rearrange to yield $r(\bar{c}_\eta) \leq \frac{1}{1-F_1(1)} r_{Q_\eta}$.

Chapter 6

Afterword: In Data-Driven Reachability, You Can Pick Two

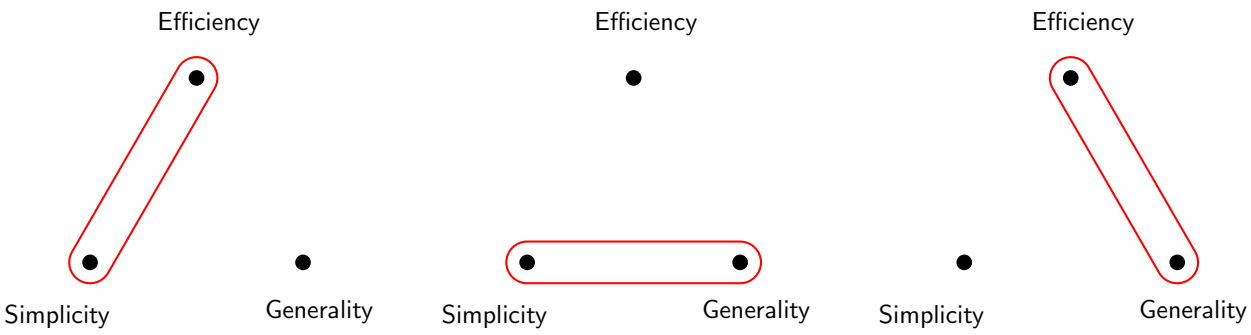

Figure 6.1: Each of the three techniques of Part I excel in two qualities but suffer in a third. To choose among the methods for a particular application, decide which quality you can sacrifice.

Chapters 3, 4, and 5 follow an upward arc of analytic generality. The method of interest in each chapter answers some obstacle insurmountable to the methods of the last, until we meet, in Chapter 5, a method with essentially no restrictions as long as one is willing to put in enough analytic elbow grease. This narrative framing might suggest that each method completely succeeds the previous, so that the Bayesian PAC method of Chapter 5 is the only admissible choice for data-driven reachability, but this is not the case. Indeed, the core method of each chapter—scenario optimization, classical PAC analysis, and Bayesian PAC analysis—is the best method to use under some circumstances. To get a sense of when to use each method, let's compare their relative utility by three qualities:

- *Ease of use*, How much calculation is required to establish a probabilistic safety guarantee;

- *Generality*, the range of estimator geometries to which the method can be applied;
- *Sample complexity*, the amount of data required by each method to furnish a probabilistic guarantee for a fixed estimator geometry.

The technique of Chapter 3, the scenario approach, is the least general; but for estimator geometries that are compatible with its requirements, it is by far the easiest path to a probabilistic guarantee. Indeed, one needs only form the convex program (3.6),[1] and the guarantee proves itself via Proposition 1.

The technique of Chapter 4, the classical PAC approach, is more general than the scenario approach, but requires somewhat more work to use. To use classical PAC, one must provide a bound on the VC dimension of the class of admissible estimators, which is generally more work than formulating a convex program. The advantage, of course, is that one is no longer restricted to estimators that can be expressed in the rather particular form of being a solution to a program like (3.6). There is a second cost to this generality: when an estimator is amenable to both techniques, the classical PAC approach tends to have greater sample complexity. In other words, it requires more data to prove the same bound as the scenario approach—by a factor of about 5 in my experiments, as the coefficients in (4.5) suggest—so when both methods apply, the tie goes to the scenario approach.

The technique of Chapter 5, as we've already covered, is the most general. It provides a probabilistic safety guarantee for any concept class for which you can find appropriate prior and posterior distributions, and whose stochastic risk and relative entropy can be practically computed. The cost, of course, is that the PAC-Bayes theorem is not constructive in this regard: the onus lies on the analyst to find the appropriate distributions and to work out the necessary calculations. A cursory review of Chapters 3 and 5 demonstrates that the degree of analytical elbow grease required by the Bayesian PAC approach single class of estimators, empirical kernel Christoffel functions, is immense compared to that required by the scenario approach for all compatible estimators. On the other hand, Bayesian PAC manages to be the most sample efficient of the three, in many cases offering an order of magnitude improvement over classical PAC.

From this qualitative review, we can see that each method possesses two of the qualities but lacks the third, as illustrated in Figure 6.1. The way to choose an estimator, then, is to decide which quality you can do without. If your desired estimator geometry admits convex parameterization with a volume proxy, then the scenario approach is by far the best choice. If your desired geometry doesn't satisfy this requirement, and the computations required to find prior and posterior distributions (and their related risks and relative entropy) appear impractical, then the classical PAC approach is a good fit. On the other hand, if maximal sample efficiency is critical, then the Bayesian PAC approach may be a better choice.

[1] Actually *solving* the program is another question, but a question irrelevant to how easy it is to prove the bound. The good news is that (3.6) is always convex, which makes it likely that an efficient solver is around the corner.

Part II

Robustness Analysis with Gaussian Process Models

Chapter 7

Background

In this part, we turn our attention to the part of data-driven control that intersects with *learning-based* control theory; in particular, the problem of how to design controllers and prove safety certificates for dynamical models that contain a learning component. Unlike in the previous part, where we assumed no system knowledge beyond measurability, we will make use of a dynamical model, factored into a fully known *nominal part* and a partially known *uncertainty*. As this thesis is concerned with probabilistic guarantees, we will represent this uncertainty using the formalism of probability; in other words, the nominal part of the model will be a fixed function, while the uncertainty will be a random function. Fixing such a model structure, our goal is twofold: to devise a method to refine the unknown part of the model using data, and to prove safety guarantees for the learned model.

As the title of this part suggests, we consider specifically the use of Gaussian processes as the stochastic model for the uncertain part of the model. The reason for this is that GPs are particularly well-suited to solving two fundamental challenges of learning-based control theory. The first challenge is the need for quantified uncertainty. The role of quantified uncertainty in our formulation of the learning-based control problem is evident from the name; it constitutes the uncertain part of the dynamical model. However, many popular learning models do not provide such a quantification. Take the case of a neural net: although it enjoys universal approximation properties under quite mild conditions, without some additional structure (e.g. Bayesian weights, dropout, committee models, &c.) it provides us no means by which to assess the accuracy of its approximation. There has been incredible progress recently in providing safety guarantees for neural dynamical models [93, 54], but these guarantees are made under the same proviso as ordinary model-driven control theory: any model inaccuracies are sufficiently small that feedback will correct for them. GPs, on the other hand, come with uncertainty quantification built-in: the standard interpretation of a GP regression model is that the posterior mean constitutes the approximate dynamics while the posterior variance quantifies the accuracy of approximation. This convenient quantification of uncertainty is what has driven GP models to experience a renaissance among control theorists in recent years, with a number of large conferences hosting workshops and special sessions exclusively for GP learning-based control.

The second challenge is a sort of "no free lunch" problem: the better a model is at learning general dynamics, the harder it is to prove safe, since there is no control-theoretic method to analyze arbitrary nonlinear dynamics without introducing a great deal of conservatism. Take stability for example: the only method with universal application is to search for a Lyapunov function, but finding a valid Lyapunov function for a general learning-based model is essentially the problem of establishing that a black-box function is everywhere negative (or at least negative in some region of the state space), a problem that is practically infeasible. The solution to this challenge, the balancing act to be performed between the two ends of the fundamental tension, is to find a learning model that provides sufficient analytical structure to feasibly establish safety guarantees while still being sufficiently general learning model to adapt to the unknowns we expect to face. Thus, while a technique to learn system dynamics can be as universal as their underlying function approximators, our ability to analyze the safety of a controller developed with one of these methods must to an extent be done on a case-by-case method. Our best hope for universality, then, is to find a learning model that can easily incorporate the structure of specific control problems. GPs are in a good position to solve this problem as well, since it's easy to add analytical structure to a GP model. The functional structure of a GP regression model is determined entirely by the structure of the prior covariance function: thus GP models can be made functionally compatible with many types of special structure simply by imposing that structure on the prior covariance. Classical examples of this phenomenon are continuity and differentiability properties of the covariance function being conferred to realizations of the process, and in the following chapters we will see examples that are more germane to the task of safety verification.

Polynomial GPs for State-space Uncertainties

Consider the case of a nonlinear state-space model, over a state space $\mathcal{X}$ and input space $\mathcal{U}$, with additive uncertainty: then our dynamical model is of the form $\dot{x} = f(x,u) + \Delta_\xi(x,u)$, where $f : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$ represents the nominal part and $\Delta : \Xi \times \mathcal{X} \times \mathcal{U}$ represents the uncertainty with the help of a probability space $(\Xi, \mathcal{X}, \mathbb{P})$. If we impose a likelihood model on some available data (taken from a trajectory roll-out or replay buffer, for instance), we can refine Δ by conditioning on the data to yield a posterior uncertainty model. Establishing that we can carry this learning procedure out without exposing the system to danger is a Bayesian, state-space instance of the problem of *safe learning* [53, 89, 26, 27, 4, 25]. The "safe" part of safe learning is effected by the usual state-space functional certificates– Lyapunov functions for stability [14, 13, 15, 74, 55, 48, 56], barrier functions for positive invariance [90, 82, 3, 46], Hamilton-Jacobi analysis for reach-avoid specifications [47, 4].

GPs have been a popular candidate for safe learning for a number of years due to their aforementioned property of uncertainty quantification; For example, [14, 13] use a Lyapunov approach to guarantee that a partially unknown system with a fixed policy is stable with high probability, and to compute a region of attraction. This approach works by verifying that the Lyapunov condition holds with high probability for the GP model over a grid of points in the state space. A theorem in [79] implies under certain conditions that this guarantee on the GP

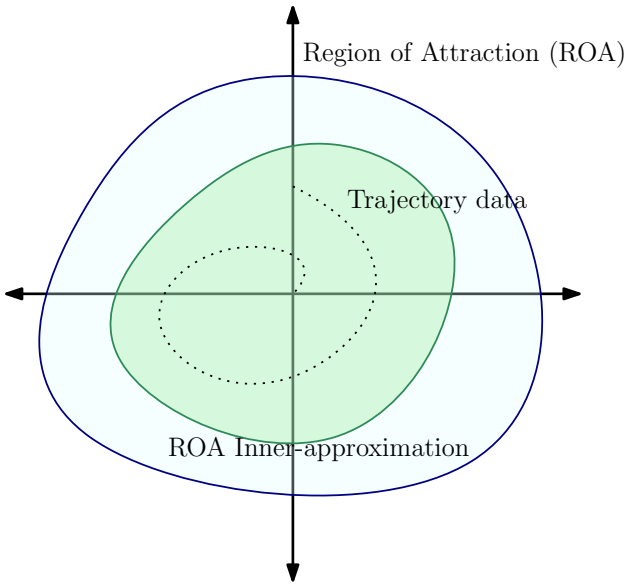

Figure 7.1: The problem of estimating an inner-approximation of a region of attraction from a GP dynamical model conditioned on trajectory data.

model holds for the true dynamics as well. While this approach is effective for verifying the stability of the learned model, it requires a base stabilizing policy and Lyapunov function. The policy and Lyapunov function stay fixed, and cannot be improved as a more accurate model is learned. One difficulty that stymied these initial works is a lack of structure in the covariance of the GP models; by allowing arbitrary covariances, only the most general techniques– in this case a quasi-Monte Carlo search over the state space– are available for analysis.

In Chapter 8, we investigate how the addition of polynomial structure can benefit safe learning techniques through the use of polynomial covariance functions that vanish at the origin, based on work begun in [37]. When applied to a GP model for additive uncertainties in a polynomial state space model, this structure allows us to synthesize stabilizing controllers and provide accuracy-only guarantees for an inner approximation of a region of attraction, as illustrated in Figure 7.1. While it places a significant restriction on the choice of covariance, polynomial structure is sufficient for learning local system dynamics, making it suitable for safely learning an inner approximation of a region of attraction. The benefit of polynomial structure is that it allows for the use of sum-of-squares analysis techniques [52] to construct a stabilizing policy that seeks to maximize the size of the region of attraction while simultaneously establishing a probabilistic guarantee that the inner approximation is valid.

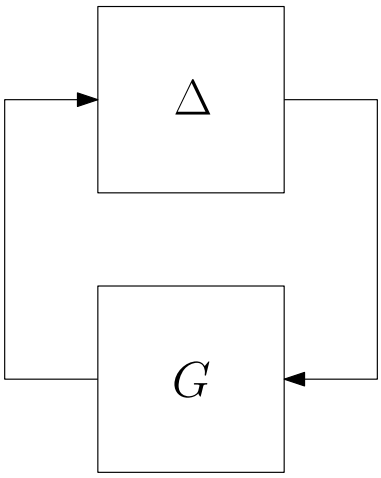

Figure 7.2: Signal flow diagram of a typical robust control system, comprising a nominal plant G in a feedback interconnection with an uncertainty Δ. Ordinarily Δ is treated as an ensemble; in Chapter 9, we investigate how to model Δ as a GP that can be refined with data.

H_∞ Gaussian Processes: A Learning Model Bespoke for Robust Control

Robust control theory is the study of the stability and sensitivity of systems subject to ensembles of uncertainty. In a standard robust control model, such as the one shown in the signal flow diagram in Figure 7, the system of interest is partitioned into three subsystems according to epistemology and control authority: a nominal LTI plant G that models what we know, and models our epistemic uncertainty with the uncertain LTI system Δ. G is taken to be fixed, known system, since it comprises our sphere of knowledge and authority. On the other hand, Δ must have a more indefinite structure in order to model the fact that our knowledge of the system is only partial.

Standard robust control theory uses an ensemble model for uncertainty: Δ is assumed to lie in a set that captures all systems consistent with our limited knowledge. A simple, and common, example of such an ensemble is the set of all Δ such that $\|\Delta\| < u$ for some $u > 0$, where $\|\cdot\|$ denotes an operator norm. The literature of robust control demonstrates that the ensemble structure is well suited for modeling and analysis of uncertainty in the context of control. Unfortunately, ensemble models are not amenable to refinement from data beyond schemes that discard uncertainties that are inconsistent with data. Essentially, there is too little structure for inference. Chapter 9 uses the probabilistic structure of a GP uncertainty model to allow for both learning and probabilistic robustness analysis. The central idea is to use a GP as a model for Δ instead of an ensemble: the uncertainty can then be refined using GP regression, and the robustness of the overall system can still be

ascertained probabilistically.

The use of GP models to estimate LTI systems (both in the time domain [70, 24] and frequency domain [62, 80]) and robust control of LTI systems with probabilistic uncertainty [57, 10, 16, 78, 19] are popular, but generally independent, streams of research. A union of these independent streams would be a significant advance for learning-based control– a transfer function learned with Bayesian techniques could be made provably safe using probabilistic robustness techniques. Chapter 9 provides conditions under which a GP transfer function model can effect such a union, expanding on the work begun in [36] Specifically, we provide conditions under which realizations of a complex Gaussian process of a complex variable correspond to the z-transform of an LTI, causal, BIBO stable, and real system with probability one. Since an LTI, causal, and BIBO stable system is characterized by a z-transform that resides in the Hardy space H_∞, we refer to such processes as H_∞ GPs. The utility of H_∞ GPs is that they simultaneously support learning through a complex extension of standard GP regression, and robustness through probabilistic accuracy-only guarantees that the behaviors of the model are covered by an integral quadratic constraint.

symbol	definition
GP	Gaussian process
RKHS	Reproducing kernel Hilbert space
SOS	sum of squares
IQC	integral quadratic constraint
k	covariance function for real GPs; Hermitian covariance for complex GPs
$\tilde{k}$	complementary covariance function for complex GPs;
m_q, k_q	posterior mean and covariance functions
K	kernel Gramian matrix
$\mathcal{N}(\mu, \Sigma)$	Multivariate normal with mean μ and covariance Σ
$\mathcal{CN}\left(\mu, \Sigma, \tilde{\Sigma}\right)$	Complex normal with mean μ, Hermitian covariance Σ, and complementary covariance $\tilde{\Sigma}$
$\mathbb{R}[\zeta]$	Polynomials in ζ with real coefficients
$\mathbb{R}[\zeta]^m$	Polynomial vectors in ζ with real coefficients
$\mathbb{R}[\zeta]^{m\times n}$	Polynomial matrices in ζ with real coefficients
$\Sigma[\zeta]$	sum-of-squares polynomials in ζ
f, g	Nominal system components
Δ	System uncertainty
$\mathcal{D}$	data set used in GP regression
σ_n^2	GP regression noise variance
V, γ	Lyapunov function candidate and sublevel parameter
$\mathcal{R}$	Region of attraction
κ	stabilizing control policy
s_V, s_{d_i}, s_γ	S-procedure certificates
β_N	probabilistic upper bound on Var
η	upper bound on β_N

H_∞	Space of functions $\mathbb{C} \to \mathbb{C}$ bounded on $\lvert z\rvert = 1$, analytic on $\lvert z\rvert > 1$
H_2	Space of functions $\mathbb{C} \to \mathbb{C}$ with integrable modulus on $\lvert z\rvert = 1$, analytic on $\lvert z\rvert > 1$
$\underline{k}$, $\underline{K}$	Augmented covariance function; augmented kernel Gramian
ℓ^1	space of square-summable sequences
N_u	number of u-level gain upcrossings
$P_u(f)$	u-level gain bound violation probability of an H_∞ GP f

Table 7.1: Important symbols and acronyms used in Part II.

Chapter 8

Polynomial Gaussian Processes for State-Space Uncertainties

We begin our investigation into robustness analysis of GP uncertainty models by exploring the case of a state-space model with an additive uncertainty, as this modeling approach is by far the most common in the GP control literature. Specifically, this chapter continues the investigation into safe learning problem introduced in Chapter 7; namely to estimate a region of attraction for a dynamical model of the form

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + \Delta(x(t)) \tag{8.1}$$

when f and g are fixed functions and Δ is a GP. Our specific approach, which sets this work apart from the literature reviewed in Chapter 7, is the introduction of polynomial structure: this structure allows us to perform certain computations that would otherwise be practically infeasible, such as simultaneously searching for a stabilizing policy and ROA inner-approximation with a probabilistic guarantee using SOS programming [52].

Our analysis has three goals. The first is to use data collected from system trajectories to model the unknown part of the dynamics. Section 8.1 lays out the specific model and structural assumptions required for polynomial GP analysis. Section 8.2 reviews GP models for state-space dynamics, introduces a class of polynomial covariance functions suitable for estimating unknown dynamics about a known equilibrium, and establishes that the assumptions laid out in Section 8.1 are sufficient to provide an accuracy-confidence bound on the GP estimator.

The second goal is to use the learned model to synthesize a stabilizing controller for the system and a ROA inner-approximation which holds for the true dynamics with high probability. In Section 8.3, we develop the theoretical results that make this possible: a theorem establishing an accuracy-confidence guarantee of the validity of an ROA inner approximation for a control policy and a Lyapunov function candidate compatible with the uncertainty bound of the previous section, and an SOS program to automatically construct such a policy and Lyapunov function if any exist.

The third goal is to design an exploration controller to maximize the information collected during the exploration trajectory while maintaining it inside the ROA. Section 8.4 introduces this controller, which is built on the stabilizing policy synthesized in the previous section. This auxiliary *exploration controller* guides the system state to regions with high variance under the GP model, thus providing informative new data. The overall learning/robustness procedure is laid out in Algorithm 7: the key idea is to collect data with the exploration controller, update the GP model, update the stabilizing policy and ROA approximation, and repeat. In Section 8.6, this approach is demonstrated on a nonlinear inverted pendulum, demonstrating Algorithm 7's ability to improve a conservative initial approximation for the ROA by judicious collection of data.

Notation

The subscript x_i denotes the i^{th} element of the vector x. The superscript $x^{(i)}$ with parentheses denotes the data point in the data set $\mathcal{D}$ with index i. The superscript x^i without parentheses denotes an object associated with the i^{th} iteration of an algorithm.

When applied to vectors, the orders $>$, $\leq$ are applied elementwise. The operator $\mathbb{E}[\cdot]$ denotes expectation with respect to a probability distribution.

For $\zeta \in \mathbb{R}^n$, $\mathbb{R}[\zeta]$ represents the set of polynomials in ζ with real coefficients, and $\mathbb{R}^m[\zeta]$ and $\mathbb{R}^{m\times p}[\zeta]$ denote all vector- and matrix-valued polynomial functions. The subset $\Sigma[\zeta] := \{\pi = \sum_{i=1}^{M} \pi_i^2 : \pi_1, ..., \pi_M \in \mathbb{R}[\zeta]\}$ of $\mathbb{R}[\zeta]$ is the set of SOS polynomials in ζ.

8.1 Model Structure and Assumptions

Consider a continuous-time nonlinear system of the form

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + \Delta(x(t)), \tag{8.2}$$

with state $x(t) \in \mathbb{R}^{n_x}$ and input $u(t) \in \mathbb{R}^{n_u}$. The system dynamics comprise a known nominal control-affine part, f and g, and an uncertainty Δ which we aim to learn.

We will assume the true model has a known equilibrium at the origin, so that a stabilizing policy and region of attraction (ROA) can be constructed.

Assumption 1. *The origin* $(x = 0, u = 0)$ *is an equilibrium of* (8.2)*, that is* $f(0) = w(0) = 0$.

To allow for sum-of-squares (SOS) analysis, we make the following assumption:

Assumption 2. *The known dynamics are polynomials:* $f(x) \in \mathbb{R}^{n_x}[x]$ *and* $g(x) \in \mathbb{R}^{n_x\times n_u}[x]$.

The true dynamics need not be polynomial, as the non-polynomial terms can be absorbed into Δ. We also assume that the unknown term can be approximated by a polynomial-kernel Gaussian process (GP).

Assumption 3. *The term $\Delta(x)$ can be approximated by a polynomial in a region $\mathcal{X} \in \mathbb{R}^{n_x}$ containing the origin. Specifically, for a given $\epsilon > 0$ there is a polynomial $q(x)$ such that $\|\Delta(x) - q(x)\| \leq \epsilon$ for all $x \in \mathcal{X}$.*

For example, if Δ is analytic in a ball $\mathcal{B}$ containing the origin, then Taylor's theorem ensures that Assumption 3 holds in $\mathcal{B}$.

Aside from any prior knowledge, our information about the system will come from measurements of the form $(x^{(i)}, u^{(i)}, \dot{x}^{(i)})$. Typically $\dot{x}$ itself is not directly measurable, and is estimated using a finite-difference approximation from measurements of x. The finite-difference approximation will be a noisy estimate of $\dot{x}$, and the measurements of x may in practice be noisy as well.

Assumption 4. *We have access to measurements of $\dot{x}$ which are corrupted by noise which is uniformly bounded by σ_n.*

Remark 4. *In* (8.2)*, we assume Δ depends only on x. If it depends both on x and u, we introduce an auxiliary input state $x_u(t) \in \mathbb{R}^{n_u}$ for u, and design the new input $v(t) \in \mathbb{R}^{n_u}$ for x_u. This leads to the augmented system*

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))x_u(t) + \Delta(x(t), x_u(t)) \\ \dot{x}_u(t) &= v(t), \end{aligned}$$

which recovers the form in (8.2)*. This formulation is demonstrated in Section 8.6.*

8.2 Estimating the Unmodeled Dynamics

To estimate the unknown term in a Bayesian framework, we must choose a *prior distribution* for the system dynamics. The prior model is a probability distribution of candidate functions for w, which represents what we know about the system prior to seeing any data. From Assumption 3 we know that the system can be approximated by a polynomial in a region about the equilibrium, so we will choose a prior over polynomial functions. Assumption 2 implies that $f(x) + g(x)u$ is an estimate for the true dynamics: assuming this is the best estimate we can make without data, we will take the prior mean for w to be zero.

We will use a GP as our prior distribution. A GP h is a probability distribution over functions which is completely characterized by its mean $m(x) = \mathbb{E}[h(x)]$ and covariance $k(x, y) = \mathbb{E}[(h(x) - m(x))(h(y) - m(y))]$. The covariance of a GP prior is also called the *kernel function* of the process. The kernel function determines the class of functions over which the distribution is defined. When $k(x, y)$ is polynomial in x and y, the distribution will be over a space of polynomial functions. We will therefore choose $k(x, y)$ to be a polynomial.

Typically, GPs are presented as distributions of scalar-valued functions. Since the unknown term w is vector-valued, we will model each entry w_i with a separate scalar-valued GP of functions with domain $\mathbb{R}^{n_x}$. We write our prior distribution for the dynamics as

$$\dot{x}(t) = f(x(t)) + g(x(t))u(t) + \hat{\Delta}(x(t)), \tag{8.3}$$

where $\hat{\Delta}(x)$ is a vector of GPs $\hat{\Delta}_i$, each with mean zero and kernel k_i.

As we collect data from a system trajectory, we condition the prior distribution on the data to obtain the *posterior distribution*. Like the prior, the posterior is a distribution over functions. For a GP prior, the posterior will also be a GP, but with a different mean and covariance which more accurately represent the ground truth than the prior.

GPs with polynomial kernels

Consider a scalar GP prior h with mean zero and kernel $k(x, y)$, and a data set $\mathcal{D} = \{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$ of states $x^{(i)} \in \mathbb{R}^{n_x}$ and labels $y^{(i)} \in \mathbb{R}$. Then the posterior distribution, that is the prior conditioned on the data, is also a GP, whose mean and variance have closed-form solutions [73]. The posterior mean has the form

$$m_i q(x) = \mathbb{E}[h(x)|\mathcal{D}] = y^\top (K + \sigma_n I)^{-1} k_* \tag{8.4}$$

where K is the kernel Gramian matrix with elements $(K)_{ij} = k(x^{(i)}, x^{(j)})$, k_* is the vector with elements $(k_*)_i = k(x, x^{(i)})$, and y is the vector with $y_i = y^{(i)}$. Letting $c = y^\top (K+\sigma_n I)^{-1}$, we can re-express the mean as

$$m(x) = \sum_{i=1}^{N} c_i k(x, x^{(i)}). \tag{8.5}$$

When $k(x, y)$ is a polynomial in x and y, (8.5) shows that $m(x)$ is also a polynomial, of the same degree as the kernel.

The posterior variance has the form

$$\begin{aligned} \sigma_q^2(x) &= \mathbb{E}[(h(x) - m(x))^2|\mathcal{D}] \\ &= k(x, x) - k_*^\top (K + \sigma_n I)^{-1} k_*. \end{aligned} \tag{8.6}$$

The posterior variance is a polynomial when $k(x, y)$ is a polynomial, whose degree is twice the degree of the posterior mean owing to the quadratic dependence on the prior covariance.

Choice of polynomial kernel

The spaces of polynomials from which the mean and variance are drawn depend on the specific choice of polynomial kernel. In particular, the mean is drawn from the *reproducing kernel Hilbert space* (RKHS) $\mathcal{H}(k)$ of the kernel k. The kernel must be chosen so that the functions in $\mathcal{H}(k)$ satisfy the Assumptions outlined in Section 9.1, in particular Assumption 1. The most common choices for polynomial kernels are the homogeneous polynomial kernel $k(x, y) = \alpha^2 (x^\top y)^p$, and the inhomogeneous polynomial kernel $k(x, y) = \alpha^2 (x^\top y + 1)^p$. Here, α^2 and p are hyperparameters: α^2 is a scaling parameter, and p sets the polynomial degree. In order for our estimate to satisfy Assumption 1, the mean of the GP must satisfy $m(0) = 0$. Since functions from the RKHS of the inhomogeneous kernel are generally not zero at the

origin (due to the constant term), they do not satisfy this assumption. To construct a suitable kernel, we use two classic results which follow from [6]:

Proposition 2. *The RKHS $\mathcal{H}(k)$ of the* homogeneous polynomial kernel $k(x,y) = \alpha^2(x^\top y)^p$ *is spanned by the monomials of degree p, that is by monomials $\prod_{i=1}^d x_i^{p_i}$ such that $\sum_i p_i = p$.*

Here, α^2 and p are hyperparameters: α^2 is a scaling factor, and p sets the polynomial degree.

Proposition 3. *Let k_1 and k_2 be two kernels of finite-dimensional RKHSs. Then $k_1 + k_2$ is also a kernel, and $\mathcal{H}(k_1 + k_2)$ is spanned by the concatenation of the spans of $\mathcal{H}(k_1)$ and $\mathcal{H}(k_2)$.*

For example, the function $(x^\top y)^2 + (x^\top y)^3$ is a kernel function whose RKHS is spanned by the monomials of degrees 2 and 3. This motivates the following choice of kernel:

$$k(x,y) = \alpha_1^2(x^\top y) + \alpha_2^2(x^\top y)^2 + \ldots + \alpha_p^2(x^\top y)^p. \tag{8.7}$$

By Propositions 2 and 3, the RKHS of this kernel is spanned by all monomials of degree $\leq p$ *except* for degree zero. In other words, the RKHS spans all polynomials q of degree $\leq p$ that satisfy $q(0) = 0$.

In Section 8.2 we will see that the range of possible unknown terms admitted by the prior model is bounded with high probability by a multiple of σ_q, so that a higher variance admits a larger class of functions for the unknown term. Therefore, any prior knowledge about the general form of the unknown term or the range of values it can take on should be used to select the kernel. While keeping the form (8.7), this information can be used to choose the hyperparameters α_i^2. For instance, if the baseline f and g are known to be accurate up to degree 2, then α_1^2 and α_2^2 can be set to small values, while the other α_i are set to high values. Another example is if the dynamics are known *a priori* to be even (or odd); then, the prior kernel need only contain terms of even (or odd) degree.

Probabilistic Bounds on the GP Model

The following inequality from [79] provides a probabilistic bound on the values that the functions in the distribution of a GP can take over its domain.

Lemma 7 (Theorem 6 of [79])**.** *Suppose we have data $\{x^{(i)}, y^{(i)}\}_{i=1}^N$ from a function $h \in \mathcal{H}(k)$ that satisfies $\|h\|_k \leq \infty$, where $\|\cdot\|_k$ is the norm of $\mathcal{H}(k)$. The data may be corrupted with noise uniformly bounded by σ_n. Let $\beta_N = 2\|h\|_k^2 + 300\gamma_N \log^3(N/\delta)$, where γ_N is the maximum mutual information that can be obtained for the GP prior with N samples corrupted with noise bounded by σ_n. Let $\delta \in (0,1)$. Then the inequality*

$$|h(x) - m_{\hat{h}}(x)| \leq \sqrt{\beta_N}\sigma_{\hat{h}}(x) \tag{8.8}$$

holds with probability $\geq 1 - \delta$, where $m_{\hat{h}}(x)$ and $\sigma_{\hat{h}}(x)$ are the mean and standard deviation of the GP $\hat{h}$ with kernel function k conditioned on the data.

The inequality (8.8) transforms the problem of providing a probabilistic guarantee for a GP into the problem of providing a guarantee over functions with a given upper and lower bound. In Section 8.3, we will show that for GPs with polynomial kernels, this further transforms into a problem that may be solved with SOS programming.

The assumptions we have made on the system allow us to use this inequality in our analysis.

Proposition 4. *For a kernel $k(x, y)$ of the form (8.7) with sufficiently high degree p, measurements of Δ can be used to construct a GP model which satisfies the inequality in Lemma 7.*

Proof. By Assumption 3, we know that in a region $\mathcal{X}$ containing the origin, each Δ_i can be approximated by a polynomial q_i with uniform error ϵ. The measurements of Δ_i are effectively measurements of Δ_i corrupted by this uniformly-bounded noise. Let p_{q_i} be the degree of this polynomial, and let $k_i(x, y)$ be a kernel of the form (8.7) with $p \geq p_{q_i}$. Then $q_i \in \mathcal{H}(k_i)$, by Propositions 2 and 3. Since $\mathcal{H}(k_i)$ is finite-dimensional, the norm $\|q_i\|_{k_i}$ is finite. By assumption 4, the measurements of Δ_i are also subject to an additional noise uniformly bounded by σ_n. Since the measurements of w_i act as measurements of q_i with noise uniformly bounded by $\sigma_n + \epsilon$, and $\|q_i\|_{k_i} \leq \infty$, the function q_i and the kernel k_i satisfy the assumptions of Lemma 7. □

The quantity γ_N is difficult to compute exactly for most kernels, and differs for each data set size N. However, for many commonly-used kernels it has a sublinear dependence on N, and can be effectively approximated up to a constant [79]. Following [15], we will assume through the rest of the paper that the quantity $\sqrt{\beta_N}$ can be bounded by a constant parameter η. The parameter η is higher for smaller values of δ, i.e. for probabilistic bounds of higher confidence.

8.3 Estimating the Region of Attraction

For safe learning with a GP model we must ensure that there is a region of state space which we are confident can be explored safely. To do this, we will synthesize a memoryless, state feedback control policy κ and a Lyapunov function V which guarantee that the closed-loop system

$$\dot{x}(t) = f(x(t)) + g(x(t))\kappa(x(t)) + \Delta(x(t)) \tag{8.9}$$

is stable around the origin with high confidence according to our model. By this, we mean that the closed-loop system satisfies the inequality

$$\frac{\partial V(x)}{\partial x} \cdot (f(x) + g(x)\kappa(x) + \Delta(x)) < 0 \tag{8.10}$$

with probability $\geq 1 - \delta$, with $\delta \in (0, 1)$, for all points in a set $\mathcal{A}\backslash 0$, where

$$\mathcal{A} = \{x \in \mathbb{R}^{n_x} | V(x) \leq \gamma\}, \text{ for some } \gamma > 0. \tag{8.11}$$

This set is an inner-approximation of the origin's ROA, which we will labor to make as large as possible.

We can ensure that the inequality (8.10) holds with high probability on the true dynamics by ensuring it holds for the deterministic bounds $|\Delta_i(x) - m_i(x)| \leq \eta\sigma_i(x)$, $i = 1, \ldots, n_x$, which follow from Lemma 7. The link between the deterministic bound on the GP model and the probabilistic guarantee of stability for the true dynamics is stated in the following theorem.

Theorem 5. *Suppose the true dynamics satisfy the assumptions outlined in Section 9.1. Let η be a bound on the parameter $\sqrt{\beta_N}$ such that Lemma 7 holds for each Δ_i with a given $\delta \in (0, 1)$. Let σ be the vector of standard deviations comprising the standard deviations σ_{qi} of the GPs $\hat{\Delta}_i$. Given f, g defined in (8.2), and $\gamma > 0$, if there exists a control law $\kappa : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$, and a $\mathcal{C}^1$ function $V : \mathbb{R}^{n_x} \to \mathbb{R}$, such that $V(0) = 0$ and $V(x) > 0$ for all $x \in \mathbb{R}^{n_x} \backslash 0$, and*

$$\frac{\partial V(x)}{\partial x} \cdot (f(x) + g(x)\kappa(x) + m_q(x) + d(x)) < 0 \tag{8.12}$$

holds in a bounded region $\mathcal{A}\backslash 0 \subset \mathbb{R}^{n_x}$ for all vector functions d bounded by $-\eta\sigma(x) \leq d(x) \leq \eta\sigma(x)$, then $\mathcal{A}$ is an inner-approximation to the ROA of (8.2) with probability $\geq 1 - \delta$ over the GP distribution.

Proof. Proposition 4 establishes that Lemma 7 holds for the true dynamics and the GP model. Therefore, the bounds $|\Delta_i(x) - m_{qi}(x)| \leq \eta\sigma_i(x)$, and equivalently the bound $-\eta\sigma(x) \leq d(x) \leq \eta\sigma(x)$, hold with probability $\geq 1 - \delta$. Since V guarantees that (8.12) holds for $-\eta\sigma(x) \leq d(x) \leq \eta\sigma(x)$, it follows by Lemma 7 that the same V ensures that (8.10) holds with probability $\geq 1 - \delta$ for $x \in \mathcal{A}\backslash 0$. This ensures that $\mathcal{A}$ is a ROA inner-approximation for the true dynamics with probability $\geq 1 - \delta$. □

We restrict decision variables V and κ to be polynomials in order to use SOS analysis [52] to synthesize them such that the condition (8.12) holds. The condition (8.12) is a set containment constraint, and the generalized S-procedure [67] can be used to derive the corresponding SOS constraint for it. To do this, we must express the bound $-\eta\sigma(x) \leq d(x) \leq \eta\sigma(x)$ as a semi-algebraic set. The bound can be described by a number of n_x quadratic constraints: for $i = 1, ..., n_x$,

$$\eta^2\sigma_i(x)^2 - d_i^2(x) = \eta^2\sigma_{qi}^2(x) - d_i^2(x) \geq 0 \ \forall x, \tag{8.13}$$

which use the polynomial Var directly. Define polynomials $p_{d,i}(x, d) = \eta^2\sigma_{qi}^2(x) - d_i^2$ for $i = 1, ..., n_x$. By choosing the volume of $\mathcal{A}$ as the reward function to be maximized, and applying the generalized S-procedure to (8.12), we obtain the following SOS optimization problem:

$$\begin{aligned} &\sup_{V,\kappa,s} \text{Volume}(\mathcal{A}) \\ &\text{s.t.} \ \ s_V, s_{d,i} \in \Sigma[(x, d)], \end{aligned}$$

$$V - \epsilon_1 x^\top x \in \Sigma[x], \kappa \in \mathbb{R}^{n_u}[x], \tag{8.14a}$$

$$-\left(\tfrac{\partial V}{\partial x} \cdot (f + g\kappa + m_q + d) + \epsilon_2 [x; d]^\top [x; d]\right) + (V - \gamma) s_V - \sum_{i=1}^{n_x} s_{d,i} p_{d,i} \in \Sigma[(x, d)], \tag{8.14b}$$

where ϵ_1 and ϵ_2 are small positive numbers. The optimization (8.14) is a non-convex problem, since it is bilinear in two sets of decision variables V and (s_V, κ). It can be handled by alternating the search over these two sets of decision variables, since holding one set fixed while optimizing over the other results in a convex problem. The procedure is summarized in Algorithm 8 in Appendix 8.8.

8.4 Exploring the Region of Attraction

In order to increase the information gained from the trajectory data, we would like for each trajectory to explore a different region of the state space while remaining in the ROA inner-approximation. While the policy κ synthesized from SOS programming ensures that system stays in the inner-approximation, it does not ensure that new areas of the state space will be explored. Therefore, as an alternative to the control policy $\kappa(x)$, we propose an *exploration policy* $\kappa_e(x)$ which guides the system to areas of the state space with little data.

The posterior variances $\sigma_{qi}(x)$ of the GPs $\hat{\Delta}_i$ can be used to track which areas of the state space have not been visited. In regions close to a data point, $\sigma_{qi}^2(x)$ will be close to the noise level σ_n^2; in regions far from any data, $\sigma_{qi}^2(x)$ will be close to the prior variance $k(x, x)$. Therefore, guiding the system to areas of high variance will lead it to areas which have not been explored. We can ensure this by choosing κ_e to increase $\sigma_{qi}^2(x(t))$, the variance of $\hat{\Delta}_i$ at the current state, over time. To account for each $\hat{\Sigma}_i$, we will try to increase the sum $\sum_i \sigma_{qi}^2(x(t))$.

The exploration policy will choose a control action by solving an optimization problem. The problem will be to maximize the time derivative of $\sum_i \sigma_{qi}^2(x(t))$, the sum of the variances at the present system state. The derivative is

$$\begin{aligned} \tfrac{d}{dt} \sum_{i=1}^{n_x} \sigma_{qi}^2 &= \sum_{i=1}^{n_x} \tfrac{\partial \sigma_{qi}^2}{\partial x} \dot{x} \\ &= \left(\sum_{i=1}^{n_x} \tfrac{\partial Var_i}{\partial x} \right) (f + g\kappa_e + \hat{w}). \end{aligned} \tag{8.15}$$

To maximize this expression using $\kappa_e(x)$ as a decision variable, we need only consider the $(\sum_i \frac{\partial \sigma_{qi}^2}{\partial x}) g\kappa_e$ term.

At the same time, $\kappa_e(x)$ must not take the system outside of the ROA. Therefore, κ_e must satisfy

$$\dot{V}_{low} = \frac{\partial V}{\partial x} \cdot (f + g\kappa_e + m - \eta\sigma) \leq 0, \tag{8.16}$$

$$\dot{V}_{up} = \frac{\partial V}{\partial x} \cdot (f + g\kappa_e + m + \eta\sigma) \leq 0. \tag{8.17}$$

To ensure a unique solution, we will also include a quadratic regularizing term on κ_e in the objective. The form of the exploration policy κ_e is then

$$\begin{aligned} \kappa_e(x) = \quad & \arg\max_u \left(\sum_{i=1}^{n_x} \frac{\partial \sigma_{qi}^2}{\partial x} \right) gu - \lambda u^\top u \\ & \text{s.t. } \frac{\partial V}{\partial x} \cdot (f + gu + m - \eta\sigma) \leq 0 \\ & \quad \frac{\partial V}{\partial x} \cdot (f + gu + m + \eta\sigma) \leq 0, \end{aligned} \tag{8.18}$$

where $\lambda > 0$ is a regularization parameter. The policy (8.18) is a quadratic program, since for a fixed x the objective is quadratic and the constraints linear in u. Since quadratic programs can be efficiently solved in real time, the exploration policy is suitable for online use.

When the policy κ from (8.14) exists, $u = \kappa(x)$ is a feasible solution to (8.18). This means that (8.18) is feasible when (8.14) is feasible.

8.5 An Algorithm for Safe Learning

Algorithm 7 below shows how the results of sections 8.2, 8.3, and 8.4 can be combined to perform safe exploration and robust policy synthesis.

The first step is to establish the prior information available for the system dynamics and encode it into a prior model. This comprises choosing the terms f and g in the control-affine base model, and selecting a prior kernel $k(x, y)$ for the unknown term. The base model f and g may come, for example, from a linearized model of the system. The kernel should be chosen so as to capture any further knowledge about the unknown part of the dynamics.

With the prior model in place, the next step is to synthesize a prior control policy κ^0, a prior Lyapunov function V^0 and a prior γ^0 by solving the SOS program (8.14) using Algorithm 8. The prior Lyapunov function acts as a certificate that κ^0 stabilizes the equilibrium with high probability, that is for a large probability mass of candidates for w admitted by the prior model. Sublevel sets of V^0 also act as inner-approximations of the ROA created by κ^0. We take the prior ROA as the sublevel set $\mathcal{A}^0 = \{x \in \mathbb{R}^{n_x} | V^0(x) \leq \gamma^0\}$.

After synthesizing the prior ROA inner-approximation, the next step is to collect data to form the posterior model. This data will come from a system trajectory whose initial condition we may choose. In order to collect data safely, we choose an initial condition inside the prior ROA estimate (step 4), so that the system is guaranteed to eventually return to the origin. Rather than use the prior policy κ^0, we will use the exploration policy

κ_e to guide the trajectory of the system during data collection (step 5). This will ensure that the system trajectory visits regions where model variance (i.e. uncertainty) is high.

After collecting data, the next step (step 6) is to compute the posterior model using (8.4) and (8.6). With the posterior model, we can solve the SOS program (8.14) using the posterior model (step 7) to synthesize a posterior policy κ^1 and posterior Lyapunov function V^1, and compute an inner-approximation of the ROA for the posterior policy. Since κ^1 and V^1 are computed using a more accurate model of the dynamics, the posterior ROA estimate will generally be larger than the one for the prior policy.

With the posterior policy in place, we can repeat steps 4 through 7 to update the posterior model any number of times before stopping. Supposing that we perform T iterations of this process, the final output of the algorithm will be the posterior model, the posterior policy κ^T, and the posterior ROA estimate $\mathcal{A}^T$.

Algorithm 7: Bayesian Safe ROA Learning with a polynomial GP model

Input: Base model $f(x) + g(x)u$; prior kernel degree p and hyperparameters $\{\alpha_i^2\}_{i=1}^p$; GP regression noise parameter σ_n^2; number T of iterations.

Output: Posterior control policy κ^T; posterior Lyapunov function V^T; posterior ROA $\mathcal{A}^T = \{x \in \mathbb{R}^{n_x} | V^T(x) \leq \gamma^T\}$

1 Construct the prior model $\dot{x} = f(x) + g(x)u + \hat{\Delta}(x)$, where $\hat{\Delta}(x)$ is a GP with mean zero and kernel $k(x, y) = \alpha_1^2(x^\top y) + \ldots + \alpha_p^2(x^\top y)^p$. Construct an empty data set $\mathcal{D}^0 = \{\}$;

2 Solve the SOS program described in (8.14) using the prior model to compute the prior policy κ^0, prior Lyapunov function V^0, and prior ROA $\mathcal{A}^0$;

3 **for** $i \in \{1, \ldots, T\}$ **do**

4 Select an initial condition $x_0^{i-1} \in \mathcal{A}^{i-1}$;

5 Collect data $\{x^{(j)}\}_{j=1}^{N^i}$ and $\{\dot{x}^{(j)}\}_{j=1}^{N^i}$ of N^i points from a trajectory on the true dynamics with initial condition x_0^{i-1}, using the exploration policy κ_e^i defined in (8.18). Add this to the data set, setting $\mathcal{D}^i = \mathcal{D}^{i-1} \cup \{(x^{(j)}, \dot{x}^{(j)})\}_{j=1}^{N^i}$;

6 Use (8.4) and (8.6) to compute the mean and variance of the GP with the data set $\mathcal{D}^i$;

7 Solve the SOS program described in (8.14) using the posterior model to compute the posterior policy κ^i, posterior Lyapunov function V^i, and posterior ROA $\mathcal{A}^i$;

8.6 Example: Inverted Pendulum with Input Saturation

In this section, we demonstrate Algorithm 7 by using it to investigate the dynamics near the unstable equilibrium of a two-state inverted pendulum model. The pendulum model,

adapted from [14], includes an input saturation which prevents the system from being globally stabilized. Reference [14] analyzes the stability of this system for a fixed policy and Lyapunov function determined from a linearized model, and uses a GP with a non-polynomial kernel to verify a sublevel set of the fixed Lyapunov function as a ROA estimate. For our analysis, we do not need a prior safe policy and Lyapunov function to be given: we instead take a prior model (also based on a linearization) and a kernel function, and use it to synthesize a prior controller and a ROA inner-approximation. We then collect a trajectory inside the prior ROA using the exploration policy, and use this data to compute a posterior model and synthesize a new policy and ROA.

The true system dynamics for the pendulum are

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \tfrac{g}{\ell}\sin(x_1) - \tfrac{\mu}{M\ell^2}x_2 + \tfrac{1}{M\ell^2}\mathrm{Sat}(u), \end{aligned} \tag{8.19}$$

where M is the mass of the pendulum, ℓ is its length, and g is gravitational acceleration. The coordinates are chosen so that $x_1 = 0, x_2 = 0$ is the unstable equilibrium. The $\mathrm{Sat}(\cdot)$ function limits the input action to stay within the range $[-Mg\ell\sin(30^\circ), Mg\ell\sin(30^\circ)]$. With this input saturation in place, the inverted pendulum cannot return to the upright position once it deviates from upright by more than 30 degrees.

The input saturation also means that this system is not input-affine. To remedy this, we use the formulation from the Remark in Section 9.1: we introduce an auxiliary *input state* x_u and augment (8.19) to

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \tfrac{g}{\ell}\sin(x_1) - \tfrac{\mu}{M\ell^2}x_2 + \tfrac{1}{M\ell^2}\mathrm{Sat}(x_u) \\ \dot{x}_u &= v. \end{aligned} \tag{8.20}$$

We start the algorithm with the linearization of (8.20); that is, we take

$$f(x) = \begin{bmatrix} x_2 \\ \frac{g}{\ell}x_1 - \frac{\mu}{M\ell^2}x_2 + \frac{1}{M\ell^2}x_u \\ 0 \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{8.21}$$

as the inputs f and g to Algorithm 7. For the prior kernel, we use the degree 3 kernel

$$k(x, y) = \alpha_1^2(x^\top y) + \alpha_2^2(x^\top y)^2 + \alpha_3^2(x^\top y)^3. \tag{8.22}$$

Since the dynamics of x_1 are purely kinematic, we can assume that the given model is accurate. Similarly, since x_u is a constructed state, we can assume its dynamics are accurate. Therefore, we assume that the vector of unknown dynamics has the form $\Delta(x) = [0 \;\; \Delta_2(x) \;\; 0]^\top$, requiring only one GP model for the unknown dynamics of x_2.

To complete the prior model, we select kernel hyperparameters for $\Delta_2(x)$. We will take as prior knowledge that our linearization is accurate, and that the nonlinear terms contain a strong odd component. We incorporate this knowledge into the prior model by setting α_1^2

and α_2^2 to a small value, namely $\alpha_1^2 = \alpha_2^2 = 0.075$. Since we have no further prior knowledge of the third-order term, we will set α_3^2 to be larger than α_1 and α_2, namely $\alpha_3^2 = 1.5$. We will also assume that our $\dot{x}$ measurements, taken from a finite-difference approximation on the observed states, are reasonably accurate, and use this knowledge by setting the GP regression noise parameter σ_n^2 to a low value value, namely $\sigma_n^2 = 0.01$.

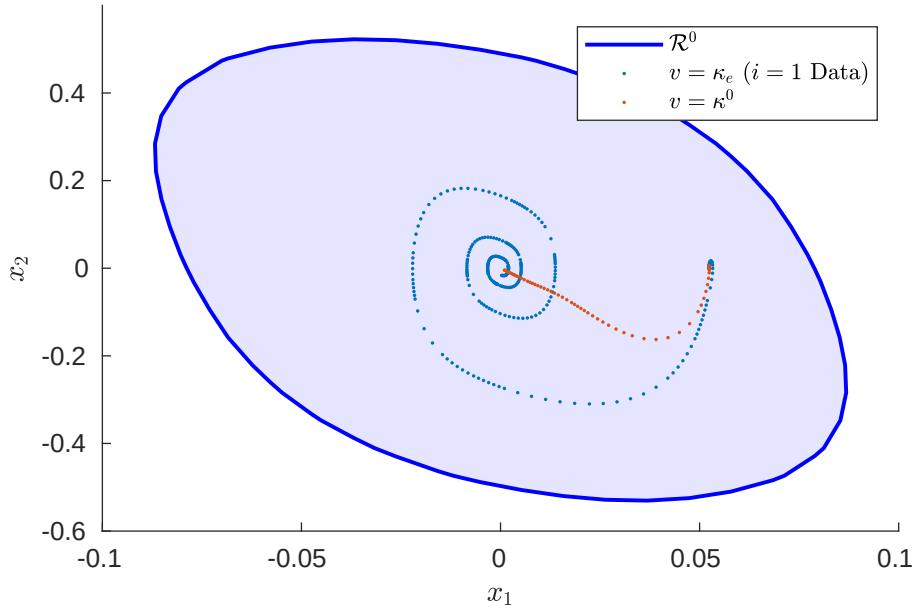

Figure 8.1: The prior ROA computed using the prior system (8.21) and prior kernel (8.22), projected onto x_1 and x_2. Two trajectories are also shown using the two prior control policies, the base SOS policy $v(t) = \kappa(x(t))$ and the exploration policy $v(t) = \kappa_e(x(t))$. The exploration policy visits more of the state space than the base policy.

Figure 8.1 shows the results of lines 2, 4, and 5 of Algorithm 7 using the selected f, g, and k. The decision variables in the SOS analysis—V^0, κ^0, and the S-procedure certificates—$s_V, s_{d,i}, s_\gamma$ are degree 4 polynomials, and we take $\eta = 3$. The prior ROA certifies that the prior policy κ^0 can restore angle deviations in the range of about ± 4.5 degrees, starting from rest, in the presence of any w_2 that is bounded above and below by $-\eta\sigma_2(x) \leq \Delta_2(x) \leq \eta\sigma_2(x)$, where $\sigma_2(x) = \sqrt{k(x,x)}$ is the prior variance.

For step 4, we select an initial condition which starts from rest with an initial angle deviation of 3°; that is, we take $x_1 = 3\pi/180$, $x_2 = x_u = 0$. Figure 8.1 shows data from two trajectories on the true dynamics with this initial condition. One trajectory, following step

5 of Algorithm 7, uses the exploration policy $v(t) = \kappa_e(x(t))$. This trajectory will be used as the data set $\mathcal{D}^0$ for the next step. The other trajectory uses the prior policy κ^0, from the same initial condition chosen in step 4. The exploration policy provides data from a wider area of the prior ROA before settling to the equilibrium, by allowing more transients to remain than the prior policy.

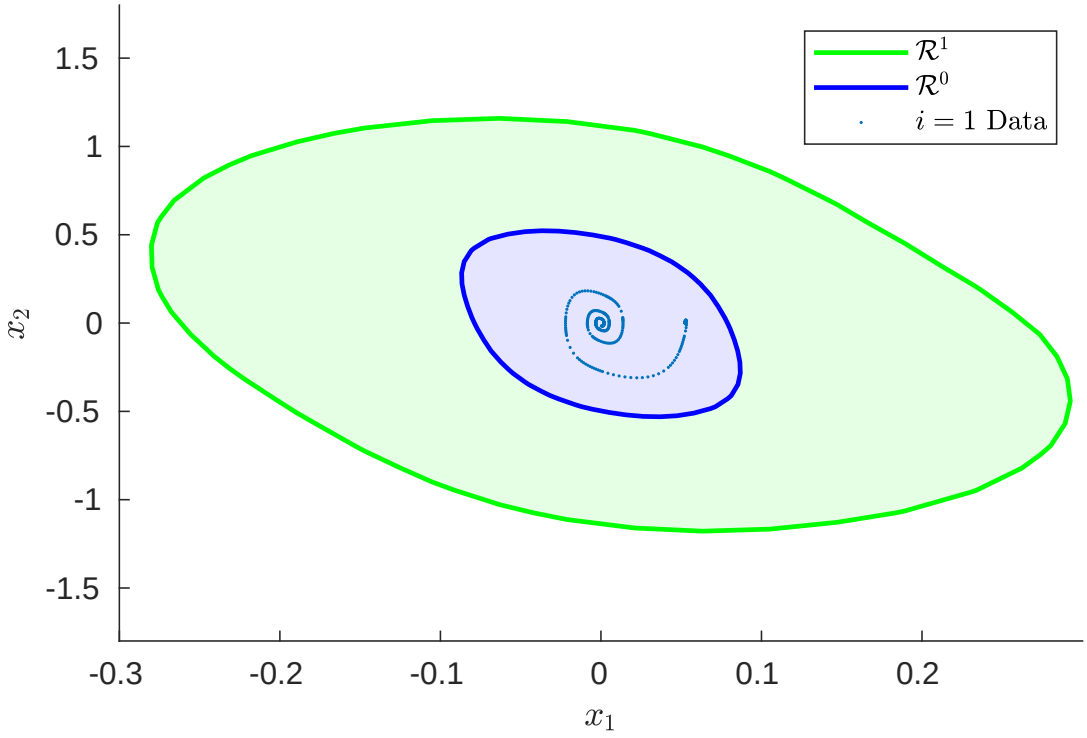

Figure 8.2: The posterior ROA, projected onto x_1 and x_2. The posterior model incorporates the data collected by the exploration trajectory from iteration $i = 1$ of Algorithm 7.

Figure 8.2 shows the posterior ROA $\mathcal{A}^1$ computed by step 7 of Algorithm 7. The posterior model was computed using the data set $\mathcal{D}^0$ comprising the x data points from the prior trajectory with the exploration policy and finite-difference approximations for $\dot{x}$. With $T = 1$, this is the final step of the algorithm. By incorporating the trajectory data, the posterior analysis successfully extends the size of the ROA. In particular, the range of safe angle deviations from rest is extended to ± 16.5 degrees.

There are two mechanisms which allow the posterior ROA to be larger. First, the posterior model more closely matches the true dynamics than the prior, since it includes higher-order terms that are fit from data. Second, the posterior variance is less than the prior variance at all points in the state space. Since the variance determines the constraints on w in the SOS problem, the posterior controller can be robust against a smaller class of unknowns than the prior model while upholding the same probabilistic guarantee.

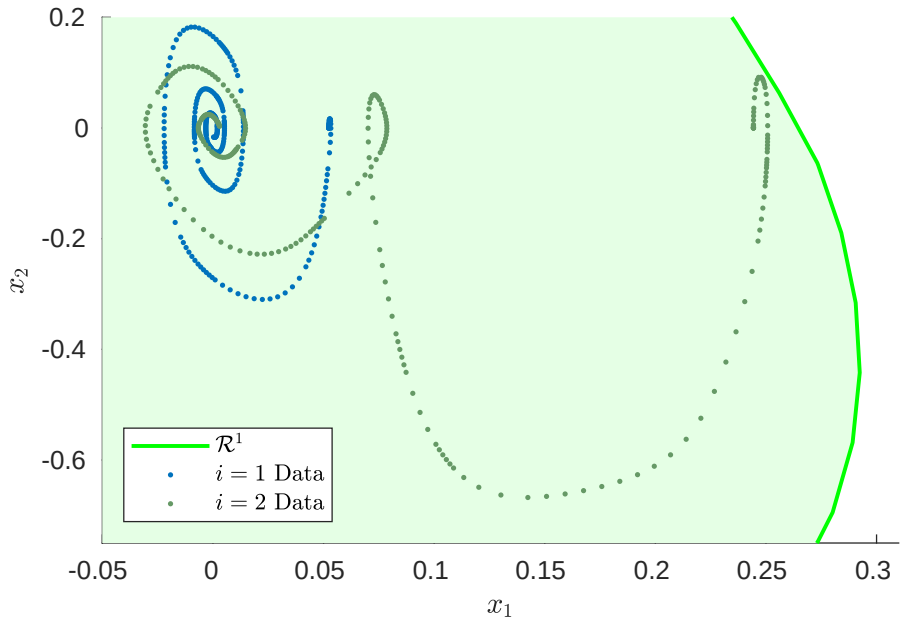

Figure 8.3: The exploration trajectory from iteration $i = 2$ of Algorithm 7, projected onto x_1 and x_2. The objective of (8.18) encourages the exploration policy to avoid the data from previous trajectories.

Though only one trajectory is needed to complete one iteration of Algorithm 7, we demonstrate how the exploration policy responds to data from previous iterations by simulating an exploration trajectory for the $i = 2$ iteration. To that end, we pick an initial condition in the posterior ROA, starting from rest with an initial angle deviation of 14°, and simulate a trajectory on the true dynamics using the exploration policy guided by the posterior variance.

The resulting trajectory is shown in Figure 8.3. Recall that, by maximizing the objective in (8.18), the exploration policy is encouraged to increase the total variance of states that the system visits. The exploration policy increases the information gained by the $i = 2$ trajectory in two ways. First, it avoids the $i = 1$ trajectory as much as possible, so that it does not collect data in parts of the state space that have already been visited. Second, it adds several excitations into the trajectory—once at the start of the trajectory, and again near $(x_1 = .075, x_2 = 0)$– where it briefly reverses direction, increasing the amount of time the system can explore before settling into the equilibrium.

8.7 Conclusion

The proposed method can take an initial prior model for the dynamics of a system and improve the model using data, while ensuring that the process of collecting takes place in a safe ROA. Since the GP model learns an estimate for the system dynamics in closed-form with quantified uncertainty, the learned model can be guaranteed safe with high confidence. The proposed method lifts two limitations faced by earlier work in safe learning with GPs. First, we are not restricted to a fixed, given policy and Lyapunov function: using polynomial kernel functions allows for policies and Lyapunov functions to be synthesized by SOS analysis. Second, we do not need to assume the existence of an *a priori* safe controller to initialize the safe learning process: by establishing prior information into a Bayesian prior model, we can compute an exploration controller which is guaranteed to be safe on the prior model dynamics.

However, the restriction to polynomial kernels places a limit on the types of unknown dynamics the system can learn. In particular, the condition that the unknown dynamics be well-approximated by a polynomial prohibits the method from learning dynamics with discrete transitions or discontinuities. Extending the method to work on a larger class of dynamics would increase the utility of the method. Another useful extension would be to allow for the synthesis of other types of safety guarantees than ROAs for learned systems, for instance barrier certificates or reachable sets.

A more fundamental limitation of the method proposed in this chapter is the restriction to static additive uncertainties. We have implicitly assumed– as many data-driven state-space methods do– that our chosen state space is rich enough to capture the full dynamical state of the system. If we have in fact "missed" some states, this chapter does not tell us how to learn of their existence and add them to the model, since we can only refine the dynamics of the same state variables that have been there from the start. Now, unmodeled dynamics can still be modeled as an uncertainty, but not an additive one: they hail from the class of *feedback uncertainties*. While the GP-based method of this chapter cannot model feedback uncertainties, this is not true of GP models in general. In the next chapter, we will investigate a class of GPs explicitly designed to model feedback uncertainties, specifically as a nonparametric frequency-domain that is capable of learning feedback uncertainties of arbitrary dynamical order.

8.8 Appendix: Iterative algorithm for solving the SOS problem (8.14)

A linear state feedback for the linearization of f and g about the origin is used to compute the initial iterate, $\bar{V}$.

Algorithm 8: Iterative method for solving (8.14)

Input: function $\bar{V}$ such that constraints (8.14) are feasible by proper choice of $s_V, s_{d,i}, \kappa, \gamma$.

Output: (κ, γ, V) such that with the volume of $\mathcal{A}$ having been enlarged.

for $j \in \{1, ..., N_{iter}\}$ **do**

  γ-step: decision variables $(s_V, s_{d,i}, \kappa, \gamma)$. Maximize γ subject to (8.14) using $V = \bar{V}$. This yields $(\bar{s}_V, \bar{\kappa})$ and optimal reward $\bar{\gamma}$.

  V-step: decision variables $(s_\gamma, s_{d,i}, V)$; Maximize the feasibility subject to (8.14) as well as $s_\gamma \in \Sigma[x]$, and

$$(\bar{\gamma} - V) + (\bar{V} - \bar{\gamma}) s_\gamma \in \Sigma[x], \tag{8.23}$$

  using $\gamma = \bar{\gamma}, s_V = \bar{s}_V, \kappa = \bar{\kappa}$. This yields $\bar{V}$.

Chapter 9

H_∞ Gaussian Processes for Frequency-domain Uncertainties

This chapter develops the foundational theory for a class of GPs designed to serve as non-parametric, frequency-domain, input-output models for feedback uncertainties. The defining property of this class is that realizations of the process are always H_∞ functions (i.e. transfer functions of stable, causal systems), so we call it the class of H_∞ Gaussian processes.

Probabilistic models of input-output dynamical systems, where the input-output relationship itself contains probabilistic elements separate from input noise or measurement error, have important applications both in system identification and probabilistic robust control. In system identification, probabilistic systems act as models of prior belief in Bayesian estimates of the system dynamics. In robust control, probabilistic models form the core of probabilistic robustness analysis: the objective is then to verify, or design a controller such that, the ensemble of uncertainties for which the system is stabilized has high probability.

These two lines of research– Bayesian system identification and probabilistic robust control– have generally been developed separately, and use different types of probabilistic models. This chapter investigates a class of probabilistic models for uncertain systems that is amenable in equal parts to both, allowing for safe learning and to robustness analysis in the framework of linear robust control. As discussed in Section 7, this is the class of H_∞ Gaussian processes. An uncertain system modeled by an H_∞ Gaussian process admits refinement through data through Bayesian regression: By conditioning on point observations in the frequency domain, the model becomes more accurate, though still uncertain in unobserved frequency ranges. On the other hand, an H_∞ Gaussian process model admits robustness analysis by virtue of the fact that it represents an ensemble of systems, precisely as in robust control, with the additional structure of a weight (the probability measure) over members of the ensemble.

The analysis in this chapter has three goals. The first goal is to provide a mathematical foundation for the class of H_∞ Gaussian Processes. Section 9.1 introduces the system setup, reviews background information on complex-valued random variables and stochastic processes, and introduces the classes of H_∞ GPs. Section 9.2 then provides a set of sufficient

conditions (Theorem 6 and Proposition 6) for a complex-valued GP to be an H_∞ GP. In addition to the general conditions, we provide a complete characterization (Theorem 7) of the covariance structure of a special class of H_∞ Gaussian process, namely those whose Hermitian covariance is stationary. Each Hermitian stationary H_∞ process is parameterized by a summable sequence of nonnegative reals, which lead to computationally tractable closed forms for certain choices of sequences.

The second goal is to establish how to refine an H_∞ GP using data. Section 9.3 reviews widely linear and strictly linear complex estimators for complex Gaussian process regression and presents numerical examples of Bayesian system identification. Contrary to other recent work in Bayesian system identification, we choose to use the *strictly linear estimator* for our Gaussian process models instead of the *widely linear estimator*. Although the widely linear estimate is superior for general processes, we find that for H_∞ Gaussian process models the strictly linear estimator works nearly as well while being simpler and more stable to compute than the widely linear estimator. To verify the utility of H_∞ GP models for Bayesian transfer function estimation, we apply the technique to two second-order systems using a mixture of a Hermitian stationary H_∞ processes constructed with Theorem 7 and an H_∞ process designed to model resonance peaks.

The final goal is to establish probabilistic guarantees of robustness for system models that use an H_∞ GP as a feedback uncertainty. In Section 9.4, we shall see that establishing accuracy-only probabilistic guarantees for a number of several robustness certificates– particularly those based on small-gain arguments and IQCs– comes down to establishing an inequality of the form $\mathbb{P}\left(\|f\|_\infty < u\right) \geq 1 - \delta$ for some H_∞ Gaussian process f with known mean and covariance functions. Bounds of this form, in turn, can be established by computing the expected number of gain upcrossings of f, which can be carried out by means of Belyaev formulas.

Notation

For a complex vector or matrix X, X^* denotes the complex conjugate and X^H denotes the conjugate transpose. We denote the exterior of the unit disk as $E = \{Z \in \mathbb{C} : |z| > 1\}$, and its closure as $\bar{E} = \{Z \in \mathbb{C} : |z| \geq 1\}$. L_2 is the Hilbert space of functions $f : \mathbb{C} \to \mathbb{C}$ such that $\int_{-\pi}^{\pi} |f(Re^{j\Omega})|^2 d\Omega < \infty$, equipped with the inner product $\langle f, g \rangle_2 = \int_{-\pi}^{\pi} f(e^{j\Omega}) g^*(e^{j\Omega}) d\Omega$. H_2 is the Hilbert space of functions $f : \mathbb{C} \to \mathbb{C}$ that are bounded and analytic for all $z \in E$ and $\int_{-\pi}^{\pi} |f(Re^{j\Omega})|^2 d\Omega < \infty$, equipped with the inner product $\langle f, g \rangle_2 = \int_{-\pi}^{\pi} f(e^{j\Omega}) g^*(e^{j\Omega}) d\Omega$. It is a vector subspace of L_2. H_∞ is the Banach space of functions $f : \bar{E} \to \mathbb{C}$ that are bounded and analytic for all $z \in E$ and $\sup_{\Omega \in [-\pi, \pi]} |f(e^{j\Omega})| < \infty$, equipped with the norm $\|f\|_\infty = \sup_{\Omega \in [-\pi, \pi]} |f(e^{j\Omega})|$. ℓ^1 is the space of absolutely summable sequences, that is sequences $\{a_n\}_{n=0}^{\infty}$ such that $\sum_{n=0}^{\infty} |a_n| < \infty$. $\mathcal{N}(\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and covariance Σ; likewise, $\mathcal{CN}\left(\mu, \Sigma, \tilde{\Sigma}\right)$ denotes a complex Gaussian distribution with mean μ, Hermitian covariance Σ, and complementary covariance $\tilde{\Sigma}$.

9.1 Preliminaries

H_∞ Gaussian processes are nonparametric statistical models for causal, LTI, BIBO stable systems in the frequency domain. Since our main focus will be the probabilistic aspects of the model, we restrict our attention to the simplest dynamical case: a single-input single-output system in discrete time. Thus, our dynamical systems are frequency-domain multiplier operators $H_f : L_2 \to L_2$ whose output is defined pointwise as $(H_f u)(\omega) = f(\omega)u(\omega)$, where $f : \mathbb{C} \to \mathbb{C}$ is the system's transfer function. Thanks to the bijection $H_f \leftrightarrow f$, we generally mean the function f when we refer to "the system".

Since our aim is to construct a probabilistic model for the system that is not restricted to a finite number of parameters, we must work directly with random complex functions of a complex variable: this is a special type of complex stochastic process that we call a z-domain process.

Definition 1. *Let $(\Xi, F, \mathbb{P})$ denote a probability space. A z-domain stochastic process with domain $D \subseteq \mathbb{C}$ is a function $f : \Xi \times D \to \mathbb{C}$.*

Note that each value of $\xi \in \Xi$ yields a function $f_\xi = f(\xi, \cdot) : D \to \mathbb{C}$, which is called either a "realization" or a "sample path" of f. If we take ξ to be selected at random according to the probability law $\mathbb{P}$, then f_ξ represents a "random function" in the frequentist sense. Alternatively, if we have a prior belief about the likelihood of some f_ξ over others, we may encode this belief in a Bayesian sense using the measure $\mathbb{P}$. We drop the dependence of f on ξ from the notation outside of definitions, as it will be clear when $f(z)$ refers to the random variable $f(\cdot, z)$ or when f stands for a realization f_ξ.

Complex Random Variables and Stochastic Processes

Complex random variables and processes are essentially no different from real ones, but certain statistical descriptions for real random variables do not extend to the complex case unless suitably augmented. The following is an example of what can go wrong.

Example 1. *A real Gaussian random variable is completely determined by its mean and variance. However, this is not true for complex Gaussian random variables. Consider the random variables $Z = X + jY$, and $W = j\sqrt{2}X$, where $X, Y \overset{i.i.d.}{\sim} \mathcal{N}(0, 1)$. These are distinct random variables, as evidenced by the fact that they have different supports; however, their means are $\mathbb{E}[Z] = \mathbb{E}[W] = 0$, and their variances are $\mathbb{E}[ZZ^*] = \mathbb{E}[WW^*] = 2$.*

We have defined the variance of a mean-zero complex random variable Z to be $\mathbb{E}[ZZ^*]$: this is required in order for the variance to be a real nonnegative number, and specializes to the standard variance in the purely real case.

What statistic, not required in the real case, distinguishes Z and W? It turns out to be the "real" variance $\mathbb{E}[Z^2]$; in the example, we have $\mathbb{E}[Z^2] = 0$ and $\mathbb{E}[W^2] = -2$. In general, a complex Gaussian Z is completely specified by $\mathbb{E}[Z]$, $\mathbb{E}[ZZ^*]$, and $\mathbb{E}[Z^2]$. We call the

statistic $\sigma^2 = \mathbb{E}[ZZ^*]$ the *Hermitian variance*, and $\tilde{\sigma}^2 = \mathbb{E}[Z^2]$ the *complementary variance*.[1] We carry this nomenclature to stochastic processes, assigning to a z-domain process the Hermitian covariance function $k(z,w) = \mathbb{E}[f(z)f^*(w)]$ and complementary covariance function $\tilde{k}(z,w) = \mathbb{E}[f(z)f(w)]$.

Definition 2. *A Gaussian z-domain process is a z-domain process f such that, for any n, the random vector $(f(z_1), \ldots, f(z_n))$ is complex multivariate Gaussian-distributed for all $(z_1, \ldots, z_n) \in D^n$.*

Analogous to the way that a real Gaussian process is determined by its mean and covariance, a Gaussian z-domain process is completely specified by its mean $m : D \to \mathbb{C}$, Hermitian covariance $k : D \times D \to \mathbb{C}$, and complementary covariance $\tilde{k} : D \times D \to \mathbb{C}$.

A complex random variable $f = x + jy$ may also be represented in its *augmented form* $\underline{f} = (f, f^*)$. This form is useful despite being redundant, as it yields a convenient expression of the second-order statistics in terms of an *augmented covariance matrix*

$$\underline{K}_f = \mathbb{E}\left[\begin{bmatrix} f \\ f^* \end{bmatrix} \begin{bmatrix} f^* & f \end{bmatrix}\right] = \begin{bmatrix} K_f & \tilde{K}_f \\ \tilde{K}_f^* & K_f^* \end{bmatrix} \tag{9.1}$$

where K_f, $\tilde{K}_f$ are the Hermitian and complementary covariances of f. We use the augmented form in some proofs in the following section. Similarly, the second-order statistics of a complex process can be expressed by the *augmented covariance function*

$$\underline{k}(z,w) = \mathbb{E}\left[\begin{bmatrix} f(z) \\ f^*(z) \end{bmatrix} \begin{bmatrix} f^*(w) & f(w) \end{bmatrix}\right] = \begin{bmatrix} k(z,w) & \tilde{k}(z,w) \\ \tilde{k}^*(z,w) & k^*(z,w) \end{bmatrix} \tag{9.2}$$

Generally, an underline denotes an augmented representation, either of a process or of a covariance function or matrix.

H_∞ Gaussian Processes

Consider a deterministic input-output operator H_g with transfer function function $g : D \to \mathbb{C}$. The condition that H_g belong to the operator space H^∞ of LTI, causal, and BIBO stable systems is that g belong to the function space H_∞. Now suppose we wish to construct a random operator H_f using the realizations of a z-domain process f as its transfer function: the analogous condition is that the realizations of f lie in H_∞ with probability one.

Definition 3. *A z-domain process is called an H_∞ process when the set $\{\xi \in \Xi : f_\xi \in H_\infty\}$ has measure one under $\mathbb{P}$.*

[1] Other names for the complementary (co)variance in the literature of complex random variables and stochastic processes are the *pseudo(co)variance* and *relation*.

Less formally, an H_∞ process is a z-domain process f such that $\mathbb{P}(f \in H_\infty) = 1$. Having $f_\xi \in H_\infty$ implies that $\bar{E} \subseteq D$: we usually take $D = \bar{E}$. If we also require that H_g give real outputs to real inputs in the time domain, g must satisfy the conjugate symmetry relation $g(z^*) = g^*(z)$ for all $z \in D$. The analogous condition for H_f is to require that f satisfy the condition with probability one.

Definition 4. *A z-domain process f is called conjugate symmetric when the set $\{\xi \in \Xi : f_\xi(z^*) = f_\xi^*(z),\ \forall z \in D\}$ has measure one under $\mathbb{P}$.*

Combining definitions 2, 3, and 4, we arrive at our main object of study: conjugate-symmetric H_∞ Gaussian processes.

Example 2 ("Cozine" process)**.** *The random transfer function*

$$f(z) = \frac{X - a(X\cos(\omega_0) - Y\sin(\omega_0))z^{-1}}{1 - 2a\cos(\omega_0)z^{-1} + a^2z^{-2}}, \tag{9.3}$$

where $X, Y \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$, $a \in (0,1)$, $\omega_0 \in [0,\pi]$, is a z-domain Gaussian process. From the form of the transfer function, we see that H is bounded on the unit circle, analytic on E, and conjugate symmetric with probability one, from which it follows that f is a conjugate symmetric H_∞ process. Since f corresponds to the z-transform of an exponentially decaying discrete cosine with random magnitude and phase, we call it a "cozine" process*. Its Hermitian and complementary covariances are*

$$\begin{aligned} k(z,w) &= \frac{1 - a\cos(\omega_0)(z^{-1} + (w^*)^{-1}) + a^2(zw^*)^{-1}}{(1 - 2a\cos(\omega_0)z^{-1} + a^2z^{-2})(1 - 2a\cos(\omega_0)(w^*)^{-1} + a^2(w^*)^{-2})}, \\ \tilde{k}(z,w) &= \frac{1 - a\cos(\omega_0)(z^{-1} + w^{-1}) + a^2(zw)^{-1}}{(1 - 2a\cos(\omega_0)z^{-1} + a^2z^{-2})(1 - 2a\cos(\omega_0)w^{-1} + a^2w^{-2})}. \end{aligned} \tag{9.4}$$

As a Bayesian prior for an H^∞ system, this process represents a belief that the transfer function exhibits a resonance peak (of unknown magnitude) at ω_0. Knowing ω_0 in advance is a strong belief, but it can be relaxed by taking a hierarchical model where ω_0 enters as a hyperparameter. When used as a prior, the hierarchical model represents the less determinate belief that there is a resonance peak *somewhere*, whose magnitude can be made arbitrarily small if no peak is evident in the data.

The construction in Example 2, where properties of conjugate symmetry and BIBO stability can be checked directly, may be extended to random transfer functions of any finite order. However, the technique does not carry to the infinite-order H_∞ processes required for nonparametric Bayesian system identification, or more generally for applications that do not place an *a priori* restriction on the order of the system. We are therefore motivated to find conditions under which a z-domain process is a conjugate-symmetric H_∞ Gaussian process expressed directly in terms of k and $\tilde{k}$.

9.2 Constructing H_∞ Gaussian Processes

In this section, we consider a z-domain Gaussian process f with zero mean, Hermitian covariance function k, and complementary covariance function $\tilde{k}$. Taking zero mean implies no loss in generality: to lift any of these conditions to a process with nonzero mean, we simply ask that the desired property (inhabiting H_∞, possessing conjugate symmetry, or both) also hold for the mean.

Our first step towards finding conditions under which a z-domain process is a conjugate-symmetric H_∞ process is the observation that all functions in H_∞ are also in H_2: $\|f\|_\infty < \infty$ implies that $\int_{-\pi}^{\pi} |f(Re^{j\Omega})|^2 d\Omega$ converges for all $R \geq 1$. Indeed, $f \in H_\infty$ precisely when $f \in H_2$ and $\|f\|_\infty < \infty$. Our general strategy for proving that a z-domain process is an H_∞ process f is to show that $f_\xi \in H_2$ and $\|f_\xi\|_\infty < \infty$ hold for realizations f_ξ of the process with probability one.

To show when $f \in H_2$ with probability one, we use the fact that H_2 is a reproducing kernel Hilbert space.

Definition 5. *A reproducing kernel Hilbert space (RKHS) is a Hilbert space of functions on a domain D for which the* evaluation functionals *E_z, defined pointwise as $E_z f = f(z)$, are bounded in the sense that $E_z f \leq M(z)\|f\|$ for some nonnegative function $M(z)$.*

Applying the Riesz representation theorem to the evaluation functionals, which are linear and by assumption bounded, we recover the *reproducing kernel* k that satisfies $\langle k(z,\cdot), f\rangle = f(z)$ and that $k(z,z)$ is the least $M(z)$ such that $E_z f \leq M(z)\|f\|$ holds. The form of k can be derived from an orthonormal basis for the RKHS using the following result.

Lemma 8 ([68], Theorem 2.4)**.** *Let H be an RKHS with reproducing kernel k. If $e_0, e_1, \ldots$ form an orthonormal basis for H, then $k(z,w) = \sum_{n=0}^{\infty} e_n(z)e_n^*(w)$ where the series converges pointwise.*

Discrete-time H_2 is a type of *Hardy class*, which is a space of complex functions that are analytic on a domain of the complex plane and satisfy a bounded-growth condition on the boundary. When this domain is a half-plane or the interior of the unit disk, it is well known that these spaces are RKHSs. It is therefore not surprising that the same is true when the domain is the exterior of the unit disk, as it is for our H_2. However, we are not aware of a citable proof of this fact, nor of formula for its kernel, so we provide both here for completeness.

Proposition 5. *Discrete-time H_2 is a reproducing kernel Hilbert space with kernel function*

$$k(z,w) = \frac{zw^*}{zw^* - 1}$$

and orthonormal basis $\{e_n\}_{n=0}^{\infty}$, $e_n(z) = z^{-n}$.

Proof. The first step is to compute the orthonormal basis. We begin with the standard fact (see [94, theorem 13.3]) that $e_n(z) = z^{-n}, n \in \mathbb{Z}$ form an orthonormal basis for L_2. Since H_2 is a subspace of L_2, we can take the orthogonal projection of $z^{-n}, n \in \mathbb{Z}$ onto H_2 to yield a sequence that spans H_2. For $n < 0$, z^{-n} is unbounded on the exterior of the unit disk, so $P(e_n) = 0$ for $n < 0$. On the other hand, $z^{-n} \in H_2$ for $n \geq 0$, so $P(e_n) = e_n$ for $n \geq 0$. Discarding the zeros, we have that $H_2 = \text{span}(\{e_n\}_{n=0}^\infty)$. Since L_2 and H_2 have the same norm, we already know that e_n, $n \geq 0$ are orthonormal in H_2. The combined facts of orthonormality and spanning the space ensure (e.g. by [49, chapter 2, §8, Theorem 3]) that $e_n, n \geq 0$ form an orthonormal basis for H_2.

Now we establish that the evaluation functionals $E_z(f) = f(z)$ are bounded. Let $f \in H_2$: from the paragraph above, we expand f as $f(z) = \sum_{n=0}^\infty a_n z^{-n}$. By the Parseval identity $\|f\|_2 = \sum_{n=0}^\infty |a_n|^2$, we have

$$|f(z)| = \left|\sum_{n=0}^\infty a_n z^{-n}\right| \leq \sum_{n=0}^\infty |a_n||z|^{-n} \leq \sqrt{\sum_{n=0}^\infty |a_n|^2}\sqrt{\sum_{n=0}^\infty (|z|^{-2})^n} = \|f\|_2 \frac{1}{\sqrt{1-|z|^{-2}}}. \quad (9.5)$$

Since $E_z(f) \leq M(z)\|f\|_2$ for $M(z) = 1/\sqrt{1-|z|^{-2}}$, it follows that H_2 is an RKHS. This allows us to apply Lemma 8 to compute

$$k(z,w) = \sum_{n=0}^\infty e_n(z)e_n^*(w) = \sum_{n=0}^\infty z^{-n}(w^*)^{-n} = \frac{1}{1-(zw^*)^{-1}} = \frac{zw^*}{zw^*-1}. \quad (9.6)$$

□

The fact that H_2 is an RKHS allows us to use Driscoll's zero-one theorem to establish if the realizations of a z-domain Gaussian process belong to H_2.

Lemma 9 ([41])**.** *Let f be a mean zero Gaussian process on a parameter set T with covariance function k. Let r be the reproducing kernel of an RKHS of functions with domain T. Let $t_1, t_2, \ldots$ denote a countably dense set of points in T, and define $K^n, R^n \in \mathbb{R}^{n\times n}$ as $(K_n)_{ij} = k(t_i, t_j)$, $(R_n)_{ij} = r(t_i, t_j)$. Then the realizations of f are in the RKHS with kernel k with probability either zero or one, according respectively to whether $\sum_n \text{trace}\, K_n R_n^{-1}$ is infinite or finite.*

To ensure that $\|f\|_\infty < \infty$, we need a sufficient condition under which the realizations of f are bounded on the unit circle. The following result provides a sufficient condition in terms of the continuity of the covariance.

Lemma 10 ([2], Theorem 1.4.1)**.** *Let f be a real-valued Gaussian process with mean zero defined on a compact parameter set $T \subseteq R^n$. If there exist positive constants C, α, and δ such that the covariance function k satisfies*

$$k(s,t) = k(s,s) + k(t,t) - 2k(s,t) \leq \frac{C}{|\log|\theta-\phi||^{1+\alpha}} \quad (9.7)$$

for $s, t \in T$ such that $|s - t| < \delta$, then

$$P\left(\sup_{t\in T} |f(t)| < \infty\right) = 1. \tag{9.8}$$

We are now prepared to return to H_∞ Gaussian processes. The following result provides the general test to determine if f is an H_∞ Gaussian process, by establishing with probability one that $f_\xi \in H_2$ and $\|f_\xi\|_\infty < \infty$.

Theorem 6. *Let f be a z-domain Gaussian process with mean zero and continuous Hermitian covariance k and complementary covariance $\tilde{k}$. Let $k_r = \frac{1}{2}\operatorname{Re}[k + \tilde{k}]$, $k_i = \frac{1}{2}\operatorname{Re}[k - \tilde{k}]$ denote the covariance functions of the real and imaginary parts of f respectively. Then f is an H_∞ process under the following conditions:*

1. *There exist positive, finite constants C_r, C_i, α_r, α_i, δ_r, δ_i, such that k_r and k_i, restricted to the unit circle, satisfy the following continuity conditions:*

$$\begin{aligned} k_r(e^{j\theta}, e^{j\theta}) + k_r(e^{j\phi}, e^{j\phi}) - 2k_r(e^{j\theta}, e^{j\phi}) &\le \frac{C_r}{|\log|\theta - \phi||^{1+\alpha_r}} \quad \forall |\theta - \phi| < \delta_r \\ k_i(e^{j\theta}, e^{j\theta}) + k_i(e^{j\phi}, e^{j\phi}) - 2k_i(e^{j\theta}, e^{j\phi}) &\le \frac{C_i}{|\log|\theta - \phi||^{1+\alpha_i}} \quad \forall |\theta - \phi| < \delta_i. \end{aligned} \tag{9.9}$$

2. *Let $\{z_n\}_{n=1}^\infty$ be a countable dense sequence of points in E. For $n \in \mathbb{N}$, define the Gramian matrices $K_r^n, K_i^n, R^n \in \mathbb{R}^{n\times n}$ as $(K_r^n)_{jl} = k_r(z_j, z_l)$, $(K_i^n)_{jl} = k_i(z_j, z_l)$, and $(R^n)_{jl} = r(z_j, z_l)$, where $r(z_j, z_l) = z_i z_j^*/(z_i z_j^* - 1)$. K_r^n, K_i^n, and R^n satisfy*

$$\sup_{n\in\mathbb{N}} \operatorname{trace} K_r^n (R^n)^{-1} < \infty \quad \text{and} \quad \sup_{n\in\mathbb{N}} \operatorname{trace} K_i^n (R^n)^{-1} < \infty. \tag{9.10}$$

Proof. To apply Lemmas 9 and 10, we work separately with the real and imaginary parts of the process. To that end, we write $f = x + jy$, where x and y are real Gaussian processes with covariance functions k_r and k_i.

First, suppose that both conditions hold. Since r is the reproducing kernel of H_2 by Proposition 5, condition (9.10) ensures by Lemma 9 that the sample paths of x and y lie in H_2 with probability one, ensuring the same for f. Since k_r and k_i satisfy the hypotheses of Lemma 10, it follows that x and y are bounded with probability one, which implies the same for f. □

Remark 5. *According to Driscoll's theorem, the probability that $f \in H_2$ is either zero or one. (Zero occurs when either supremum in condition* (9.10) *is infinite.) Similarly, the realizations of a Gaussian process are bounded with probability zero or one ([58]). This means that the realizations of a z-domain Gaussian process are either almost surely H_∞ functions or almost surely not: there are no "part-time H_∞" Gaussian processes.*

Remark 6. *Condition* (9.10) *is necessary and sufficient for f to inhabit H_2 with probability one. On the other hand, condition* (9.9) *is sufficient but not necessary for $\|f\|_\infty$ to be bounded. Indeed, necessary and sufficient conditions for a stochastic process to be almost surely bounded are generally not available even for real-valued Gaussian processes except in special cases. Fortunately, covariance functions in practice often satisfy a stronger condition that implies* (9.9) *([1, eq. 2.5.17]), namely that $k(s,t) = k(s,s) - q(s-t) + O(|s-t|^{2+\delta})$, for small $|s-t|$, where q is a positive definite quadratic form and $\delta > 0$.*

The general condition for a process to be conjugate symmetric is given by the following result.

Proposition 6. *Let f be a z-domain Gaussian process with domain D, covariance k, and complementary covariance $\tilde{k}$. Then f is conjugate-symmetric if and only if k and $\tilde{k}$ satisfy the conditions*

$$k(z,z) = k(z^*,z^*), \qquad k(z,z) = \tilde{k}(z,z^*) \tag{9.11}$$

for all $z \in D$.

Proof. For a fixed $z \in D$, consider the random vector $(f^*(z), f(z^*))$. This is a multivariate complex normal whose augmented covariance matrix is

$$\mathbb{E}\left[\begin{bmatrix} f^*(z) \\ f(z^*) \\ f(z) \\ f^*(z^*) \end{bmatrix} \begin{bmatrix} f^*(z) & f(z^*) & f(z) & f^*(z^*) \end{bmatrix}\right] = \begin{bmatrix} k^*(z,z) & \tilde{k}^*(z,z^*) & \tilde{k}^*(z,z) & k^*(z,z^*) \\ \tilde{k}(z^*,z) & k(z^*,z^*) & k(z^*,z) & \tilde{k}(z^*,z^*) \\ \tilde{k}(z,z) & k(z,z^*) & k(z,z) & \tilde{k}(z,z^*) \\ k^*(z^*,z) & \tilde{k}^*(z^*,z^*) & \tilde{k}^*(z^*,z) & k^*(z^*,z^*) \end{bmatrix}. \tag{9.12}$$

Since augmented covariance matrices have the form

$$\begin{bmatrix} K & \tilde{K} \\ \tilde{K}^* & K^* \end{bmatrix} \tag{9.13}$$

where K and $\tilde{K}$ are the Hermitian and complementary covariances, the Hermitian and complementary covariances of $(f^*(z), f(z^*))$ are

$$K = \begin{bmatrix} k^*(z,z) & \tilde{k}^*(z,z^*) \\ \tilde{k}(z^*,z) & k(z^*,z^*) \end{bmatrix}, \qquad \tilde{K} = \begin{bmatrix} \tilde{k}^*(z,z) & k^*(z,z^*) \\ k(z^*,z) & \tilde{k}(z^*,z^*) \end{bmatrix}. \tag{9.14}$$

Since $(f^*(z), f(z^*))$ is Gaussian with mean zero, this means that

$$\begin{bmatrix} f^*(z) \\ f(z^*) \end{bmatrix} \sim \mathcal{CN}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} k^*(z,z) & \tilde{k}^*(z,z^*) \\ \tilde{k}(z^*,z) & k(z^*,z^*) \end{bmatrix}, \begin{bmatrix} \tilde{k}^*(z,z) & k^*(z,z^*) \\ k(z^*,z) & \tilde{k}(z^*,z^*) \end{bmatrix}\right). \tag{9.15}$$

Under the conditions given on k and $\tilde{k}$, this reduces to

$$\begin{bmatrix} f^*(z) \\ f(z^*) \end{bmatrix} \sim \mathcal{CN}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, k(z,z)\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, \tilde{k}(z,z)\begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}\right), \tag{9.16}$$

from which it follows that

$$f^*(z) - f(z^*) = \begin{bmatrix} 1 & -1 \end{bmatrix} \begin{bmatrix} f^*(z) \\ f(z^*) \end{bmatrix} \sim \mathcal{CN}(0, 0, 0). \tag{9.17}$$

This means that $f^*(z) - f(z^*) = 0$, or equivalently $f^*(z) = f(z^*)$, with probability one, for all $z \in D$. On the other hand, if the conditions in (9.11) are not met, then for at least one $z \in D$, the reduction from (9.15) to (9.16) is not possible, in which case $f(z^*) = f^*(z)$ does not hold. □

Together, Theorem 6 and Proposition 6 give sufficient conditions on the covariance functions of a general mean-zero z-domain Gaussian process in order for it to be a conjugate-symmetric H_∞ Gaussian process. While Conditions (9.9) and (9.11) can be verified in practice, Condition (9.10) generally cannot. We are therefore motivated to find special cases of z-domain Gaussian processes for which (9.10) can be replaced by a more tractable condition. The broadest such case that we have found is where, in addition to satisfying Conditions (9.9) and (9.11), the Hermitian covariance function is stationary when restricted to the unit circle.

Definition 6. *A z-domain Gaussian process is* Hermitian stationary *when its Hermitian covariance function satisfies* $k(e^{j\theta}, e^{j\phi}) = k(e^{j(\theta-\phi)}, 1)$ *for all* $\theta, \phi \in [-\pi, \pi)$.

Using a stationary process as a prior is common practice in machine learning and control-theoretic applications of Gaussian process models. Stationary processes are useful for constructing regression priors that do not introduce unintended biases in their belief about the frequency response: since $f(e^{j\theta})$ has the same Hermitian variance across the entire unit circle, a sample path from a Hermitian stationary H_∞ process is just as likely to exhibit low-pass behavior as it is high-pass or band-pass.[2] We can obtain a "partially informative" prior by adding an H_∞ process encoding strong beliefs in one frequency range (such as the presence of a resonance peak) to an H_∞ process encoding weaker beliefs across all frequencies. The sum, also an H_∞ process, encodes a combination of these beliefs.

Under the additional condition of Hermitian stationarity, it turns out that the H_∞ process is characterized by a sequence of nonnegative constants.

Theorem 7. *Let f be a Hermitian stationary, conjugate-symmetric z-domain Gaussian process with continuous Hermitian covariance k and complementary covariance $\tilde{k}$. Then f is an H_∞ process if and only if k and $\tilde{k}$ have the form*

$$k(z, w) = \sum_{n=0}^{\infty} a_n^2 (zw^*)^{-n}, \qquad \tilde{k}(z, w) = \sum_{n=0}^{\infty} a_n^2 (zw)^{-n}, \tag{9.18}$$

where $\{a_n\}_{n=0}^{\infty}$ is a nonnegative real ℓ^1 sequence. Furthermore, f may be expanded as

$$f(z) = \sum_{n=0}^{\infty} a_n w_n z^{-n}, \tag{9.19}$$

[2] To be truly "noninformative" in the sense of zero correlation, the complementary covariance should be stationary. However, this is not possible while satisfying (9.11).

where $w_n \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$.

Proof. First, we show that f having the form (9.19) with positive $\{a_n\}_{n=0}^{\infty} \in \ell^1$ implies that f is a Hermitian stationary H_∞ process satisfying (9.11) with the given covariances. Suppose that $f(z) = \sum_{n=0}^{\infty} a_n w_n z^{-n}$. We can readily see that

$$\begin{aligned} k(z,w) = \mathbb{E}\left[f(z)f^*(w)\right] &= \mathbb{E}\left[\left(\sum_{n=0}^{\infty} a_n w_n z^{-n}\right)\left(\sum_{m=0}^{\infty} a_m w_m (w)^{-m}\right)^*\right] \\ &= \mathbb{E}\left[\sum_{n=0,m=0}^{\infty} a_n a_m w_n w_m z^{-n}(w^*)^{-n}\right] = \sum_{n=0}^{\infty} a_n^2 (zw^*)^{-n}, \end{aligned} \tag{9.20}$$

where the cross terms vanish by the independence of the w_n. Note also that $k(z,z) = k(z^*,z^*)$, which is the first part of condition (9.11). A similar calculation yields

$$\tilde{k}(z,w) = \mathbb{E}\left[f(z)f(w)\right] = \sum_{n=0}^{\infty} a_n^2 (zw)^{-n} = k(z,w^*), \tag{9.21}$$

showing that both parts of condition (9.11) are satisfied, and that f has real impulse response $h_f(n) = w_n a_n$. Recall that a SISO system is BIBO stable if its impulse response is absolutely summable. To that end, consider the sequence

$$M_T = \sum_{n=0}^{T} |h_f(n)| = \sum_{n=0}^{T} a_n |w_n| \tag{9.22}$$

of partial sums: if $\lim_{T\to\infty} M_T$ converges to a random variable that is finite with probability one, then the impulse response is absolutely summable with probability one. Since the w_n are independent and $0 < a_n|w_n| < \infty$ for all n, it follows that M_T is a submartingale and that $\mathbb{E}\left[M_T\right]$ increases monotonically. Using the summability condition on the a_n and the fact that $\mathbb{E}\left[|w_n|\right] = \sqrt{2/\pi}$ (as $|w_n|$ follows a half-normal distribution), we have

$$\mathbb{E}\left[\sum_{n=0}^{\infty} |h_f(n)|\right] = \sum_{n=0}^{\infty} a_n |w_n| = \sqrt{\frac{2}{\pi}} \sum_{n=0}^{\infty} a_n < \infty, \tag{9.23}$$

which means $\sup_T \mathbb{E}\left[M_T\right] < \infty$ by monotonicity. Since M_T is a submartingale and $\sup_T \mathbb{E}\left[M_T\right]$ is finite, it follows by the Martingale convergence theorem [43, Theorem 4.2.11] that the limit of M_T converges to a random variable that is finite with probability one. This shows that h_f is absolutely summable with probability one, implying BIBO stability and that $f \in H_\infty$ with probability one.

Next, we show that a Hermitian stationary, conjugate-symmetric H_∞ Gaussian process f must have Hermitian and complementary covariances of the form (9.18), and that this in

turn implies that f has the form (9.19). Since $f \in H_\infty$, we can use the fact that $z^{-n}, n \geq 0$ is a basis for H_∞ to expand f as

$$f(z) = \sum_{n=0}^{\infty} h_n z^{-n}, \tag{9.24}$$

where the coefficients $h_n = \langle f, z^{-n} \rangle_2$ are an infinite sequence of random variables. Since f is Gaussian and conjugate symmetric, the h_n are real Gaussian random variables that may be correlated. From this form, we can express the Hermitian and complementary covariance as

$$\begin{aligned} k(z,w) &= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathbb{E}\left[h_n h_m\right] z^{-n}(w^*)^{-m} \\ \tilde{k}(z,w) &= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathbb{E}\left[h_n h_m\right] z^{-n}(w)^{-m}, \end{aligned} \tag{9.25}$$

which shows that $\tilde{k}(z,w) = k(z,w^*)$ for $z, w \in E^2$. Restricting the covariance functions to the unit circle, we have

$$\begin{aligned} k(e^{j\theta}, e^{j\phi}) &= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathbb{E}\left[h_n h_m\right] e^{-j(n\theta - m\phi)} \\ \tilde{k}(e^{j\theta}, e^{j\phi}) &= \sum_{n=0}^{\infty}\sum_{m=0}^{\infty} \mathbb{E}\left[h_n h_m\right] e^{-j(n\theta + m\phi)}. \end{aligned} \tag{9.26}$$

By the assumption of Hermitian stationarity, we know that $k(e^{j(\theta-\phi)}, 1)$ is a positive definite function whose domain is the unit circle. We can therefore apply Bochner's theorem [75, section 1.4.3] to obtain a second expansion

$$k(e^{j(\theta-\phi)}, 1) = k(e^{j\theta}, e^{j\phi}) = \sum_{n\in\mathbb{Z}} a_n^2 e^{-jn(\theta-\phi)}, \tag{9.27}$$

where a_n are real and nonnegative. In order for the expansion of k in (9.26) and the expansion in (9.27) to be equal, the positive-power terms in (9.27) must vanish, and the cross-terms in (9.26) must vanish.

This means that $\mathbb{E}\left[h_n h_m\right] = 0$ for $m \neq n$, from which it follows that the covariances have the form

$$\begin{aligned} k(z,w) &= \sum_{n=0}^{\infty} \mathbb{E}\left[h_n^2\right] (zw^*)^{-n} = \sum_{n=0}^{\infty} a_n^2 (zw^*)^{-n} \\ \tilde{k}(z,w) &= \sum_{n=0}^{\infty} \mathbb{E}\left[h_n^2\right] (zw)^{-n} = \sum_{n=0}^{\infty} a_n^2 (zw)^{-n} \end{aligned} \tag{9.28}$$

where we identify $a_n^2 = \mathbb{E}[h_n^2]$, and that the h_n are independent. Returning to the expanded form of the process and expressing $\mathbb{E}[h_n^2] = a_n^2$, we have

$$f(z) = \sum_{n=0}^{\infty} w_n a_n z^{-n} \tag{9.29}$$

where $w_n \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1)$.

Evidently, the impulse response h_f has the same form as before, so the expected absolute sum of the impulse response is $\mathbb{E}\left[\sum_{n=0}^{\infty} |h_f(n)|\right] = \sqrt{\frac{2}{\pi}} \sum_{n=0}^{\infty} a_n$. Since $f \in H_\infty$ by assumption, it follows that $\sum_{n=0}^{\infty} |h_f(n)|$ almost surely converges, and therefore that $\mathbb{E}\left[\sum_{n=0}^{\infty} |h_f(n)|\right] < \infty$ by the Kolmogorov three-series theorem ([43, Theorem 2.5.8], condition (ii)), showing that $\{a_n\}_{n=0}^{\infty} \in \ell^1$.

□

Theorem 7 provides a useful tool for constructing conjugate-symmetric H_∞ Gaussian processes: all we need to do is select a summable sequence of nonnegative numbers.

Example 3 (Geometric H_∞ process)**.** *Take $a_n^2 = \alpha^n$ with $\alpha \in (0,1)$; this yields a conjugate-symmetric H_∞ Gaussian process with Hermitian covariance $k_\alpha(z,w) = \sum_{n=0}^{\infty} \alpha^n (zw^*)^{-n} = \frac{zw^*}{zw^* - \alpha}$ and complementary covariance $\tilde{k}_\alpha(z,w) = \frac{zw}{zw-\alpha}$.*

Example 4 (Exponential H_∞ process)**.** *Take $a_n^2 = \frac{1}{n!}$; this yields a conjugate-symmetric H_∞ Gaussian process with Hermitian covariance $k(z,w) = \sum_{n=0}^{\infty} \frac{(zw^*)^{-n}}{n!} = e^{-zw^*}$ and complementary covariance $\tilde{k}(z,w) = e^{-zw}$.*

9.3 Gaussian Process Regression in the Frequency Domain

Let $H_\Delta \in H^\infty$ denote a system uncertainty whose transfer function $\Delta \in H_\infty$ we wish to identify. While not necessarily stochastic, Δ is unknown, and we represent both our uncertainty and our prior beliefs in a Bayesian fashion with an H_∞ Gaussian process with Hermitian and complementary covariances k and $\tilde{k}$. To model our prior beliefs, the distribution of Δ should give greater probability to functions we believe are likely to correspond to the truth, and should assign probability zero to functions ruled out by our prior beliefs. As an example of the latter, the fact that $P(\Delta \in H_\infty) = 1$ encodes our belief that $\Delta \in H_\infty$, which demonstrates the importance of H_∞ Gaussian processes for prior model design.

We suppose that our data consists of n noisy frequency-domain point estimates $y_i = \Delta(z_i) + e_i$, where $e_i \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_n^2)$, $z_i \in \bar{E}$. If our primary form of data is a time-domain trace of input and output values, we first convert this data into an *empirical transfer function estimate* (ETFE). There are several well-established methods to construct ETFEs from time

traces, such as Blackman-Tukey spectral analysis, windowed filter banks, or simply dividing the DFT of the output trace by the DFT of the input trace. In our numerical examples, we will use windowed filter banks.

Our approach is essentially the same procedure as standard Gaussian process regression as described in [73] extended to the complex case. We take the mean of the prior model to be zero without loss of generality. To estimate the transfer function at a new point z, we note that $\Delta(z)$ is related to $(y_1, \ldots, y_n)$ under the prior model as

$$\begin{bmatrix} \Delta(z) \\ y \end{bmatrix} \sim \mathcal{CN}\left(0, \begin{bmatrix} K_{xx} & K_{xy} \\ K_{xy}^H & K_{yy} \end{bmatrix}, \begin{bmatrix} \tilde{K}_{xx} & \tilde{K}_{xy} \\ \tilde{K}_{xy}^H & \tilde{K}_{yy} \end{bmatrix}\right); \tag{9.30}$$

where $y \in \mathbb{C}^n$, $K_{yy} \in \mathbb{C}^{n\times n}$, $K_{xy} \in \mathbb{C}^{n\times 1}$, and $K_{xx} \in \mathbb{C}$ are defined componentwise as

$$(y)_i = y_i, \quad (K_{yy})_{ij} = k(z_i, z_j) + \sigma_n^2 \delta_{ij}, \quad (K_{xy})_{ij} = k(z, z_i), \quad K_{xx} = k(z, z) + \sigma_n^2, \tag{9.31}$$

and the components of the complementary covariance matrix are defined analogously.

By conditioning $\Delta(z)$ on the data y according to the prior model, we obtain the posterior distribution of $\Delta(z)$. According to the conditioning law for multivariate complex Gaussian random variables [76, §2.3.2], this is $\Delta(z)|y \sim \mathcal{CN}\left(\mu, \sigma_p^2, \tilde{\sigma}_p^2\right)$, where

$$\begin{aligned}
\mu_q &= (K_{xy} - \tilde{K}_{xy}(K_{yy}^*)^{-1}\tilde{K}_{yy}^*)P^{-1}y + (\tilde{K}_{xy} - K_{xy}K_{yy}^{-1}\tilde{K}_{yy})(P^*)^{-1}y^* \\
\sigma_q^2 &= k_{zz} - K_{xy}P^{-1}K_{xy}^H + \tilde{K}_{xy}K_{yy}^{-1}\tilde{K}_{yy}(P^*)^{-1}K_{xy}^H \\
&\quad - \tilde{K}_{xy}(P^*)^{-1}\tilde{K}_{xy}^H + K_{xy}K_{yy}^{-1}\tilde{K}_{yy}(P^*)^{-1}\tilde{K}_{xy}^H \\
\tilde{\sigma}_p^2 &= \tilde{k}_{zz} - K_{xy}P^{-1}(\tilde{K}_{xy}^*)^H + \tilde{K}_{xy}K_{yy}^{-1}\tilde{K}_{yy}(P^*)^{-1}(\tilde{K}_{xy}^*)^H \\
&\quad - \tilde{K}_{xy}(P^*)^{-1}(K_{xy}^*)^H + K_{xy}K_{yy}^{-1}\tilde{K}_{yy}(P^*)^{-1}(K_{xy}^*)^H
\end{aligned} \tag{9.32}$$

and where P denotes the Schur complement $P = K_{yy} - \tilde{K}_{yy}(K_{yy}^*)^{-1}\tilde{K}_{yy}^*$. The predictive mean μ_p is the *minimum mean-square error widely linear estimator* of $\Delta(z)$ given y, where "widely linear" means that μ_p is a linear combination of both y and y^*. A *strictly linear* estimator, on the other hand, uses only y. Under the same circumstances as above, the minimum least-square strictly linear estimator for $\Delta(z)$ given y and its error variance are respectively

$$\hat{\Delta}(z) = K_{xy}^H K_{yy}^{-1} y, \qquad \sigma_\Delta^2(z) = K_{zz} - K_{xy}^H K_{yy}^{-1} k_{xy}, \tag{9.33}$$

which are identical to the posterior mean and variance of a real Gaussian process regression model (cf. Equation (2.19) in [73]) except that K_{xx}, K_{xy}, and K_{yy} are complex-valued.

The widely linear estimator can only be an improvement on the linear estimator, since an estimate made using y can certainly be made using (y, y^*). The improvement is measured by the Schur complement P defined above, which is the error covariance of statistically estimating y^* from y, or equivalently estimating the real part given the imaginary part. In particular, when $P = 0$, the strictly linear and widely linear estimators coincide, and the expressions in (9.32) become ill-defined. One case where this holds is when the covariances

are *maximally improper*, in which case the imaginary part can be estimated from the real with zero error.

In our experiments with real-impulse H_∞ processes, we have found that P tends to be close to singular, and small in induced 2-norm and Frobenius norm relative to K_{yy} and $\tilde{K}_{yy}$. This makes the mean and variance computations in (9.32) numerically unstable while also implying that the strictly linear estimator will perform similarly to the widely linear estimator. We believe this is due to the symmetry condition imposed on k and $\tilde{k}$ by having real impulse response. This condition implies that the imaginary part can be computed exactly from the real part by the discrete Hilbert transform [72, §2.26]. The covariance matrices K_{yy} and $\tilde{K}_{yy}$ will not themselves be maximally improper, since the Hilbert transform requires knowledge over the entire unit circle; however, our experiments suggest that they are close to maximally improper, and we conjecture that they become maximally improper in the limit of infinite data. This suggests that the strictly linear estimator will perform well for conjugate-symmetric H_∞ priors. For this reason, as well as the numerical instability of the widely linear estimator when P is close to singular, we use the strictly linear estimator in our numerical experiments.

For $z \in D$ and $\eta > 0$, define the *confidence ellipsoid* $\mathcal{E}_\eta(z) = \{w \in \mathbb{C} : |w - \hat{\Delta}(z)|^2 \leq \eta^2 \sigma_\Delta^2(z)\}$. By Markov's inequality, we know that $\Delta(z) \in \mathcal{E}_\eta(z)$ with probability $\geq 1 - 1/\eta^2$. This implies bounds on the real and imaginary parts by projecting the confidence ellipsoid onto the real and imaginary axes: from these we can construct probabilistic bounds on the magnitude and phase of $\Delta(z)$ via interval arithmetic, which we will see in the numerical examples.

Let $\theta \in \Theta$ denote the hyperparameters of a covariance function k_θ, so that K_{yy} becomes a function of θ: then the log marginal likelihood of the data under the posterior for the strictly linear estimator is $L(\theta) = -\frac{1}{2}\left(y^H K_{yy}(\theta)^{-1} y + \log\det K_{yy}(\theta) + n\log 2\pi\right)$. Keeping the data y and input locations z_i fixed, $L(\theta)$ measures the probability of observing data y when the prior covariance function is k_θ. By maximizing L with respect to θ, we find the covariance among k_θ, $\theta \in \Theta$ that best explains the observations.[3]

To summarize, the regression process is as follows:

1. Select a family of H_∞ Gaussian process models indexed by a hyperparameter set Θ;
2. observe point estimate data, typically by an empirical transfer function estimate;
3. Select $\theta \in \Theta$ that maximizes the log likelihood $L(\theta)$;
4. Use the strictly linear estimator (9.33) to obtain an estimate $\hat{\Delta}$ and predictive variance σ_Δ.

We now demonstrate the process by identifying two second-order systems.

[3] Although it seems contradictory to choose prior parameters based on posterior data, it can be justified as an empirical-Bayes approximation to a hierarchical model with θ as hyperparameter.

Examples: Identifying Second-order Systems

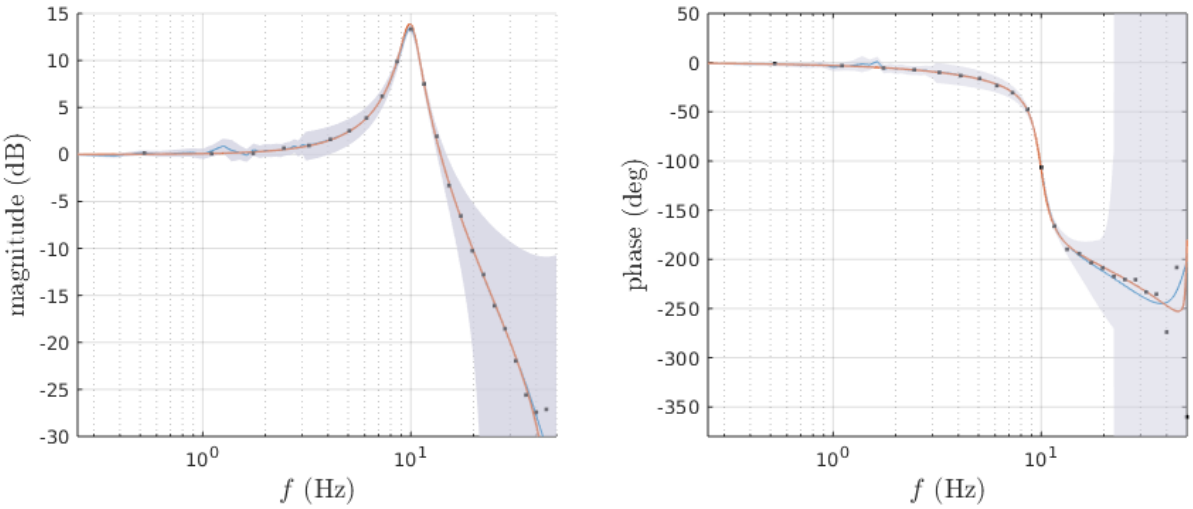

Figure 9.1: Bode plot of the second-order resonant system (orange), and its estimate (blue) using H_∞ Gaussian process regression from an empirical transfer function estimate (black points) with $\eta = 3$ confidence ellipsoid bounds (grey).

We apply the strictly-linear H_∞ Gaussian process regression method described above to the problem of identifying two second-order systems. The first test system is a second-order system that exhibits a resonance peak. The system is specified in continuous time, with canonical second-order transfer function

$$g(s) = \frac{\omega_0^2}{s^2 + 2\xi\omega_0 s + \omega_0^2}, \tag{9.34}$$

where $\omega_0 = 20\pi$ rad/s, and $\xi = 0.1$, and converted to the discrete-time transfer function $g(z)$ using a zero-order hold discretization with a sampling frequency of $f_s = 100$ Hz. We suppose that we know *a priori* that there is a resonance peak, but not about its location or half-width, and we have no other strong information about the frequency response. For this prior belief, an appropriate prior model is a weighted mixture of a cozine process and a Hermitian stationary process. In particular, we use the family of H_∞ processes with covariance functions

$$k(z,w) = \sigma_g^2 k_g(z,w) + \sigma_c^2 k_c(z,w), \tilde{k}(z,w) = \sigma_g^2 \tilde{k}_g(z,w) + \sigma_c^2 \tilde{k}_c(z,w), \tag{9.35}$$

where k_g is the covariance of the geometric H_∞ process defined in Example 3, and k_c is the covariance of the cozine process, and likewise for the complementary covariance. σ_g^2 and σ_c^2 are weights that determine the relative importance of the two parts of the model. This family

of covariances has five hyperparameters: $k_g \in [0, \infty)$, $\alpha \in (0, 1)$, $k_c \in [0, \infty)$, $\omega_0 \in [0, \pi]$, and $a \in (0, 1)$.

We suppose that an input trace $u(n)$ of Gaussian white noise with variance $\sigma_u^2 = 1/f_s$ is run through H_g yielding an output trace $y(n)$; our observations comprise these two traces, corrupted by additive Gaussian white noise of variance $\sigma^2 = 10^{-4}/f_s$. To obtain an empirical transfer function estimate, we run both observation traces through a bank of 25 windowed 1000-tap DFT filters. The impulse responses of the filter bank are $h_i(n) = e^{j\omega_i n} w(n)$ for $i = 1, \ldots, 25$, with Gaussian window $w(n) = \exp(-\frac{1}{2}(\sigma_w(n-500)/1000)^2)$ for $n = 0, \ldots, 999$, and $w(n) = 0$ otherwise, with window half-width $\sigma_w = 0.25$. Let u_i, y_i denote the outputs of filter h_i with inputs u, y respectively: $y_i(n)/u_i(n)$ gives a running estimate of $g(e^{j\omega_i})$, whose value after 1000 time steps we take as our observation at $z_i = e^{j\omega_i}$. Figure 9.1 shows the regression from the strictly linear estimator (9.33) after tuning the covariance hyperparameters via maximum likelihood, along with predictive error bounds based on $\eta = 3$ confidence ellipsoids.

The second is a second-order allpass filter. This system is specified in discrete time with the transfer function

$$g(z) = \frac{|z_0|^2 - 2\,\mathrm{Re}\,[z_0] + 1}{1 - 2\,\mathrm{Re}\,[z_0] + |z_0|^2}, \tag{9.36}$$

where $z_0 = 0.5e^{\pm j\pi/4}$ are the system's poles, with sampling frequency $f_s = 100$ Hz. For this system we assume that we do not have *a priori* information on the structure of the frequency response, so we use a Hermitian stationary H_∞ process as the prior model. In particular, we take the family of geometric H_∞ process, indexed by hyperparameter $\alpha \in (0, 1)$. To construct the empirical transfer function estimate, we use the same data model and filter bank as the previous example. Figure 9.2 shows the strictly linear regression after tuning the covariance hyperparameters, again with predictive error bounds from $\eta = 3$ confidence ellipsoids.

9.4 Robustness Analysis with H_∞ Gaussian Processes

Now that we have explored how to refine an H_∞ GP model from online data, we move to a second and equally important problem: how to establish probabilistic robustness guarantees for H_∞ processes. Our goal is to establish accuracy-only probabilistic guarantees of robustness; in other words, establishing that a given certificate of robustness is obtained with probability $\geq 1 - \epsilon$ for a prescribed $\epsilon \in (0, 1)$. For several classical robustness certificates, this reduces to the problem of bounding the *gain excursion probability*

$$P_u(f) = \mathbb{P}\left(\sup_{\Omega \in [0,\pi)} |f(e^{j\Omega})| > u\right)$$

of a general H_∞ GP f, which measures how likely the L_2 gain of H_f is to exceed the level u. A simple example of how excursion problems arise is the problem of extending small-gain

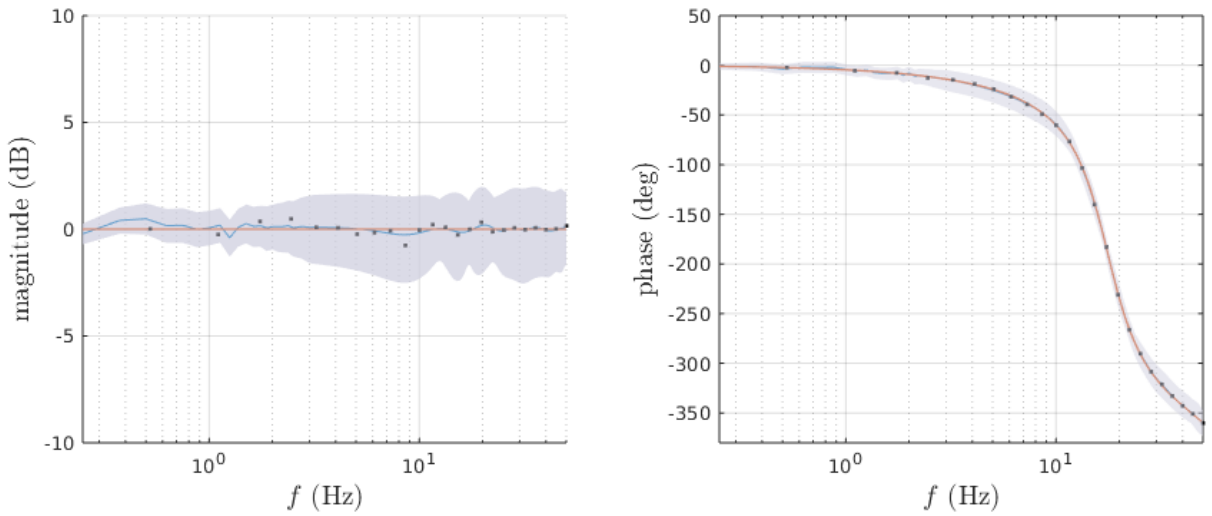

Figure 9.2: Bode plot of the second-order allpass system (orange), and its estimate (blue) using H_∞ Gaussian process regression from an empirical transfer function estimate (black points) with $\eta = 3$ confidence ellipsoid bounds (grey).

arguments to the probabilistic case with an H_∞ GP uncertainty, as in Figure 7. Consider a nominal plant G in feedback with an uncertainty Δ modeled by an H_∞ GP. According to the small-gain theorem [32, Theorem III.2.1], the interconnection will be stable if the gain of $G\Delta$ is less than unity: thus bounding $P_1(G\Delta) \leq \epsilon$ amounts to proving that the interconnection is stable for an ensemble of realizations with probability $\geq 1 - \epsilon$.

A more general example is the problem of proving an accuracy-only probabilistic guarantee that an H_∞ GP satisfies an integral quadratic constraint (IQC). In the discrete time SISO setting, an IQC with multiplier Π is a behavioral constraint of the form [51]

$$\int_{-\pi}^{\pi} \begin{bmatrix} v(e^{j\Omega}) \\ Hv(e^{j\Omega}) \end{bmatrix}^* \Pi(e^{j\Omega}) \begin{bmatrix} v(e^{j\Omega}) \\ Hv(e^{j\Omega}) \end{bmatrix} d\Omega \geq 0 \tag{9.37}$$

where $v \in H_2$, $H : H_2 \to H_2$, $\Pi : [-\pi, \pi) \to \mathbb{C}^{2\times 2}$. A system (represented by H) satisfies the IQC with multiplier Π if (9.37) is satisfied for all conjugate-symmetric $v \in H_2$. IQCs are able to express a wide range of behavioral properties, and knowing that an uncertainty satisfies a particular IQC is a powerful tool for constructing controllers that are robust against it [64, 87].

If Δ is an LTI operator, then under fairly mild conditions, the question of whether the system upholds a given IQC reduces to a pointwise quadratic constraint on its transfer function; constrained either to lie within a frequency-dependent circle, outside a frequency-dependent circle, or on either side of a frequency-dependent half-space. We shall focus on the first case, but an analogous theory can be developed for any case.

Lemma 11 (adapted from [69], Lemma 1 (i))**.** *Suppose that an LTI system with transfer function Δ satisfies an IQC with continuous, conjugate-symmetric multiplier Π. Then for each $\Omega \in [-\pi, \pi)$, $H(e^{j\Omega})$ lies in a circle in the Nyquist plane with center $\Pi_{21}(e^{j\Omega})$ and radius $\sqrt{\Pi_{11}(e^{j\Omega}) + |\Pi_{21}(e^{j\Omega})|^2}$.*

This condition is evidently equivalent to the condition that

$$\frac{|\Delta(e^{j\Omega}) - \Pi_{21}(e^{j\Omega})|}{\sqrt{\Pi_{11}(e^{j\Omega}) + |\Pi_{21}(e^{j\Omega})|^2}} \leq 1 \ \forall \Omega \in [-\pi, \pi). \tag{9.38}$$

Thus the problem of establishing that H_∞ satisfies an IQC with probability $\geq 1-\epsilon$– in other words, establishing an accuracy-only probabilistic guarantee holds– reduces an excursion probability problem on the gain of an H_∞ GP, namely proving that

$$\mathbb{P}\left(\sup_{\Omega\in[-\pi,\pi)} \left| \frac{\Delta(e^{j\Omega}) - \Pi_{21}(e^{j\Omega})}{\sqrt{\Pi_{11}(e^{j\Omega}) + |\Pi_{21}(e^{j\Omega})|^2}} \right| \geq 1 \right) \leq \epsilon, \tag{9.39}$$

or in other words that

$$P_1\left(\frac{\Delta(e^{j\Omega}) - \Pi_{21}(e^{j\Omega})}{\sqrt{\Pi_{11}(e^{j\Omega}) + |\Pi_{21}(e^{j\Omega})|^2}}\right) \leq \epsilon. \tag{9.40}$$

Bounding the excursion probability

Having established how gain excursion probabilities arise in proving probabilistic safety guarantees for H_∞ uncertainties, we turn to the problem of how to control these probabilities. It is not generally possible to directly compute $P_u(f)$; however, we can bound it from above using a related quantity, the expected number of gain upcrossings.

Associated to any H_∞ Gaussian process is its *gain process* $|f(e^{j\Omega})|, \Omega \in [0, 2\pi)$. Assuming that the gain process is differentiable with respect to Ω, a *gain upcrossing* at level u is a value Ω_c such that $|f(e^{j\Omega})| = u$ and $\frac{\partial}{\partial\Omega}|f(e^{j\Omega})| > 0$. Under the assumptions given so far, there can be at most finitely many upcrossings, so the random variable $N_u = \#\{t \in T : |f(e^{j\Omega})| = u \text{ and } \frac{\partial}{\partial\Omega}|f(e^{j\Omega})| > 0\}$ is well-defined and its expectation $\mathbb{E}[N_u]$ is almost surely finite. A simple application of Markov's inequality yields the bound

$$\begin{aligned} \mathbb{P}\left(\sup_{\Omega\in[-\pi,\pi)} |f(e^{j\Omega})| > u\right) &= \mathbb{P}\left(|f(e^{j0})| > u\right) + \mathbb{P}\left(|f(e^{j0})| \leq u, N_u \geq 1\right) \\ &\leq \mathbb{P}\left(|f(e^{j0})| > u\right) + \mathbb{E}\left[N_u\right]. \end{aligned} \tag{9.41}$$

The reason that we consider a bound for P_u rather than a direct computation is that direct computation of P_u is only possible in the simplest cases. On the other hand, $\mathbb{E}[N_u]$ can be computed with a closed-form (though sometimes complicated) expression as long as the process is differentiable.

The lack of Gaussian structure in $|f(e^{j\Omega})|$ would make it difficult to compute this formula directly from f (e.g. by applying a Rice formula like [9, Theorem 3.4]). To overcome the difficulty, we reframe the problem as a *vector crossing* problem on the vector Gaussian process $g(\Omega) = (x(e^{j\Omega}), y(e^{j\Omega}))$ formed from the real and imaginary parts: the gain process $|f(e^{j\Omega})|$ crosses from $\leq u$ to $> u$ precisely when the vector process $g(\Omega)$ crosses from the interior of the circle $x^2 + y^2 = u^2$ to the exterior. By taking this perspective, we relinquish the topological simplicity of the scalar crossing problem in order to retain the Gaussian structure of the stochastic process. While we cannot apply Rice formulas in the vector setting, there are analogous results for counting vector crossings. We use the following result due to Belyaev.

Theorem 8 (first-order Belyaev formula [12])**.** *Let $f : \Xi \times [0,T] \to \mathbb{R}^n$ be a vector-valued stochastic process and $\Phi : \mathbb{R}^n \to \mathbb{R}$ a boundary function satisfying the following conditions:*

1. *f is continuously differentiable with probability one, and the random variables $f(t)$, $t \in T$ all possess densities $p_{f(t)}$;*
2. *The conditional densities $p(x|y)$ exist for $x = f(t)$, $y = f'(t)$, and the densities depend continuously on x.*
3. *Φ is continuously differentiable, and to each ϵ-neighborhoods of the surface $S_\Phi = \{x \in \mathbb{R}^n : \Phi(x) = 0\}$ we can associate the coordinate system $(\phi, \zeta_1, \ldots, \zeta_{n-1})$, where $\phi = \inf_{y \in S_\phi} ||x - y||_2$;*

Let N_ϕ denote the number of times a realization f_ξ of f exits the surface S_ϕ: then

$$\mathbb{E}\left[N_\Phi\right] = \int_0^T \int_{S_\phi} \mathbb{E}\left[n((x)^\top f(t))_+ | f'(t) = x\right] p_{f(t)}(x) ds(x) dt, \tag{9.42}$$

where $n(x)$ is the unit normal vector of S_ϕ at the point x.

Applying the first-order Belyaev formula to the real and imaginary parts of an H_∞ GP and the surface $x^2 + y^2 = u^2$ yields the following formula for the expected number of gain upcrossings.

Theorem 9. *Consider an H_∞ Gaussian process f with mean $m_x + jm_y$ and Hermitian and complementary covariances k, $\tilde{k}$. Let N_u denote the integer-valued random variable that counts the number of u-level gain upcrossings of f. Suppose that $m_x(e^{j\Omega})$ and $m_y(e^{j\Omega})$ are differentiable with respect to Ω and that $k(e^{j\Omega_1}, e^{j\Omega_2})$ and $\tilde{k}(e^{j\Omega_1}, e^{j\Omega_2})$ are thrice differentiable with respect to Ω_1 and Ω_2. Then the expected number of gain upcrossings between frequencies 0 and π is is*

$$\begin{aligned}\mathbb{E}\left[N_u\right] = \int_0^\pi \int_0^{2\pi} \frac{u}{2\pi} \left(\frac{\sigma_{z(\theta)}(\Omega)}{\sqrt{2\pi}} e^{-\frac{1}{2}(\mu_{z(\theta)}(\Omega)/\sigma_{z(\theta)}(\Omega))^2} + \frac{1}{2}\mu_{z(\theta)}(\Omega) \left(1 + \operatorname{erf}\left(\frac{\mu_{z(\theta)}(\Omega)}{\sqrt{2}\sigma_{z(\theta)}(\Omega)}\right)\right)\right) \\ \times \det \Sigma(\Omega,\Omega)^{-1/2} e^{-\frac{1}{2}(z(\theta)-m(\Omega))^\top \Sigma(\Omega,\Omega)^{-1}(z(\theta)-m(\Omega)} d\theta d\Omega,\end{aligned} \tag{9.43}$$

where

$$\mu_z(\Omega) = u^{-1} z^\top m(\Omega)' + u^{-1} z^\top C(\Omega,\Omega)\Sigma(\Omega,\Omega)^{-1}(z - m(\Omega)) \tag{9.44}$$
$$\sigma_z(\Omega) = u^{-2} z^\top (\Sigma'(\Omega,\Omega) - C(\Omega,\Omega)(\Sigma(\Omega,\Omega)^{-1} C(\Omega,\Omega)^\top) z \tag{9.45}$$

$$\Sigma(\Omega,\Omega) = \begin{bmatrix} k_x(\Omega,\Omega) & k_c(\Omega,\Omega) \\ k_c(\Omega,\Omega) & k_y(\Omega,\Omega) \end{bmatrix} \tag{9.46}$$
$$\Sigma'(\Omega,\Omega) = \begin{bmatrix} k_x^{12}(\Omega,\Omega) & k_c^{12}(\Omega,\Omega) \\ k_c^{12}(\Omega,\Omega) & k_y^{12}(\Omega,\Omega) \end{bmatrix}, \tag{9.47}$$
$$C(\Omega,\Omega) = \begin{bmatrix} k_x^{1}(\Omega,\Omega) & k_c^{1}(\Omega,\Omega) \\ k_c^{2}(\Omega,\Omega) & k_y^{1}(\Omega,\Omega) \end{bmatrix}, \tag{9.48}$$

$$k_x(\Omega_1,\Omega_2) = \tfrac{1}{2}\,\mathrm{Re}\left[k(e^{j\Omega_1}, e^{j\Omega_2}) + \tilde{k}(e^{j\Omega_1}, e^{j\Omega_2})\right] \tag{9.49}$$
$$k_y(\Omega_1,\Omega_2) = \tfrac{1}{2}\,\mathrm{Re}\left[k(e^{j\Omega_1}, e^{j\Omega_2}) - \tilde{k}(e^{j\Omega_1}, e^{j\Omega_2})\right] \tag{9.50}$$
$$k_c(\Omega_1,\Omega_2) = \tfrac{1}{2}\,\mathrm{Im}\left[\tilde{k}(e^{j\Omega_1}, e^{j\Omega_2}) - k(e^{j\Omega_1}, e^{j\Omega_2})\right] \tag{9.51}$$
$$m(\Omega) = \begin{bmatrix} m_x(e^{j\Omega}) \\ m_y(e^{j\Omega}) \end{bmatrix}, \tag{9.52}$$

and where the superscripts denote derivatives of the covariance functions, e.g.

$$k_x^{12} = \partial^2 k_x / \partial\Omega_1\Omega_2.$$

Proof. To apply Belyaev's formula, we must first establish that the conditions of Theorem 8 are satisfied by the vector GP g composed of the real and imaginary parts of f. That $g(e^{j\Omega})$ and $g'(e^{j\Omega})|g(e^{j\Omega})$ have distributions is given by the fact that g is a Gaussian process: $g(e^{j\Omega})$ is Gaussian-distributed by definition; the derivative of a Gaussian process is itself a Gaussian process; and a Gaussian random variable conditioned on another Gaussian random variable is itself Gaussian. The given conditions on differentiability ensure that g is differentiable: a mean-zero process with thrice-differentiable covariances is at least once-differentiable, and adding a differentiable mean to the process preserves this property. Finally, the surface $x^2+y^2-u^2=0$ satisfies the third condition, as an ϵ-neighborhood of S_ϕ is simply an annulus in the plane with inner radius $u-\epsilon$ and outer radius $u+\epsilon$, which can be parameterized using polar coordinates.

With these conditions established, we know that the expected number of u-level gain upcrossings of f is given by (9.42) with $g = (x, y)$ and $\Phi(x, y) = x^2 + y^2 - u^2$; all that remains is to show how to express (9.42) in terms computable from k, $\tilde{k}$, and m, that is to derive (9.43). First, there is the matter of integration over S_Φ: this can be handled by integrating over its circular parameterization $S_\Phi = \{(u\cos\theta, u\sin\theta), \theta \in [-\pi, \pi)\}$ in which case

$$z = z(\theta) = (u\sin\theta, u\cos\theta), \quad ds(z) = u d\theta. \tag{9.53}$$

To obtain the distribution $p_{g(\Omega)}$, we require the first- and second-order statistics of g. Since the real and imaginary parts of f are Gaussian, it follows that g is a vector Gaussian process. Its mean and variance are

$$\mathbb{E}\left[g(e^{j\Omega})\right] = \begin{bmatrix}\mathbb{E}\left[x(e^{j\Omega})\right] \\ \mathbb{E}\left[y(e^{j\Omega})\right]\end{bmatrix} = m(\Omega), \tag{9.54}$$

$$\begin{aligned}&\Sigma(\Omega_1,\Omega_2) = \\ &\begin{bmatrix}\mathbb{E}\left[(x(\Omega_1)-m_x(\Omega_1))(x(\Omega_2)-m_x(\Omega_2))\right] & \mathbb{E}\left[(x(\Omega_1)-m_x(\Omega_1))(y(\Omega_2)-m_y(\Omega_2))\right] \\ \mathbb{E}\left[(y(\Omega_1)-m_y(\Omega_1))(x(\Omega_2)-m_x(\Omega_2))\right] & \mathbb{E}\left[(y(\Omega_1)-m_y(\Omega_1))(y(\Omega_2)-m_y(\Omega_2))\right]\end{bmatrix} \\ &\qquad = \begin{bmatrix}k_x(\Omega_1,\Omega_2) & k_c(\Omega_1,\Omega_2) \\ k_c(\Omega_1,\Omega_2) & k_y(\Omega_1,\Omega_2)\end{bmatrix},\end{aligned} \tag{9.55}$$

which can be verified by substituting

$$\begin{aligned}x(e^{j\Omega}) &= \frac{1}{2}(f(e^{j\Omega}) + \bar{f}(e^{j\Omega})), \\ y(e^{j\Omega}) &= \frac{1}{2j}(f(e^{j\Omega}) - \bar{f}(e^{j\Omega})),\end{aligned} \tag{9.56}$$

and working out the expectations. From this, it follows that

$$p_{f(\Omega)}(z) = (2\pi)^{-1}\Sigma(\Omega,\Omega)e^{-\frac{1}{2}(z-m(\Omega))^\top \Sigma(\Omega,\Omega)(z-m(\Omega))}. \tag{9.57}$$

To compute the conditional expectation, we also require the joint distribution of $g(\Omega)$ and its derivative $g'(\Omega)$: this is another multivariate normal with mean and covariance

$$\mathbb{E}\left[\begin{bmatrix}x'(\Omega) \\ y'(\Omega) \\ x(\Omega) \\ y(\Omega)\end{bmatrix}\right] = \begin{bmatrix}m_x'(\Omega) \\ m_y'(\Omega) \\ m_x(\Omega) \\ m_y(\Omega)\end{bmatrix} \tag{9.58}$$

$$\mathbb{E}\left[\begin{bmatrix}x'(\Omega)-m_x(\Omega) \\ y'(\Omega)-m_y(\Omega) \\ x(\Omega)-m_x(\Omega) \\ y(\Omega)-m_y(\Omega)\end{bmatrix}\begin{bmatrix}x'(\Omega)-m_x(\Omega) \\ y'(\Omega)-m_y(\Omega) \\ x(\Omega)-m_x(\Omega) \\ y(\Omega)-m_y(\Omega)\end{bmatrix}^\top\right] = \begin{bmatrix}\Sigma'(\Omega,\Omega) & C(\Omega,\Omega) \\ C^\top(\Omega,\Omega) & \Sigma(\Omega,\Omega)\end{bmatrix}, \tag{9.59}$$

We next compute the conditional distribution of g given $g' = z$; this is again a multivariate normal: dropping Ω from the submatrices for brevity, we have

$$g(\Omega)|g'(\Omega) = z \sim \mathcal{N}\left(m' + C\Sigma^{-1}(z-m), \Sigma' - C\Sigma^{-1}C^\top\right). \tag{9.60}$$

The normal vector $n_\Phi(z)$ for Φ—a circle with center zero and radius u– is simply u^-1. Thus the linear mapping $(g(\Omega)|g'(\Omega) = z) \mapsto n_\Phi(z)^\top(g(\Omega)|g'(\Omega) = z)$ gives us the distribution

$$\begin{aligned}n_\Phi(z)^\top(g(\Omega)|g'(\Omega) = z) &\sim \mathcal{N}\left(u^{-1}z^\top m' + u^{-1}z^\top C\Sigma^{-1}(z-m), u^{-2}z^\top(\Sigma' - C(\Sigma^{-1}C^\top)z\right) \\ &\triangleq \mathcal{N}\left(\mu_z(\Omega), \sigma_z^2(\Omega)\right).\end{aligned} \tag{9.61}$$

The remaining step is to compute the expectation of the positive part. Since

$$n_\Phi(z)^\top (g(\Omega)|g'(\Omega) = z) \tag{9.62}$$

is a Gaussian scalar, the positive part is a rectified Gaussian, whose mean is

$$\mathbb{E}\left[(n_\Phi(z)^\top (g(\Omega))_+|g'(\Omega) = z)\right] = \frac{\sigma_z(\Omega)}{\sqrt{2\pi}} e^{-\frac{1}{2}(\mu_z(\Omega)/\sigma_z(\Omega))^2} + \frac{1}{2}\mu_z(\Omega)\left(1 + \operatorname{erf}\left(\frac{\mu_z(\Omega)}{\sqrt{2}\sigma_z(\Omega)}\right)\right). \tag{9.63}$$

Finally, applying (9.53), (9.56), and (9.62) to the Belyaev formula (9.42) yields the expression (9.43). □

Example: Bounding the gain of a geometric H_∞ Process

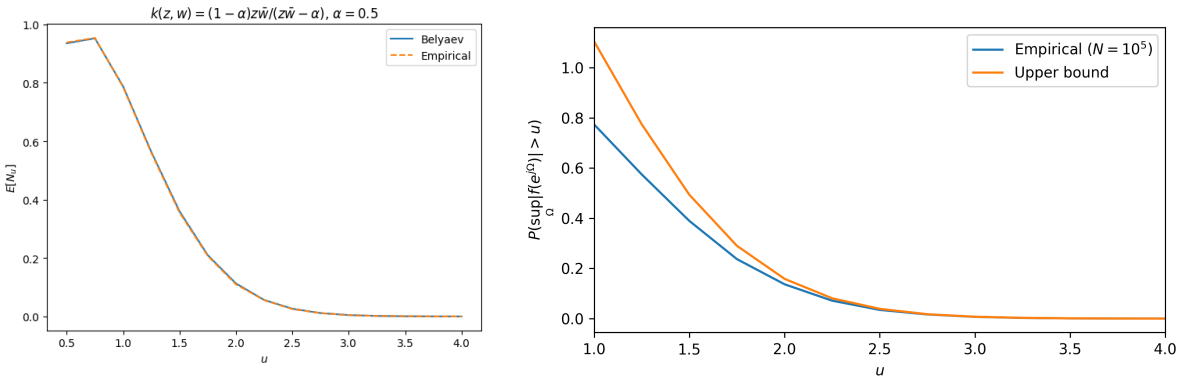

Figure 9.3: Evaluating the Belyaev formula and the excursion gain bound on a geometric H_∞ process with $\alpha = 0.5$. Left: Numerical comparison of (9.42) with an empirical estimate ($N = 100,000$) of $\mathbb{E}[N_u]$. Right: Numerical comparison of (9.41) with an empirical estimate ($N = 100,000$) of $P_u(f)$.

To demonstrate the validity of the Belyaev formula (9.42) and to test the tightness of the bound (9.41), we use them to numerically evaluate $\mathbb{E}[N_u]$ and a bound for $P_u(f)$ for a geometric H_∞ GP f with $\alpha = 0.5$ over a range $u \in [1, 4]$. The expected H_2 norm of a geometric H_∞ process is 1, so it's reasonable to expect $\mathbb{E}[N_u]$ and $P_u(f)$ to be relatively high near $u = 1$, and to taper off relatively quickly. Figure 9.3 shows the results of the numerical evaluation of Equations (9.42), (9.41) and compares the results to empirical approximations of $\mathbb{E}[N_u]$ and $P_u(f)$ made using $N = 100,000$ process realizations. We can conclude from the figure that (9.42) accurately computes $\mathbb{E}[N_u]$ as expected. Furthermore, we see that (9.41) indeed upper bounds $P_u(f)$; while the bound is conservative in regions where $P_u(f)$ is high, it quickly becomes tighter where $P_u(f)$ is small, levelling off to overestimate $P_u(f)$ by $\sim 10\%$ for $u > 2.5$. Since we will generally engineer $P_u(f)$ to be tight in applications, the experiment shows that (9.41) is not a conservative bound in regions of practical interest.

9.5 Conclusion

This chapter has three principal contributions: how to construct H_∞ GPs, how to refine an H_∞ GP model with data, and how to prove robustness guarantees for systems with H_∞ feedback uncertainties. There are significant improvements to be made in all three.

Regarding construction of H_∞ GPs, the general existence conditions are difficult to evaluate in practice; while the Hermitian stationary case provides a convenient workaround, it would be better to have more convenient general conditions. The Driscoll zero-one condition can likely be replaced by a direct appeal to the analyticity of the covariance, as in [11].

Regarding regression, the regression-based refinement method described in Section 9.3 depends strongly on the method used to convert time-domain data to frequency-domain, many of which (such as filter banks and ETFEs) are sensitive to measurement noise. A more direct application of time-domain data to the frequency-domain model, perhaps effected by a *projected-process regression* [73, §8.3.4]. Even within the regime of refinement via frequency-domain data, there remains some mystery around the performance gap between strictly linear and widely linear estimators. The strictly linear estimator, which is suboptimal for general complex Gaussian process priors, appears to provide transfer function estimates that are close to optimal for conjugate-symmetric H_∞ priors. We have numerical evidence that suggests that as the number of frequency data points increases, the covariance becomes *maximally improper*, a case in which the strictly linear is indeed optimal.

Regarding robustness, there is not a clear path to extend the Belyaev-based excursion probability bounds to the MIMO case. In this case, we would seek to bound the maximum singular value (over all frequencies) of a matrix of random transfer functions; in this case, restoring Gaussian structure would be more difficult than in the SISO case.

Chapter 10

Afterword: A Bespoke Learning Model for Robust Control

To conclude Part II, I'd like to remind you of a fundamental contradiction of learning-based control that we identified in Chapter 7: A dynamical model that is better for learning is necessarily "worse" for control, in the sense that our ability to synthesize controllers and establish safety certificates diminishes as the complexity and generality of the dynamical model increases. In the light of this contradiction, I would venture that there is no singular "best" model for safe, learning-based control. At least, there are no such models currently in the literature. If we wish one to exist, we must make it; that is, we must make a bespoke learning model for robust control. The models we've explored in Part II demonstrate promising directions to navigating the fundamental contradiction and finding such a bespoke model, though they fall short of perfection in their current forms. In a break from much of the learning-based control literature, which uses GP models and neural nets directly as they are used in general machine learning tasks and then beat the control analysis to fit, these directions elect weaken the power of the learning model in order to retain more of the control-theoretic structure, so that we can retain as many of the tools of robust control as possible.

In Chapter 8, we took an existing ML model, a GP regression model with polynomial covariance, and with very little modification added it[1] to an existing control model, a polynomial control-affine state-space system. This type of modeling, wherein we adjoin essentially independent learning and control models, is far and away the most common approach in literature. Rather than retain complete freedom of covariance structure and try to figure out what control analysis and synthesis tools will still work (the answer is very few), our approach was to select a tool at the outset and restrict the covariance structure so that the model is compatible by design. The tool was SOS analysis, and the corresponding structure was polynomial structure. By altering the model to fit the tool, we achieved a model suitable for both learning and robustness. This was obtained at the cost of a considerable reduction

[1] Literally, as an additive uncertainty.

of the flexibility of the model, but in some cases (e.g. learning dynamics in the neighborhood of an equilibrium) this limitation is acceptable.

In Chapter 9, we designed a learning model expressly for the control task of interest. The control task of interest was robust control of H_∞ linear dynamical systems in frequency domain, and the learning model we constructed was the H_∞ Gaussian process. By their very construction, H_∞ GPs fit nicely into the paradigm of robust control. Specifically, a number of established methods for proving robust stability and other behavioral constraints—small-gain arguments and integral quadratic constraints, for instance—have clear extensions for H_∞ GPs. Furthermore, the way to prove the corresponding probabilistic robustness guarantees is straightforward, reducing to the problem of bounding the supremum of a stochastic process. We can even find closed-form, if perhaps cumbersome, expressions for bounds on these probabilities using only the covariance structure of the GP.

Despite being designed from the ground up with robust control in mind, the class of H_∞ GPs constructed in Chapter 9 is not universally appropriate for modeling uncertain systems. First there is the fact that the theory is so far only developed for SISO systems. Formally extending to a MIMO model presents no difficulty; if nothing else, one can form a transfer function matrix whose components are H_∞ GPs. Regression extends to this case rather easily, but robustness does not. In particular, the criterion for robustness becomes more complicated, changing from the supremum of a single H_∞ process to the supremum of the singular values of a matrix of H_∞ processes, which lacks the structure to effectively compute the terms of the Belyaev formula. There is no obvious way to recover this structure, so establishing robustness of MIMO H_∞ processes remains an open problem.

A curious property of H_∞ processes is that the loci of their poles —always a subset of the unit disk, of course—are fixed by their covariance, and are the same for all realizations. Only the location of the zeros differ among realizations. This is due to the fact that the gain at a particular frequency is a series of independent Gaussians, and such series converge either with probability zero or probability one. That is not to say that the model cannot adapt to the poles of the ground truth: both Bayesian regression and hyperparameter tuning will alter the poles by altering the covariance structure. Nevertheless, it demonstrates an asymmetry that I believe is fundamental to Gaussian transfer function models: zeros are parameters, and poles are hyperparameters.

Bibliography

[1] Robert J Adler and Jonathan E Taylor. *Applications of random fields and geometry: applications and case studies*. Unpublished manuscript, 2007. URL: `https://robert.net.technion.ac.il/publications/`.

[2] Robert J Adler and Jonathan E Taylor. *Random fields and geometry*. Vol. 80. Springer, 2007.

[3] Mohamadreza Ahmadi, Arie Israel, and Ufuk Topcu. "Safe Controller Synthesis for Data-Driven Differential Inclusions". In: *IEEE Transactions on Automatic Control* (2020).

[4] Anayo K Akametalu et al. "Reachability-based safe learning with Gaussian processes". In: *53rd IEEE Conference on Decision and Control*. IEEE. 2014, pp. 1424–1431.

[5] Teodoro Alamo, Roberto Tempo, and Eduardo F Camacho. "Randomized strategies for probabilistic solutions of uncertain feasibility and optimization problems". In: *IEEE Transactions on Automatic Control* 54.11 (2009), pp. 2545–2559.

[6] Nachman Aronszajn. "Theory of reproducing kernels". In: *Transactions of the American mathematical society* 68.3 (1950), pp. 337–404.

[7] Armin Askari, Forest Yang, and Laurent El Ghaoui. "Kernel-based outlier detection using the inverse Christoffel function". In: *arXiv preprint arXiv:1806.06775* (2018).

[8] Karl J Åström and Björn Wittenmark. *Adaptive control*. Addison-Wesley, 1995.

[9] Jean-Marc Azaïs and Mario Wschebor. *Level sets and extrema of random processes and fields*. John Wiley & Sons, 2009.

[10] Gary Balas, Peter Seiler, and Andrew Packard. "Analysis of an UAV flight control system using probabilistic μ". In: *AIAA Guidance, Navigation, and Control Conference*. 2012, p. 4989.

[11] Yu K Belyaev. "Analytic random processes". In: *Theory of Probability & Its Applications* 4.4 (1959), pp. 402–409.

[12] Yu K Belyaev. "On the number of exits across the boundary of a region by a vector stochastic process". In: *Theory of Probability & Its Applications* 13.2 (1968), pp. 320–324.

[13] Felix Berkenkamp, Angela P Schoellig, and Andreas Krause. "Safe controller optimization for quadrotors with Gaussian processes". In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2016, pp. 491–496.

[14] Felix Berkenkamp et al. "Safe learning of regions of attraction for uncertain, nonlinear systems with Gaussian processes". In: *2016 IEEE 55th Conference on Decision and Control (CDC)*. IEEE. 2016, pp. 4661–4666.

[15] Felix Berkenkamp et al. "Safe model-based reinforcement learning with stability guarantees". In: *Advances in neural information processing systems*. 2017, pp. 908–918.

[16] Jean-Marc Biannic et al. "Advanced probabilistic μ-analysis techniques for AOCS validation". In: *European Journal of Control* 62 (2021), pp. 120–129.

[17] Patrick Bouffard. "On-board model predictive control of a quadrotor helicopter: design, implementation, and experiments". In: (2012). URL: `http://www.eecs.berkeley.edu/Pubs/TechRpts/2012/EECS-2012-241.html`.

[18] Giuseppe C Calafiore and Marco C Campi. "The scenario approach to robust control design". In: *IEEE Transactions on automatic control* 51.5 (2006), pp. 742–753.

[19] Giuseppe C Calafiore and Fabrizio Dabbene. "Probabilistic robust control". In: *2007 American Control Conference*. IEEE. 2007, pp. 147–158.

[20] Frank M Callier and Charles A Desoer. *Linear system theory*. Springer Science & Business Media, 1991.

[21] Marco C Campi and Simone Garatti. *Introduction to the scenario approach*. SIAM, 2018.

[22] Marco C Campi and Simone Garatti. "Wait-and-judge scenario optimization". In: *Mathematical Programming* 167.1 (2018), pp. 155–189.

[23] Marco C Campi, Simone Garatti, and Federico A Ramponi. "Non-convex scenario optimization with application to system identification". In: *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE. 2015, pp. 4023–4028.

[24] Tianshi Chen, Henrik Ohlsson, and Lennart Ljung. "On the estimation of transfer functions, regularizations and Gaussian processes—Revisited". In: *Automatica* 48.8 (2012), pp. 1525–1535.

[25] Richard Cheng et al. "End-to-End Safe Reinforcement Learning through Barrier Functions for Safety-Critical Continuous Control Tasks". In: *arXiv:1903.08792* (Mar. 2019).

[26] Yinlam Chow et al. "A Lyapunov-based approach to safe reinforcement learning". In: *Advances in neural information processing systems*. 2018, pp. 8092–8101.

[27] Yinlam Chow et al. "Lyapunov-based safe policy optimization for continuous control". In: *arXiv preprint arXiv:1901.10031* (2019).

[28] Samuel Coogan and Murat Arcak. "A benchmark problem in transportation networks". In: *arXiv preprint arXiv:1803.00367* (2018).

[29] Antonio Cuevas and Ricardo Fraiman. "A plug-in approach to support estimation". In: *The Annals of Statistics* 25.6 (1997), pp. 2300–2312.

[30] Carlos F Daganzo. "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory". In: *Transportation Research Part B: Methodological* 28.4 (1994), pp. 269–287.

[31] Claudio De Persis and Pietro Tesi. "Formulas for data-driven control: Stabilization, optimality, and robustness". In: *IEEE Transactions on Automatic Control* 65.3 (2019), pp. 909–924.

[32] Charles A Desoer and Mathukumalli Vidyasagar. *Feedback systems: input-output properties*. SIAM, 2009.

[33] Alex Devonport and Murat Arcak. "Data-driven reachable set computation using adaptive Gaussian process classification and Monte Carlo methods". In: *2020 American Control Conference (ACC)*. IEEE. 2020, pp. 2629–2634.

[34] Alex Devonport and Murat Arcak. "Estimating reachable sets with scenario optimization". In: *Proceedings of the 2nd Conference on Learning for Dynamics and Control*. Ed. by Alexandre M. Bayen et al. Vol. 120. Proceedings of Machine Learning Research. The Cloud: PMLR, 2020, pp. 75–84. URL: http://proceedings.mlr.press/v120/devonport20a.html.

[35] Alex Devonport, Adnane Saoud, and Murat Arcak. "Symbolic abstractions from data: A PAC learning approach". In: *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE. 2021, pp. 599–604.

[36] Alex Devonport, Peter Seiler, and Murat Arcak. "Frequency Domain Gaussian Process Models for H^∞ Uncertainties". In: *Learning for Dynamics and Control Conference*. PMLR. 2023, pp. 1046–1057.

[37] Alex Devonport, He Yin, and Murat Arcak. "Bayesian safe learning and control with sum-of-squares analysis and polynomial kernels". In: *2020 59th IEEE Conference on Decision and Control (CDC)*. IEEE. 2020, pp. 3159–3165.

[38] Alex Devonport et al. *Data-Driven Reachability analysis and Support set Estimation with Christoffel Functions*. 2021. arXiv: 2112.09995 [eess.SY].

[39] Alex Devonport et al. "Data-driven reachability analysis with Christoffel functions". In: *2021 60th IEEE Conference on Decision and Control (CDC)*. IEEE. 2021, pp. 5067–5072.

[40] Alex Devonport et al. "PIRK: scalable interval reachability analysis for high-dimensional nonlinear systems". In: *Computer Aided Verification*. Ed. by Shuvendu K. Lahiri and Chao Wang. Cham: Springer International Publishing, 2020, pp. 556–568. ISBN: 978-3-030-53288-8.

[41] Michael F Driscoll. "The reproducing kernel Hilbert space structure of the sample paths of a Gaussian process". In: *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* 26.4 (1973), pp. 309–316.

[42] Richard M Dudley. "Central limit theorems for empirical measures". In: *The Annals of Probability* (1978), pp. 899–929.

[43] Rick Durrett. *Probability: theory and examples*. Vol. 49. Cambridge University Press, 2019.

[44] Peyman Mohajerin Esfahani, Tobias Sutter, and John Lygeros. "Performance bounds for the scenario approach and an extension to a class of non-convex programs". In: *IEEE Transactions on Automatic Control* 60.1 (2014), pp. 46–58.

[45] C. Fan et al. "DryVR: data-driven verification and compositional reasoning for automotive systems". In: *International Conference on Computer Aided Verification*. Springer. 2017, pp. 441–461.

[46] David D. Fan et al. "Bayesian Learning-Based Adaptive Control for Safety Critical Systems". In: *arXiv preprint arXiv:1910.02325*, arXiv:1910.02325 (Oct. 2019). arXiv: `1910.02325 [eess.SY]`.

[47] Jaime F Fisac et al. "A general safety framework for learning-based control in uncertain robotic systems". In: *IEEE Transactions on Automatic Control* 64.7 (2018), pp. 2737–2752.

[48] Marco Gallieri et al. "Safe Interactive Model-Based Learning". In: *arXiv:1911.06556* (Nov. 2019).

[49] Gilbert Helmberg. *Introduction to spectral theory in Hilbert space*. North-Holland, 1969.

[50] Zhong-Sheng Hou and Zhuo Wang. "From model-based control to data-driven control: Survey, classification and perspective". In: *Information Sciences* 235 (2013), pp. 3–35.

[51] Bin Hu, Márcio J Lacerda, and Peter Seiler. "Robustness analysis of uncertain discrete-time systems with dissipation inequalities and integral quadratic constraints". In: *International Journal of Robust and Nonlinear Control* 27.11 (2017), pp. 1940–1962.

[52] Zachary Jarvis-Wloszek et al. "Controls Applications of Sum of Squares Programming". In: *Positive Polynomials in Control*. Vol. 312. Springer, Berlin, Heidelberg, 2005.

[53] Ming Jin and Javad Lavaei. "Stability-certified reinforcement learning: A control-theoretic perspective". In: *arXiv preprint arXiv:1810.11505* (2018).

[54] Guy Katz et al. "Reluplex: An efficient SMT solver for verifying deep neural networks". In: *Computer Aided Verification: 29th International Conference, CAV 2017, Heidelberg, Germany, July 24-28, 2017, Proceedings, Part I 30*. Springer. 2017, pp. 97–117.

[55] S Mohammad Khansari-Zadeh and Aude Billard. "Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions". In: *Robotics and Autonomous Systems* 62.6 (2014), pp. 752–765.

[56] S. Mohammad Khansari-Zadeh and Aude Billard. "Learning control Lyapunov function to ensure stability of dynamical system-based robot reaching motions". In: *Robotics and Autonomous Systems* 62.6 (2014), pp. 752 –765. ISSN: 0921-8890.

[57] Sven Khatri and Pablo A Parrilo. "Guaranteed bounds for probabilistic μ". In: *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No. 98CH36171)*. Vol. 3. IEEE. 1998, pp. 3349–3354.

[58] Henry J Landau and Lawrence A Shepp. "On the supremum of a Gaussian process". In: *Sankhyā: The Indian Journal of Statistics, Series A* (1970), pp. 369–378.

[59] John Langford and Robert Schapire. "Tutorial on Practical Prediction Theory for Classification". In: *Journal of Machine Learning Research* 6.3 (2005).

[60] Jean B Lasserre and Edouard Pauwels. "The empirical Christoffel function in statistics and machine learning". In: *arXiv preprint arXiv:1701.02886* (2017).

[61] Jean B Lasserre and Edouard Pauwels. "The empirical Christoffel function with applications in data analysis". In: *Advances in Computational Mathematics* 45.3 (2019), pp. 1439–1468.

[62] John Lataire and Tianshi Chen. "Transfer function and transient estimation by Gaussian process regression in the frequency domain". In: *Automatica* 72 (2016), pp. 217–229.

[63] David A McAllester. "Some PAC-Bayesian theorems". In: *Machine Learning* 37.3 (1999), pp. 355–363.

[64] Alexandre Megretski and Anders Rantzer. "System analysis via integral quadratic constraints". In: *IEEE transactions on automatic control* 42.6 (1997), pp. 819–830.

[65] Pierre-Jean Meyer, Alex Devonport, and Murat Arcak. "TIRA: toolbox for interval reachability analysis". In: *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*. ACM. 2019, pp. 224–229.

[66] Ian M Mitchell, Jacob Budzis, and Andriy Bolyachevets. "Invariant, viability and discriminating kernel under-approximation via zonotope scaling". In: *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*. 2019, pp. 268–269.

[67] Pablo Parrilo. "Structured Semidefinite Programs and Semialgebraic Geometry Methods in Robustness and Optimization". In: *PhD thesis, California Institute of Technology* (2000).

[68] Vern I Paulsen and Mrinal Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*. Vol. 152. Cambridge University Press, 2016.

[69] Harald Pfifer and Peter Seiler. "Integral quadratic constraints for delayed nonlinear and parameter-varying systems". In: *Automatica* 56 (2015), pp. 36–43.

[70] Gianluigi Pillonetto and Giuseppe De Nicolao. "A new kernel-based approach for linear system identification". In: *Automatica* 46.1 (2010), pp. 81–93.

[71] B. Qi et al. "DryVR 2.0: a tool for verification and controller synthesis of black-box cyber-physical systems". In: *Proceedings of the 21st International Conference on Hybrid Systems: Computation and Control (part of CPS Week)*. 2018, pp. 269–270.

[72] Lawrence R Rabiner and Bernard Gold. "Theory and application of digital signal processing". In: *Englewood Cliffs: Prentice-Hall* (1975).

[73] Carl Edward Rasmussen and Christopher K.I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[74] Spencer M Richards, Felix Berkenkamp, and Andreas Krause. "The Lyapunov neural network: Adaptive stability certification for safe learning of dynamical systems". In: *arXiv preprint arXiv:1808.00924* (2018).

[75] Walter Rudin. *Fourier analysis on groups*. Interscience, 1962.

[76] Peter J Schreier and Louis L Scharf. *Statistical signal processing of complex-valued data: the theory of improper and noncircular signals*. Cambridge University Press, 2010.

[77] Matthias Seeger. "PAC-Bayesian generalisation error bounds for Gaussian process classification". In: *Journal of Machine Learning Research* 3.Oct (2002), pp. 233–269.

[78] Franca Somers et al. "Probabilistic gain, phase and disk margins with application to AOCS validation". In: *IFAC-PapersOnLine* 55.25 (2022), pp. 1–6.

[79] Niranjan Srinivas et al. "Gaussian process optimization in the bandit setting: No regret and experimental design". In: *arXiv preprint arXiv:0912.3995* (2009).

[80] Jeremy G Stoddard et al. "Gaussian process regression for the estimation of generalized frequency response functions". In: *Automatica* 106 (2019), pp. 161–167.

[81] Wentao Tang and Prodromos Daoutidis. "Data-driven control: Overview and perspectives". In: *2022 American Control Conference (ACC)*. IEEE. 2022, pp. 1048–1064.

[82] Andrew Taylor et al. "Learning for Safety-Critical Control with Control Barrier Functions". In: *arXiv preprint arXiv:1912.10099* (2019).

[83] Roberto Tempo, Giuseppe Calafiore, and Fabrizio Dabbene. *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media, 2012.

[84] Michael J Todd. *Minimum-volume ellipsoids: theory and algorithms*. SIAM, 2016.

[85] Leslie G Valiant. "A theory of the learnable". In: *Communications of the ACM* 27.11 (1984), pp. 1134–1142.

[86] Charles F Van Loan and G Golub. *Matrix computations*. The Johns Hopkins University Press, 1996.

[87] Joost Veenman, Carsten W Scherer, and Hakan Köroğlu. "Robust stability and performance analysis based on integral quadratic constraints". In: *European Journal of Control* 31 (2016), pp. 1–32.

[88] Mathukumalli Vidyasagar. *Learning and Generalisation: With Applications to Neural Networks*. Springer Science & Business Media, 2002.

[89] Julia Vinogradska et al. "Stability of controllers for Gaussian process dynamics". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 3483–3519.

[90] Li Wang, Evangelos A Theodorou, and Magnus Egerstedt. "Safe learning of quadrotor dynamics using barrier certificates". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2018, pp. 2460–2465.

[91] Christopher Williams and Matthias Seeger. "Using the Nyström method to speed up kernel machines". In: *Proceedings of the 14th Annual Conference on Neural Information Processing Systems*. 2001, pp. 682–688.

[92] *xkcd: Group Chat Rules*. `https://xkcd.com/2235/`. Accessed: 2023-08-08.

[93] He Yin, Peter Seiler, and Murat Arcak. "Stability analysis using quadratic constraints for systems with neural network controllers". In: *IEEE Transactions on Automatic Control* 67.4 (2021), pp. 1980–1987.

[94] Nicholas Young. *An introduction to Hilbert space*. Cambridge University Press, 1988.