Fakultät für Elektrotechnik und Informationstechnik
Technische Universität München

Formal Synthesis of Controllers for Complex Dynamical Systems: State-Space Discretization-free Approaches

Pushpak Jagtap

Vollständiger Abdruck der von der Fakultät für Elektrotechnik und Informationstechnik der Technischen Universität München zur Erlangung des akademischen Grades eines

Doktor-Ingenieurs (Dr.-Ing.)

genehmigten Dissertation.

Vorsitzender:
Prof. Dr.-Ing. Klaus Diepold

Prüfende der Dissertation:
1. Prof. Dr.-Ing./Univ. Tokio Martin Buss
2. Prof. Dr. Majid Zamani,
University of Colorado Boulder, USA and Ludwig Maximilian University of Munich, Germany

Die Dissertation wurde am 28.05.2020 bei der Technischen Universität München eingereicht und durch die Fakultät für Elektrotechnik und Informationstechnik am 09.11.2020 angenommen.

This thesis is dedicated to my family for their love, endless support, and encouragement.

Acknowledgments

This dissertation is the result of four years of my doctoral research studies in the Hybrid Control Systems (HyConSys) Lab, Department of Electrical and Computer Engineering at the Technical University of Munich (TUM), Germany. In this short note, I would like to take the opportunity and acknowledge those people who supported me to make the completion of this thesis possible.

First and foremost, I would like to express my sincere gratitude to my Ph.D. advisor Prof. Dr. Majid Zamani who offered me a Ph.D. position and showed great trust in my abilities. I would deeply appreciate his consistent support, encouragement, and generous advice throughout these years. I would like to thank him for introducing me to this interesting topic and guiding me through the whole research work. I will be forever grateful for the precious knowledge and skills he has imparted on me.

Besides, I would also like to extend my sincere gratitude to Prof. Dr.-Ing./Univ. Tokio Martin Buss for welcoming me in his research group in the Chair of Automatic Control Engineering at the Technical University of Munich, Germany since July 2019. I would greatly appreciate all his generous help, support, and consideration during this time.

My deep thanks also go to the International Graduate School of Science and Engineering (IGSSE) at the Technical University of Munich for supporting my Ph.D. studies and providing me with an interdisciplinary research environment.

I would like to thank Prof. Sadegh Soudjani for providing me a great opportunity to visit his laboratory at the School of Computing, Newcastle University, UK and also for the fruitful discussions with him during several meetings. The same thanks go to Prof. Antoine Girard for inviting me to the Signal and Systems Laboratory (L2S), CNRS, France. My thanks also go to his former Ph.D. student, my friend, Dr. Adnane Saoud for the collaborations and worthwhile discussions.

I would also like to thank Prof. George J. Pappas for hosting me during my research sabbatical at the University of Pennsylvania, PA, USA and for his valuable advice and his discussions with me. I also thank all of my colleagues at HyConSys lab *Mahmoud, Abdalla, Niloofar, Mahathi, Abolfazl, Ameneh, Matthias, Mahendra, Siyuan, Asad, Ali, and Christoph* for the inspiring discussions and the wonderful time we shared during the last four years.

Finally, I would like to thank my parents, my brother *Suraj*, and my beloved wife *Pooja* for their enduring support which brought me into the position of writing this thesis, and their sympathy and understanding when I had hard times.

Pushpak Jagtap
Munich, May 2020

Abstract

This dissertation is motivated by the challenges arising in the synthesis of controllers for complex systems enforcing complex logic specifications (usually expressed using temporal logic formulae or as (in)finite strings on automata). This thesis develops several controller synthesis approaches for various complex systems without discretizing state sets that helps us to alleviate the issue of the curse of dimensionality arises in conventional approaches discretizing state sets. The results of the thesis are divided into two main parts.

The first part of the thesis proposes a controller synthesis technique that provides finite abstractions for a class of infinite-dimensional stochastic control systems without state-space discretization under some incremental stability property. We also provide some preliminary results on the incremental stability property of (retarded) stochastic control systems. In particular, we propose a design of backstepping controllers rendering a class of stochastic control systems incrementally stable. In addition, we provide a characterization of incremental stability property for (retarded) stochastic control systems in terms of the existence of incremental Lyapunov functions. Finally, we present QUEST, a software tool developed in C++ for synthesizing controllers using finite abstractions obtained via the proposed state-space discretization-free approach.

The second part of the thesis deals with developing techniques to synthesize controllers that enforce complex logic specifications using discretization-free approaches based on notions of control barrier functions. The contribution of this part is threefold. First, we provide a systematic approach to synthesize a hybrid control policy for stochastic control systems enforcing a class of linear temporal logic specifications. It also provides two techniques to search for control barrier functions using sum-of-squares optimization and a counter-example guided inductive synthesis approach. Second, we develop a compositional framework for constructing control barrier functions for large-scale interconnected control systems. More specifically, the result provides a compositional controller synthesis scheme enforcing specifications expressed using co-Büchi automata by assuming some small-gain type conditions. In the last chapter, we provide a controller synthesis scheme for unknown dynamical systems. To provide the results, we use a data-driven approach utilizing Gaussian processes to learn the unknown model with some probabilistic guarantee on the accuracy of the learned model. Then, we utilize a notion of control barrier function to synthesize control policy for unknown systems enforcing specifications expressed using co-Büchi automata with some probabilistic guarantees.

To show the efficacy of the results proposed in the thesis, we consider various case studies such as controlled spring pendulum, temperature regulation in buildings, lane-keeping of a vehicle, a network of interconnected Kuramoto oscillators, and control of Moore-Greitzer jet engine model.

Zusammenfassung

Diese Dissertation ist motiviert durch die Herausforderungen, die sich bei der Synthese von Reglern für komplexe Systeme ergeben, die komplexe Spezifikationen erzwingen (welche üblicherweise durch Formeln der temporalen Logik oder (un)endliche Zeichenketten auf Automaten ausgedrückt werden). In der Tat werden in dieser Arbeit Ansätze zur Reglersynthese für verschiedene komplexe Systeme entwickelt, die nicht auf einer Diskretisierung des Zustandsraums basieren. Dadurch wird das Problem des Fluchs der Dimensionalität abgemildert, das sich bei konventionellen auf Diskretisierung basierenden Ansätzen ergibt.

Die Resultate dieser Arbeit sind in zwei Hauptteile gegliedert. Der erste Teil der Arbeit stellt eine Methode zur Reglersynthese vor, die für eine Klasse von unendlichdimensionalen stochastischen Kontrollsystemen unter Annahme der inkrementellen Stabilität endliche Abstraktionen ohne Zustandsdiskretisierung liefert. Wir präsentieren auch einige vorläufige Resultate über die inkrementelle Stabilitätseigenschaft von (zeitverzögerten) stochastischen Kontrollsystemen. Insbesondere stellen wir den Entwurf eines Backstepping-Reglers vor, der eine Klasse von stochastischen Kontrollsystemen inkrementell stabil macht, sowie die Charakterisierung der inkrementellen Stabilitätseigenschaft für (zeitverzögerte) stochastische Kontrollsysteme mittels inkrementeller Lyapunovfunktionen. Schließlich präsentieren wir QUEST, ein in C++ geschriebenes Software-Tool zur Synthese von Reglern, die endliche Abstraktionen nutzen, welche durch den diskretisieru-ngsfreien Ansatz erhalten werden.

Der zweite Teil der Arbeit beschäftigt sich mit der Entwicklung von Methoden zur Synthese von Reglern, die komplexe Spezifikationen erzwingen, wobei wir diskretisierungsfreie Methoden verwenden, die auf dem Begriff der Kontrollschrankenfunktion (engl. *control barrier function*) basieren. Dieser Teil der Arbeit liefert einen dreifachen Beitrag. Zunächst liefern wir einen systematischen Ansatz um eine hybride Reglerstrategie für stochastische Kontrollsysteme zu synthetisieren, die eine gewisse Klasse von temporalen Logikspezifikationen erzwingt. Er liefert auch zwei Methoden um nach parametrischen Kontrollschrankenfunktionen zu suchen unter Verwendung von Quadratsummenminimierung und einem von Gegenbeispielen geleiteten induktiven Syntheseansatz. Zum zweiten entwickeln wir ein kompositionelles Framework zur Konstruktion von Kontrollschrankenfunktionen für hochdimensionale vernetzte Kontrollsysteme. Etwas genauer liefert unser Resultat eine kompositionelle Reglersynthese, die unter der Annahme von Small-Gain-Bedingungen durch Co-Büchi-Automaten formulierte Spezifikationen erzwingt. Im letzten Kapitel präsentieren wir eine Reglersynthese für unbekannte dynamische Systeme. Um die Resultate zu beweisen, verwenden wir einen datengesteuerten Ansatz, der Gauß-Prozesse verwendet, um das unbekannte Modell mit einer gewissen probabilistischen Genauigkeitsgarantie zu erlernen. Dann verwenden wir den Begriff

der Kontrollschrankenfunktion, um Reglerstrategien für unbekannte Systeme zu synthetisieren, die Spezifikationen erzwingen, die durch Co-Büchi-Automaten mit gewissen probabilistischen Garantien ausgedrückt werden.

Um die Effizienz der in unserer Arbeit vorgestellten Resultate zu demonstrieren, betrachten wir verschiedene Fallstudien wie z.B. ein kontrolliertes Federpendel, Temperaturregelung in Gebäuden, Spurhalten eines Fahrzeugs, Netzwerke von Kuramoto-Oszillatoren und die Steuerung eines Moore-Greitzer-Triebwerkmodells.

Publications by the Author during Ph.D.

Journal Papers

1. A. Saoud*, **P. Jagtap***, M. Zamani, and A. Girard, "Compositional Abstraction and Controller Synthesis for Interconnected Systems: An Approximate Composition Approach," In: *the IEEE Transactions on Control of Network Systems*, 2021, in press.

2. **P. Jagtap**, F. Abdi, M. Rungger, M. Zamani, and M. Caccamo, "Software Fault Tolerance for Cyber-Physical Systems via Full System Restart," In: *the ACM Transaction on Cyber-Physical Systems*, vol. 4, no. 4, pp. 1-20, 2020.

3. N. Jahanshahi*, **P. Jagtap***, and M. Zamani, "Control Barrier Functions based Synthesis of Partially Observable Jump-Diffusion Systems," In: *the IEEE Control Systems Letter (L-CSS)*, vol. 5, no. 1, pp. 253-258, 2021.

4. **P. Jagtap**, S. Soudjani, and M. Zamani, "Formal Synthesis of Stochastic Systems via Control Barrier Certificates," In: *the IEEE Transactions on Automatic Control*, 2021, in press.

5. **P. Jagtap** and M. Zamani, "Symbolic Models for Retarded Jump-Diffusion System," In: *Automatica*, vol. 111, pp. 108666, 2020.

6. **P. Jagtap** and M. Zamani, "Backstepping Design for Incremental Stability of Stochastic Hamiltonian Systems with Jumps," In: *the IEEE Transactions on Automatic Control*, vol. 63, no. 1, pp. 255-261, 2018.

Book Chapters

7. **P. Jagtap**, S. Soudjani, and M. Zamani, "Temporal Logic Verification of Stochastic Systems Using Barrier Certificates," In: *the International Symposium on Automated Technology for Verification and Analysis (ATVA)*, LNCS 11138, pp. 177-193, Springer, 2018.

8. **P. Jagtap** and M. Zamani, "QUEST: A Tool for State-Space Quantization-free Synthesis of Symbolic Controllers," In: *the 14th International Conference on Quantitative Evaluation of SysTems (QEST),* Lecture Notes in Computer Science 10503, pp 309-313, Springer, 2017. `https://www.hcs.ei.tum.de/en/software/quest/`

* The authors contributed equally to this work.

Conference Papers

9. **P. Jagtap**, G. J. Pappas, and M. Zamani, "Control Barrier Functions Based Synthesis of Unknown Systems Using Gaussian Processes," In: *the IEEE Conference on Decision and Control (CDC)*, 2020, accepted for publication.

10. N. Jahanshahi*, **P. Jagtap***, and M. Zamani, "Synthesis of Stochastic Systems with Partial Information via Control Barrier Functions," In: *the 21st IFAC World Congress 2020*, .

11. **P. Jagtap***, A. Swikir*, and M. Zamani, "Compositional Construction of Control Barrier Functions for Interconnected Control Systems," In: *the 23rd ACM International Conference on Hybrid Systems: Computation and Control (HSCC)*, pp. 1-11, 2020.

12. P. Ashok, M. Jackermeier, **P. Jagtap**, J. Kretinsky, M. Weininger, and M. Zamani, "dtControl: Decision Tree Learning Algorithms for Controller Representation," In: *the 23rd ACM International Conference on Hybrid Systems: Computation and Control (HSCC)*, pp. 1-7, 2020.

13. M. Anand*, **P. Jagtap***, and M. Zamani, "Verification of Switched Stochastic Systems via Barrier Certificates," In: *the IEEE Conference on Decision and Control (CDC)*, pp. 4373-4378, 2019.

14. A. Saoud*, **P. Jagtap***, M. Zamani, and A. Girard, "Compositional Abstraction-Based Synthesis for Cascade Discrete-Time Control Systems," In: *the IFAC Conference on Analysis and Design of Hybrid Systems (ADHS)*, pp. 13-18, 2018.

15. **P. Jagtap** and M. Zamani, "On Incremental Stability of Time-Delayed Stochastic Control Systems," In: *the 54th Annual Allerton Conference on Communication, Control, and Computing*, pp. 577-581, 2016.

16. **P. Jagtap** and M. Zamani, "Backstepping Design for Incremental Stability of Stochastic Hamiltonian Systems," In: *the 55th IEEE Conference on Decision and Control (CDC)*, pp. 5367-5372, 2016.

* The authors contributed equally to this work.

Contents

List of Figures

List of Tables

List of Abbreviations

RJDS	retarded jump diffusion system
DJDS	delayed jump diffusion system
ISS	input-to-state stability
LMI	linear matrix inequality
BMI	bilinear matrix inequality
i.i.d.	independent identically distributed
LTL	linear temporal logic
LTL_F	linear temporal logic over finite traces
DFA	deterministic finite automata
DCA	deterministic co-Büchi automata
DBA	deterministic Büch automata
CBF	control barrier function
CEGIS	counter example guided inductive synthesis
SOS	sum-of-squares
SMT	satisfiability modulo theory
GP	Gaussian process

Notations

Number Sets

$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Set of positive real numbers
$\mathbb{R}_0^+$	Set of nonnegative real numbers
$\mathbb{N}:=\{1,2,3,\ldots\}$	Set of positive integers
$\mathbb{N}_0:=\{0,1,2,\ldots\}$	Set of nonnegative integers
$\mathbb{R}^{n\times m}$	Set of real matrices with n rows and m columns
I_n	Identity matrix in $\mathbb{R}^{n\times n}$
$0_{n\times m}$	Zero matrix in $\mathbb{R}^{n\times m}$
$\mathbf{1}_n$	Vector in $\mathbb{R}^n$ with all elements being one
e_i	Vector in $\mathbb{R}^n$ whose only i^{th} element is one and rest are zero
$\triangle$	Diagonal set defined as $\triangle := \{(x,x)\|x\in\mathbb{R}^n\}$

For $a,b\in\mathbb{R}$ with $a\leq b$ and $x,y\in\mathbb{N}_0$ with $x\leq y$, we use following notations:

$\|a\|$	Absolute value of a
$[a,b]$	Closed interval in $\mathbb{R}$
(a,b)	Open interval in $\mathbb{R}$
$[a,b)$, $(a,b]$	Half-open intervals in $\mathbb{R}$
$[x;y]$	Closed interval in $\mathbb{N}$
$(x;y)$	Open interval in $\mathbb{N}$
$[x;y)$, $(x;y]$	Half-open intervals in $\mathbb{N}$

Matrices and Vectors

Given matrices $A\in\mathbb{R}^{n\times m}$, $B\in\mathbb{R}^{n\times n}$, symmetric matrix $P\in\mathbb{R}^{n\times n}$, and vectors $x,y,z\in\mathbb{R}^n$, we have following notations:

$\{A\}_{ij}$	Individual element in A at ith row and jth column
A^T	Transpose of matrix A
$\mathsf{Tr}(B)$	Trace of matrix B
$\|A\|$	Euclidean norm of A
$\|A\|_F$	Frobenius norm of A defined as $\|A\|_F := \sqrt{\mathsf{Tr}(AA^T)}$
$A\otimes B$	Kronecker product of A and B
$\lambda_{\min}(P)$	Minimum eigenvalue of P

$\lambda_{\max}(P)$	Maximum eigenvalue of P
$\|x\|$	Euclidean norm of x
$\|x\|_\infty$	Infinity norm of x
$\mathbf{d}$	Metric on $\mathbb{R}^n$ mapping from $\mathbb{R}^n \times \mathbb{R}^n$ to $\mathbb{R}_0^+$ such that the following hold: (i) $\mathbf{d}(x,y) = 0$ if and only if $x = y$ (ii) $\mathbf{d}(x,y) = \mathbf{d}(y,x)$ (iii) $\mathbf{d}(x,z) \leq \mathbf{d}(x,y) + \mathbf{d}(y,z)$

Probability

$(\Omega, \mathcal{F}, \mathbb{P})$	Probability space with a sample space Ω, filtration $\mathcal{F}$, and the probability measure $\mathbb{P}$
$\mathbb{F} = (\mathcal{F}_s)_{s\geq 0}$	Filtration satisfying the usual conditions of right continuity and completeness [Øks02].
$(W_s)_{s\geq 0}$	$\check{r}$-dimensional $\mathbb{F}$-Brownian motion
$(P_s)_{s\geq 0}$	$\tilde{r}$-dimensional $\mathbb{F}$-Poison process
$\mathbb{E}$	Expectation operator
$\mathcal{N}(\mu, \Lambda)$	Multivariate normal distribution with mean μ and covariance matrix Λ

Functions

$\mathcal{C}(X, Y)$	Set of all continuous maps $f : X \to Y$ between metric spaces X and Y
$\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$	Set of all continuous maps $f : [-\tau, 0] \to \mathbb{R}^n$, for some $\tau \in \mathbb{R}_0^+$
$\mathcal{K}$	$\mathcal{K} := \{\alpha : \mathbb{R}_0^+ \to \mathbb{R}_0^+ \mid \alpha \text{ is continuous, strictly increasing, and } \alpha(0) = 0\}$
$\mathcal{K}_\infty$	$\mathcal{K}_\infty := \{\alpha \in \mathcal{K} \mid \lim_{r\to\infty} \alpha(r) = \infty\}$
$\mathcal{L}$	$\mathcal{L} := \{\alpha : \mathbb{R}_0^+ \to \mathbb{R}_0^+ \mid \alpha \text{ is strictly decreasing with } \lim_{r\to\infty} \alpha(r) = 0\}$
$\mathcal{KL}$	$\mathcal{KL} := \{\alpha : \mathbb{R}_0^+ \times \mathbb{R}_0^+ \to \mathbb{R}_0^+ \mid \alpha(\cdot, t) \in \mathcal{K}\ \forall t \in \mathbb{R}_0^+, \alpha(r, \cdot) \in \mathcal{L}\ \forall r > 0\}$
$\mathcal{I}_d$	Identity function $\mathcal{I}_d \in \mathcal{K}_\infty$ defined as $\mathcal{I}_d(r) = r\ \forall r \in \mathbb{R}_0^+$
$\|\zeta\|_{[-\tau,0]}$	$\|\zeta\|_{[-\tau,0]} := \sup_{-\tau\leq\theta\leq 0} \|\zeta(\theta)\|$ for $\zeta \in \mathcal{C}([-\tau, 0], \mathbb{R}^n)$
$\|f\|_\infty$	(essential) supremum of $f : \mathbb{R}_0^+ \to \mathbb{R}^n$ defined as (ess)sup$\{\|f(t)\|, t \geq 0\}$
$\mathcal{C}^b_{\mathcal{F}_0}([-\tau, 0]; \mathbb{R}^n)$	Set of all bounded $\mathcal{F}_0$-measurable $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$-valued random variables
$\mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau, 0]; \mathbb{R}^n)$	Set of all $\mathcal{F}_t$-measurable $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$-valued random processes $\phi := \{\phi(\theta) \mid -\tau \leq \theta \leq 0\}$ such that $\sup_{-\tau\leq\theta\leq 0} \mathbb{E}[\|\phi(\theta)\|^{\mathfrak{m}}] < \infty$ for $\mathfrak{m} > 0$ and $t \in \mathbb{R}_0^+$

Sets

Given sets X and Y, we have following notations:

$X \cap Y$	Intersection of sets X and Y
$X \cup Y$	Union of sets X and Y
$\|X\|$	Cardinality of the set X
2^X	Power set of the set X

X^c	Complement of the set X
$X \setminus Y$	X setminus Y defined as $\{x : x \in X, x \notin Y\}$
X^*	Set of all finite strings over X
X^ω	Set of all infinite strings over X
$\emptyset$	Empty set

Other

The closed ball centered at $x \in \mathbb{R}^m$ with radius R is defined by $\mathcal{B}_{\mathsf{R}}(x) = \{y \in \mathbb{R}^m \mid \|x-y\| \leq \mathsf{R}\}$. A set $B \subseteq \mathbb{R}^m$ is called a *box* if $B = \prod_{i=1}^m [c_i, d_i]$, where $c_i, d_i \in \mathbb{R}$ with $c_i < d_i$ for each $i \in [1; m]$. The *span* of a box B is defined as $span(B) = \min\{|d_i - c_i| \mid i = 1, \ldots, m\}$, where $|a|$ represents the absolute value of $a \in \mathbb{R}$. By defining $[\mathbb{R}^m]_\eta := \{z \in \mathbb{R}^m \mid z_i = \frac{k_i \eta}{\sqrt{m}}, k_i \in \mathbb{Z}\}$, the set $\bigcup_{b \in [\mathbb{R}^m]_\eta} \mathcal{B}_R(b)$ is a countable covering of $\mathbb{R}^m$ for any $\eta \in \mathbb{R}^+$ and $R \geq \eta/2$. For a box $B \subseteq \mathbb{R}^m$ and $\eta \leq span(B)$, define the η-approximation $[B]_\eta := [\mathbb{R}^m]_\eta \cap B$. Note that $[B]_\eta \neq \varnothing$ for any $\eta \leq span(B)$. We extend the notions of *span* and of approximation to finite unions of boxes as follows. Let $A = \bigcup_{j=1}^M A_j$, where each A_j is a box. Define $span(A) = \min\{span(A_j) \mid j = 1, \ldots, M\}$, and for any $\eta \leq span(A)$, define $[A]_\eta = \bigcup_{j=1}^M [A_j]_\eta$.

$\top$	`true`
$\bot$	`false`
$\vee$	Logical OR operator
$\wedge$	Logical AND operator
$\neg$	Negation operator

1 Introduction

1.1 Motivation and Contributions

Nowadays many real-world applications are expected to do complex logic tasks. Consider an example of Air Delivery Drone by Amazon, it needs to do complex logic tasks of the form "pick up a package from location A and deliver it to location B within certain time interval T while avoiding obstacles O and return to the charging station C whenever its battery is low." Such complex logic tasks can usually be expressed using temporal logic formulae or as (in)finite strings on automata [BKL08]. Formal synthesis of controllers enforcing such complex tasks has gained significant attention in the last decade. For solving such formal synthesis problems, discrete abstractions (a.k.a. symbolic models) based techniques [Tab09, BYG17, and references therin] are very popular since they provide tools for automated, correct-by-construction, controller synthesis for several classes of control systems. In particular, such abstractions provide (approximate) finite models that are related to concrete systems by aggregating concrete states and inputs to the symbolic ones. Having such finite abstractions, one can make use of the existing automata-theoretic techniques [MPS95] to synthesize hybrid controllers enforcing rich complex specifications over the original systems. In the past few years, there have been several results providing discrete abstractions for various non-stochastic as well as stochastic systems (See [Tab09, BYG17, RWR17, ZPMT11, PGT08, PPDBT10, GPT09, ZA14, ZMEM$^+$14, and references therein]). However, the abstractions obtained in these results are based on state-space quantization and suffer severely from the *curse of dimensionality*, i.e., the computational complexity increases exponentially with respect to the state-space dimension of the concrete system.

To alleviate the issue of the curse of dimensionality, under some stability property, namely, incremental stability [Ang02], the authors in [LCGG13, ZAG15, ZTA16] proposed an alternative approach for constructing finite abstractions for various classes of systems without discretizing the state-space. Alternatively, there are several results that proposed compositional construction of discrete abstractions for non-stochastic as well as stochastic systems using various approaches such as small-gain conditions [RZ16a, SZ19a] and dissipativity [ZA17, Lav19]. However, most of these techniques are restricted to some stability assumptions and are unable to solve controller synthesis problems beyond safety specifications. Apart from this, the approaches based on barrier functions [PJP07, AXGT17] have shown great potentials in solving controller synthesis problems without any discretization. Assuming a prior knowledge of control barrier functions, several techniques have been recently introduced to ensure the safety of non-stochastic dynamical systems (see [AXGT17, ACE$^+$19, and the references therein]).

Though promising, to the best of our knowledge, there are no results available in the literature to utilize the notion of barrier functions to synthesize controllers for more complex logic specifications.

On the other hand, engineering applications are becoming more complex due to many factors. Some of them include noisy dynamics, dependency on state history, lack of knowledge of the exact mathematical model, and interconnection between subsystems. In the presence of such complexities in dynamics, the aforementioned synthesis problem becomes much more challenging and difficult to solve. Motivated by the above results and their limitations, this thesis focuses on:

(i) developing state-space discretization-free controller synthesis techniques enforcing complex specifications to alleviate the issue of the curse of dimensionality and

(ii) providing formally verified controller synthesis techniques for the systems posing various complexities such as noisy dynamics, dependency on state history, lack of knowledge of the exact mathematical model, and interconnection between subsystems.

In the first part of the thesis, we provide a construction of discrete abstractions for a class of infinite-dimensional stochastic control systems, namely, retarded jump-diffusion systems, without any state-space discretization under some incremental stability property. As incremental stability is the key property in the construction of discrete abstractions, we provide sufficient conditions for checking this notion of incremental stability for (retarded) stochastic control systems in terms of a notion of incremental Lyapunov functions. In addition, we provide a backstepping control scheme rendering a class of stochastic control systems, namely, stochastic Hamiltonian systems with jumps, incrementally stable. Finally, we present QUEST, an open-source software tool for automated synthesis of state-space discretization-free discrete abstractions and controllers for safety and reachability specifications.

In the second part of the thesis, for the first time, we provide results extending the use of so-called control barrier functions for the synthesis of controllers enforcing complex specifications for stochastic control systems. In particular, the results provide a systematic way to design hybrid control policies for stochastic control systems that provides the formal probabilistic guarantee on the satisfaction of temporal logic specifications over a finite-time interval. We also provide systematic numerical approaches to search for those control barrier functions. To cope with the issue of scalability due to the numerical search of control barrier functions, this part proposes a compositional construction of control barrier function by considering a large-scale system as an interconnection of different small subsystems which also enables us to design a decentralized (or distributed) hybrid control policy enforcing complex specifications. Lastly, we provide similar results considering systems with another most common complexity where the system dynamics are unknown.

1.2 Outline of the Thesis

The thesis is divided into three parts consisting of eight chapters in total.

- Part I presents the results on a construction of state-space discretization-free finite abstractions based symbolic controllers for a class of incrementally stable infinite-dimensional systems the with incremental stability analysis of various stochastic systems.

- Part II presents the discretization-free controller synthesis approaches based on control barrier functions for various classes of systems and for enforcing complex specifications.

In the remainder, we discuss the chapter-wise organization of the thesis.

Chapter 2 presents some results on a notion of incremental stability of (retarded) stochastic control systems which serves as a key element for providing the main result in the first part of the thesis. The results of this chapter are presented based on [JZ16a, JZ17a, JZ16b, JZ19].

Under the assumption of incremental stability property, **Chapter 3** discusses a state-space discretization-free construction of finite abstractions for a class of infinite-dimensional stochastic systems, namely, retarded jump-diffusion systems. The result of this chapter is based on [JZ19].

Chapter 4 presents a software tool, QUEST, for an automated synthesis of correct-by-construction controllers using state-space discretization-free abstractions. The interested readers are referred to our tool paper [JZ17b] and tool manual at `www.hcs.ei.tum.de/software` for a more extensive discussions.

Chapter 5 provides a discretization-free approach based on the notion of control barrier functions for the synthesis of hybrid control policies for stochastic control systems enforcing complex logic specifications. It also provides systematic approaches for searching for control barrier functions using sum-of-squares optimization and counterexample guided inductive synthesis. The results of this chapter are presented based on [JSZ18, JSZ20a].

Chapter 6 discusses the synthesis of hybrid controllers for large-scale systems enforcing complex specifications via compositional construction of control barrier functions. In particular, the result helps us to alleviate the issue of computational complexity in the search algorithm proposed in Chapter 5 due to the increase in the number of search parameters of the barrier functions while dealing with large-scale systems. The results of this chapter are based on [JSZ20b].

Chapter 7 provides a formal synthesis of controllers for unknown dynamical systems enforcing complex specifications. In this chapter, we provide the synthesis of controllers by utilizing control barrier functions constructed using a model learned via Gaussian processes. The results of this chapter are based on [JPZ20].

In **Chapter 8**, we summarize the results of this thesis and outlines potential directions for future research.

For more clarity of exposition, Chapters 2 - 7 follow a common structure. Each chapter starts with a brief discussion about the chapter. Then continues with the introduction including a brief literature review and a list of the contributions made. The developed techniques are discussed in subsequent sections, followed by a section illustrating their efficiency on different case studies.

Part I

State-Space Discretization-free Abstractions for Controller Synthesis

2 Preliminary Results on Incremental Stability

This chapter presents some preliminary results on the incremental input-to-state stability property of (retarded) stochastic control systems. Here, we introduce a notion of incremental stability for stochastic control systems and retarded jump-diffusion systems. Moreover, we provide sufficient conditions for the proposed notions of incremental stability in terms of the existence of incremental Lyapunov functions. We also provide a backstepping controller design scheme providing controllers along with corresponding incremental Lyapunov functions rendering a class of stochastic control systems, namely, stochastic Hamiltonian systems with jumps, incrementally stable.

2.1 Introduction

The notion of incremental stability [Ang02] focuses on the convergence of trajectories with respect to each other rather than with respect to an equilibrium point or a fixed trajectory. This notion of stability has gained significant attention in recent years due to its potential applications in the study of nonlinear systems. Examples of such applications include synchronization of cyclic feedback systems [HSSG12], construction of symbolic models [PGT08, MZ12], modeling of nonlinear analog circuits [BML$^+$10], and synchronization of interconnected oscillators [SS07].

In this thesis, the main motivation behind investigating the incremental stability property lies in its usefulness for the construction of symbolic models and, hence, automated controller synthesis methodologies. Some of the benefits of incremental stability properties are listed below:

- It enables the construction of (approximate) bisimilar finite abstractions of non-probabilistic [GPT09, MZ12, PGT08] as well as stochastic systems [ZMEM$^+$14, ZAG15, ZA14, JZ19].
- It helps to alleviate the issue of the curse of dimensionality due to state-space discretization in conventional methods for the construction of finite abstractions by providing abstractions based on input sequences (see [LCGG13, ZAG15, ZTA16] for more details).
- It allows the construction of abstractions for infinite dimensional systems [Gir14, JZ19].

Motivating from the aforementioned benefits, we provide some preliminary results in this chapter which then will be utilized in the next chapter to provide state-space discretization-free abstractions for a class of infinite-dimensional stochastic control systems, namely, retarded jump-diffusion systems.

2.1.1 Related Literature

Incremental stability for non-probabilistic systems

In the past few years, there have been several results characterizing a notion of incremental stability for non-probabilistic dynamical systems using notions of so-called incremental Lyapunov functions and contraction metrics. The interested readers may consult the results in [Ang02, PWN06, LS98, ZT11, ZvdWM13, and references therein] for more detailed information about different characterizations of incremental stability. Furthermore, there have been several results on the construction of state feedback controllers enforcing a class of non-probabilistic control systems incrementally stable. Examples include results on smooth strict-feedback form systems [ZT11] and a class of (not necessarily smooth) control systems [ZvdWM13].

Incremental stability for stochastic systems

In recent years, similar notions of incremental stability have been introduced for different classes of stochastic systems including stochastic control systems [ZMEM$^+$14], stochastic switched systems [ZAG15], randomly switched stochastic systems [ZA14], and their descriptions using some notions of incremental Lyapunov functions. In addition, there have been several results in the literature studying the incremental stability of stochastic systems using a notion of contraction metric. Examples include the results on stochastic dynamical systems [PTS09] and a class of stochastic hybrid systems [ZCA13].

Incremental stability for retarded/delayed stochastic systems

There are very few results available in the literature on the description of incremental stability for deterministic time-delayed systems, notably, using Lyapunov-Krasovskii functionals [PPDBT10] and Razumikhin-Lyapunov approach [CPR13]. However, to the best of our knowledge, there is no work available in the literature on the analysis of incremental stability for time-delayed stochastic control systems except our results [JZ16b, JZ19] which are discussed in this chapter.

2.1.2 Contributions

In the first part of the chapter, we introduce a coordinate invariant notion of incremental stability for stochastic control systems with jumps and provide its description in terms of the existence of a notion of so-called incremental Lyapunov functions. Then, we provide a feedback controller design approach based on backstepping scheme providing controllers together with the corresponding incremental Lyapunov functions enforcing a

class of stochastic control systems, namely, stochastic Hamiltonian systems with jumps, incrementally stable.

In the second part of the chapter, we introduce a notion of incremental input-to-state stability for retarded jump-diffusion systems and provide sufficient conditions for it in terms of the existence of a notion of incremental Lyapunov functions. In the linear case, we show that the sufficient conditions reduce to a matrix inequality.

2.2 Backstepping Design for Incremental Stability of Stochastic Control Systems

In this section, we consider stochastic control systems defined formally as follows.

2.2.1 Stochastic Control Systems

Definition 2.2.1. *A stochastic control system is a tuple* $\Sigma_s = (\mathbb{R}^n, \mathbb{R}^m, \mathcal{U}, f, g, r)$*, where:*

- $\mathbb{R}^n$ *is the state space;*
- $\mathbb{R}^m$ *is the input space;*
- $\mathcal{U}$ *is a subset of the set of all* $\mathbb{F}$*-progressively measurable processes with values in* $\mathbb{R}^m$*; see [KS91, Def. 1.11];*
- $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ *satisfies the following Lipschitz assumption: there exist constants* $L_f, L_u \in \mathbb{R}^+$ *such that:* $\|f(x,u) - f(\hat{x},\hat{u})\| \leq L_f\|x - \hat{x}\| + L_u\|u - \hat{u}\|$ $\forall x, \hat{x} \in \mathbb{R}^n$ *and* $\forall u, \hat{u} \in \mathbb{R}^m$*;*
- $g : \mathbb{R}^n \to \mathbb{R}^{n\times\check{r}}$ *satisfies the following Lipschitz assumption: there exists a constant* $L_g \in \mathbb{R}_0^+$ *such that:* $\|g(x) - g(\hat{x})\| \leq L_g\|x - \hat{x}\|$ $\forall x, \hat{x} \in \mathbb{R}^n$*;*
- $r : \mathbb{R}^n \to \mathbb{R}^{n\times\tilde{r}}$ *satisfies the following Lipschitz assumption: there exists a constant* $L_r \in \mathbb{R}_0^+$ *such that:* $\|r(x) - r(\hat{x})\| \leq L_r\|x - \hat{x}\|$ $\forall x, \hat{x} \in \mathbb{R}^n$*.*

A stochastic process $\xi : \Omega \times \mathbb{R}_0^+ \to \mathbb{R}^n$ is said to be a *solution process* of Σ_s if there exists $\upsilon \in \mathcal{U}$ satisfying

$$\mathrm{d}\,\xi = f(\xi, \upsilon)\,\mathrm{d}\,t + g(\xi)\,\mathrm{d}\,W_t + r(\xi)\,\mathrm{d}\,P_t, \tag{2.2.1}$$

$\mathbb{P}$-almost surely ($\mathbb{P}$-a.s.), where f, g, and r are the drift, diffusion, and reset terms, respectively. We emphasize that postulated assumptions on f, g, and r ensure the existence and uniqueness of the solution process [ØS05]. Throughout this section, we use the notation $\xi_{a\upsilon}(t)$ to denote the value of a solution process at time $t \in \mathbb{R}_0^+$ under the input signal υ and with initial condition $\xi_{a\upsilon}(0) = a$ $\mathbb{P}$-a.s., in which a is a random variable that is measurable in $\mathcal{F}_0$. Here, we assume that the Poisson process and the Brownian motion are independent of each other. The Poisson process $P_s := [P_s^1; \ldots; P_s^{\tilde{r}}]$ models $\tilde{r}$ kinds of events whose occurrences are assumed to be independent of each other and have the constant rates of λ_i for each P_s^i, $i \in [1; \tilde{r}]$.

2.2.2 Incremental Stability for Stochastic Control Systems

This subsection introduces a coordinate invariant notion of incremental input-to-state stability for stochastic control systems. The stability notion discussed here is the generalization of the ones defined in [ZvdWM13], [ZT11] for non-probabilistic control systems.

Definition 2.2.2. *A stochastic control system Σ_s is incrementally input-to-state stable ($\delta_\exists$-ISS-$M_\mathfrak{m}$) in the $\mathfrak{m}^{th}$ moment, where $\mathfrak{m} \geq 1$, if there exist a metric $\mathbf{d}$, a function $\beta \in \mathcal{KL}$, and a function $\gamma \in \mathcal{K}_\infty$ such that for any $t \in \mathbb{R}_0^+$, any $\mathbb{R}^n$-valued random variables a and $\hat{a}$ that are measurable in $\mathcal{F}_0$, and any $\upsilon, \hat{\upsilon} \in \mathcal{U}$, the following condition is satisfied:*

$$\mathbb{E}[(\mathbf{d}(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t)))^\mathfrak{m}] \leq \beta(\mathbb{E}[(\mathbf{d}(a,\hat{a}))^\mathfrak{m}], t) + \gamma(\mathbb{E}[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}]). \tag{2.2.2}$$

Remark 2.2.3. *Note that if one uses the natural Euclidean metric rather than a general metric $\mathbf{d}$ in Definition 2.2.2, the notion reduces to the one defined in [ZMEM+14, Definition 3.1] which is not invariant under changes of coordinates. Observer that changes of coordinates are one of the main tools used in the backstepping design scheme including the one proposed in this section.*

One can describe $\delta_\exists$-ISS-$M_\mathfrak{m}$ in terms of existence of $\delta_\exists$-ISS-$M_\mathfrak{m}$ Lyapunov functions as defined next.

Definition 2.2.4. *Consider a stochastic control system Σ_s and a continuous function $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_0^+$ that is twice continuously differentiable on $\mathbb{R}^n \times \mathbb{R}^n \backslash \triangle$. The function V is called a $\delta_\exists$-ISS-$M_\mathfrak{m}$ Lyapunov function for Σ_s, if it has polynomial growth rate and there exist a metric $\mathbf{d}$, functions $\underline{\alpha}, \overline{\alpha}, \overline{\varphi} \in \mathcal{K}_\infty$, and a constant $\kappa \in \mathbb{R}^+$, such that:*

(i) *$\underline{\alpha}$(resp. $\overline{\alpha}$ and $\overline{\varphi}$) is a convex (resp. concave) function;*

(ii) *$\forall x, \hat{x} \in \mathbb{R}^n$, $\underline{\alpha}((\mathbf{d}(x,\hat{x}))^\mathfrak{m}) \leq V(x,\hat{x}) \leq \overline{\alpha}((\mathbf{d}(x,\hat{x}))^\mathfrak{m})$;*

(iii) *$\forall x, \hat{x} \in \mathbb{R}^n, x \neq \hat{x}$, and $\forall u, \hat{u} \in \mathbb{R}^m$,*

$$\begin{aligned}
\mathcal{D}V(x,\hat{x}) :=& \begin{bmatrix} \partial_x V(x,\hat{x}) & \partial_{\hat{x}} V(x,\hat{x}) \end{bmatrix} \begin{bmatrix} f(x,u) \\ f(\hat{x},\hat{u}) \end{bmatrix} \\
&+ \frac{1}{2}\mathsf{Tr}\left(\begin{bmatrix} g(x) \\ g(\hat{x}) \end{bmatrix} \begin{bmatrix} g^T(x) & g^T(\hat{x}) \end{bmatrix} \begin{bmatrix} \partial_{x,x} V & \partial_{x,\hat{x}} V \\ \partial_{\hat{x},x} V & \partial_{\hat{x},\hat{x}} V \end{bmatrix} \right) \\
&+ \sum_{i=1}^{\tilde{r}} \lambda_i \big(V(x + r(x)e_i, \hat{x} + r(\hat{x})e_i) - V(x,\hat{x}) \big) \\
\leq& -\kappa V(x,\hat{x}) + \overline{\varphi}(\|u - \hat{u}\|^\mathfrak{m}),
\end{aligned}$$

where $\mathcal{D}$ is the infinitesimal generator of the stochastic process ξ in (2.2.1) acting on function V [JP09, equation (23)] and the symbols ∂_x and $\partial_{x,\hat{x}}$ represents first and second-order partial derivatives with respect to x and $\hat{x}$, respectively. The following theorem describes $\delta_\exists$-ISS-$M_\mathfrak{m}$ in terms of existence of $\delta_\exists$-ISS-$M_\mathfrak{m}$ Lyapunov functions.

Theorem 2.2.5. *A stochastic control system Σ_s is $\delta_\exists$-ISS-$M_\mathfrak{m}$ if it admits a $\delta_\exists$-ISS-$M_\mathfrak{m}$ Lyapunov function.*

Proof. For any time instance $t \geq 0$, any $\upsilon, \hat{\upsilon} \in \mathcal{U}$, and any random variable a and $\hat{a}$ that are $\mathcal{F}_0$-measurable, one obtains

$$\begin{aligned}
\mathbb{E}\left[V(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t))\right] &= \mathbb{E}\Big[V\big(\xi_{a\upsilon}(0), \xi_{\hat{a}\hat{\upsilon}}(0)\big) + \int_0^t \mathcal{D}V(\xi_{a\upsilon}(s), \xi_{\hat{a}\hat{\upsilon}}(s))\,\mathrm{d}\,s\Big] \\
&\leq \mathbb{E}\left[V\big(\xi_{a\upsilon}(0), \xi_{\hat{a}\hat{\upsilon}}(0)\big)\right] + \mathbb{E}\Big[\int_0^t \big(-\kappa V\big(\xi_{a\upsilon}(s), \xi_{\hat{a}\hat{\upsilon}}(s)\big) + \overline{\varphi}(\|\upsilon(s) - \hat{\upsilon}(s)\|^\mathfrak{m})\big)\,\mathrm{d}\,s\Big] \\
&\leq \mathbb{E}\left[V\big(\xi_{a\upsilon}(0), \xi_{\hat{a}\hat{\upsilon}}(0)\big)\right] + \int_0^t \Big(-\kappa\mathbb{E}\left[V\big(\xi_{a\upsilon}(s), \xi_{\hat{a}\hat{\upsilon}}(s)\big)\right] + \mathbb{E}\Big[\overline{\varphi}(\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m})\Big]\Big)\,\mathrm{d}\,s,
\end{aligned}$$

where the first equality is an application of the Itô's formula for jump diffusions thanks to the polynomial rate of the function V [ØS05, Theorem 1.24] and the first inequality is because of condition iii) in Definition 2.2.4. By virtue of Gronwall's inequality, one obtains

$$\begin{aligned}
\mathbb{E}[V(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t))] &\leq \mathbb{E}[V(a, \hat{a})]\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\mathbb{E}\big[\overline{\varphi}(\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}) \\
&\leq \mathbb{E}[V(a, \hat{a})]\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big), \qquad (2.2.3)
\end{aligned}$$

where the last inequality follows from Jensen's inequality due to the concavity assumption on the function $\overline{\varphi}$ [Øks02, page 310]. In view of Jensen's inequality, inequality (2.2.3), the convexity of $\underline{\alpha}$, the concavity of $\overline{\alpha}$, and condition ii) in Definition 2.2.4, we have the following chain of inequalities

$$\begin{aligned}
&\underline{\alpha}\big(\mathbb{E}\big[(\mathbf{d}(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t)))^\mathfrak{m}\big]\big) \leq \mathbb{E}\big[\underline{\alpha}\big((\mathbf{d}(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t)))^\mathfrak{m}\big)\big] \leq \mathbb{E}\left[V(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t))\right] \\
&\leq \mathbb{E}[V(a, \hat{a})]\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big) \leq \mathbb{E}\big[\overline{\alpha}\big((\mathbf{d}(a, \hat{a}))^\mathfrak{m}\big)\big]\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big) \\
&\leq \overline{\alpha}\big(\mathbb{E}\big[(\mathbf{d}(a, \hat{a}))^\mathfrak{m}\big]\big)\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big),
\end{aligned}$$

which in conjunction with the fact that $\underline{\alpha} \in \mathcal{K}_\infty$ leads to

$$\begin{aligned}
\mathbb{E}\big[(\mathbf{d}(\xi_{a\upsilon}(t), \xi_{\hat{a}\hat{\upsilon}}(t)))^\mathfrak{m}\big] &\leq \underline{\alpha}^{-1}\Big(\overline{\alpha}\big(\mathbb{E}\big[(\mathbf{d}(a, \hat{a}))^\mathfrak{m}\big]\big)\mathsf{e}^{-\kappa t} + \frac{1}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big)\Big) \\
&\leq \underline{\alpha}^{-1}\big(2\overline{\alpha}\big(\mathbb{E}\big[(\mathbf{d}(a, \hat{a}))^\mathfrak{m}\big]\big)\mathsf{e}^{-\kappa t}\big) + \underline{\alpha}^{-1}\Big(\frac{2}{\kappa}\overline{\varphi}\big(\mathbb{E}\big[\|\upsilon - \hat{\upsilon}\|_\infty^\mathfrak{m}\big]\big)\Big).
\end{aligned}$$

Therefore, by introducing functions β and γ as

$$\beta(s, t) := \underline{\alpha}^{-1}\big(2\overline{\alpha}(y)\mathsf{e}^{-\kappa t}\big), \quad \gamma(s) := \underline{\alpha}^{-1}\Big(\frac{2}{\kappa}\overline{\varphi}(s)\Big), \qquad (2.2.4)$$

for any $s, t \in \mathbb{R}_0^+$, inequality (2.2.2) is satisfied. Note that if $\underline{\alpha}^{-1}$ satisfies the triangle inequality (i.e., $\underline{\alpha}^{-1}(a + b) \leq \underline{\alpha}^{-1}(a) + \underline{\alpha}^{-1}(b)$), one can remove the coefficients 2 in the expressions of β and γ in (2.2.4) to get a less conservative upper bound in (2.2.2). □

2.2.3 Backstepping Design Procedure

This subsection contains the main contribution of the section. Here, we propose a backstepping control design scheme for a class of stochastic control systems, namely, stochastic Hamiltonian systems with jumps. The proposed methodology provides controllers rendering the closed loop system $\delta_{\exists}$-ISS-$\mathrm{M}_{\mathfrak{m}}$. A stochastic Hamiltonian system with jumps is a stochastic control system $\Sigma_s = (\mathbb{R}^{2n}, \mathbb{R}^n, \mathcal{U}, f, g, r)$ described by stochastic differential equations

$$\Sigma_s : \begin{cases} \mathrm{d}\,\mathsf{q} = \partial_p \mathcal{H}(\mathsf{q},\mathsf{p})\,\mathrm{d}\,t, \\ \mathrm{d}\,\mathsf{p} = \Big(-\partial_q \mathcal{H}(\mathsf{q},\mathsf{p}) + b(\mathsf{q},\mathsf{p}) + G(\mathsf{q})\upsilon \Big)\,\mathrm{d}\,t + g(\mathsf{q})\,\mathrm{d}\,W_t + r(\mathsf{q})\,\mathrm{d}\,P_t, \end{cases} \tag{2.2.5}$$

where $q = \mathsf{q}(w,t) \in \mathbb{R}^n$, $\forall t \in \mathbb{R}_0^+$ and $\forall w \in \Omega$, is a generalized coordinate vector of n-degree-of-freedom system; $p = \mathsf{p}(w,t) \in \mathbb{R}^n$, $\forall t \in \mathbb{R}_0^+$ and $\forall w \in \Omega$, represents a vector of generalized momenta and defined as $\mathsf{p}\,\mathrm{d}\,t = M(\mathsf{q})\,\mathrm{d}\,\mathsf{q}$, where $M(q)$ is a symmetric, nonsingular, and positive definite inertia matrix; $b(q,p)$ is a smooth damping term; $G(q)\upsilon$ is the control force caused by $G(q)$, a nonsingular smooth square matrix, and by control input υ acting on the system; $g(q)$ is the diffusion term; $r(q)$ is the reset term capturing the magnitude of jumps; and $\partial_q \mathcal{H}$ and $\partial_p \mathcal{H}$ represent first order partial derivative of function $\mathcal{H}$ with respect to q and p, respectively, where $\mathcal{H}$ is a continuously differentiable Hamiltonian function represented in terms of total energy of the system as the following

$$\mathcal{H}(q,p) = \frac{1}{2} p^T M^{-1}(q) p + \mathfrak{N}(q), \tag{2.2.6}$$

where $\mathfrak{N}(q)$ represents potential energy of the system. By substituting (2.2.6) into (2.2.5), the dynamics of Σ_s can be rewritten as

$$\Sigma_s : \begin{cases} \mathrm{d}\,\mathsf{q} = M^{-1}(\mathsf{q})\mathsf{p}\,\mathrm{d}\,t, \\ \mathrm{d}\,\mathsf{p} = \Big(\mathfrak{N}(\mathsf{q},\mathsf{p}) + G(\mathsf{q})\upsilon \Big)\,\mathrm{d}\,t + g(\mathsf{q})\,\mathrm{d}\,W_t + r(\mathsf{q})\,\mathrm{d}\,P_t, \end{cases} \tag{2.2.7}$$

where $\mathfrak{N}(\mathsf{q},\mathsf{p}) = -\partial_q \mathcal{H}(\mathsf{q},\mathsf{p}) + b(\mathsf{q},\mathsf{p})$.

Remark 2.2.6. *Note that the dynamic considered in (2.2.7) is the generalization of the ones given in [WCS12] and [Iwa16]. It extends the former by including the jump term and the latter by adding the diffusion term and representing more general stochastic Hamiltonian systems.*

As we already emphasized after Definition 2.2.1, in order to ensure the existence and uniqueness of the solution process of Σ_s in (2.2.7), one requires a Lipschitz assumption on the drift term which implies:

$$\|M^{-1}(q)p - M^{-1}(\hat{q})\hat{p}\| \leq L_1\|q - \hat{q}\| + L_2\|p - \hat{p}\|, \tag{2.2.8}$$

for some $L_1, L_2 \in \mathbb{R}^+$ and any $q, \hat{q}, p, \hat{p} \in \mathbb{R}^n$.

We can now state the main result of the section on the backstepping controller design scheme providing controllers rendering the considered class of stochastic control systems $\delta_{\exists}$-ISS-$\mathrm{M}_{\mathfrak{m}}$ for any $\mathfrak{m} \geq 2$.

Theorem 2.2.7. *Consider the stochastic control system Σ_s of the form (2.2.7). The state feedback control law*

$$\begin{aligned} \upsilon =& G^{-1}(\mathsf{q})\Big(-\mathfrak{N}(\mathsf{q},\mathsf{p}) - \kappa_1 \frac{\mathrm{d}\, M(\mathsf{q})}{\mathrm{d}\, t}\mathsf{q} + \kappa_1^2 M(\mathsf{q})\mathsf{q} \\ & - \Big(2\kappa_1 + \lambda\frac{(2^{\mathfrak{m}-1}-1)}{\mathfrak{m}} + \frac{L_2}{s_1\varepsilon_1^{s_1}} + \frac{\min\{n,\check{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)\big(\mathsf{p}+\kappa_1 M(\mathsf{q})\mathsf{q}\big) + \hat{\upsilon}\Big), \end{aligned} \tag{2.2.9}$$

renders the closed-loop stochastic control system Σ_s $\delta_\exists$-ISS-$M_\mathfrak{m}$ for $\mathfrak{m} > 2$ with respect to input $\hat{\upsilon}$, for all

$$\kappa_1 > L_1 + \frac{\max\{L_2,1\}\varepsilon_1^{r_1}}{r_1} + \frac{\min\{n,\check{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2} + \frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}},$$

where $r_1 = \frac{\mathfrak{m}}{\mathfrak{m}-1}$, $s_1 = \mathfrak{m}$, $r_2 = \frac{\mathfrak{m}}{\mathfrak{m}-2}$, $s_2 = \frac{\mathfrak{m}}{2}$, ε_1 and ε_2 are positive constants which can be chosen arbitrarily, $\lambda = \sum_{i=1}^{\tilde{r}}\lambda_i$, and L_1, L_2, L_g, and L_r are the Lipschitz constants introduced in (2.2.8) and Definition 2.2.1, respectively.

Note that the term $\frac{\mathrm{d}\, M(\mathsf{q})}{\mathrm{d}\, t}$ in the control law (2.2.9) can be computed by using the definition of the derivative of matrix [WCS12] as

$$\frac{\mathrm{d}\, M(\mathsf{q})}{\mathrm{d}\, t} = \frac{\partial M(\mathsf{q})}{\partial q^T} \times \Big(\frac{\mathrm{d}\,\mathsf{q}}{\mathrm{d}\, t} \otimes I_n\Big) = \frac{\partial M(\mathsf{q})}{\partial q^T} \times \big(M^{-1}(\mathsf{q})\mathsf{p} \otimes I_n\big),$$

where $\frac{\partial M(\mathsf{q})}{\partial q^T} := \Big[\frac{\partial M(\mathsf{q})}{\partial q_1}\ \frac{\partial M(\mathsf{q})}{\partial q_2} \cdots \frac{\partial M(\mathsf{q})}{\partial q_n}\Big]_{n\times n^2}$.

Proof. Consider a coordinate transformation as

$$\tilde{\zeta} = \psi(\xi) = \begin{bmatrix}\tilde{\zeta}_1 \\ \tilde{\zeta}_2\end{bmatrix} = \begin{bmatrix}\mathsf{q} \\ \mathsf{p} - \alpha(\mathsf{q})\end{bmatrix}, \tag{2.2.10}$$

where $\xi = [\mathsf{q}^T\ \mathsf{p}^T]^T$ and $\alpha(\mathsf{q}) = -\kappa_1 M(\mathsf{q})\mathsf{q}$ for some $\kappa_1 > 0$. The dynamics of the stochastic control system Σ_s in (2.2.7) after the change of coordinates can be written by using Ito's differentiation [Øks02] as

$$\hat{\Sigma_s}: \begin{cases} \mathrm{d}\,\tilde{\zeta}_1 = M^{-1}(\tilde{\zeta}_1)(\tilde{\zeta}_2 + \alpha(\tilde{\zeta}_1))\,\mathrm{d}\, t, \\ \mathrm{d}\,\tilde{\zeta}_2 = \Big(\mathfrak{N}(\tilde{\zeta}_1, \tilde{\zeta}_2 + \alpha(\tilde{\zeta}_1)) + G(\tilde{\zeta}_1)\upsilon + \kappa_1\frac{\mathrm{d}\, M(\tilde{\zeta}_1)}{\mathrm{d}\, t}\tilde{\zeta}_1 + \kappa_1\big(\tilde{\zeta}_2 + \alpha(\tilde{\zeta}_1)\big)\Big)\,\mathrm{d}\, t \\ \qquad\qquad + g(\tilde{\zeta}_1)\,\mathrm{d}\, W_t + r(\tilde{\zeta}_1)\,\mathrm{d}\, P_t. \end{cases} \tag{2.2.11}$$

Now consider a candidate Lyapunov function $V_1(z_1, \hat{z}_1), \forall z_1, \hat{z}_1 \in \mathbb{R}^n$, for the $\tilde{\zeta}_1$-subsystem as follows

$$V_1(z_1, \hat{z}_1) = \frac{1}{\mathfrak{m}}\Big((z_1 - \hat{z}_1)^T(z_1 - \hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}.$$

The corresponding infinitesimal generator along $\tilde{\zeta}_1$-subsystem is given by

$$\mathcal{D}V_1(z_1,\hat{z}_1) = (z_1-\hat{z}_1)^T\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}-1} \Big[\big(M^{-1}(z_1)z_2 - M^{-1}(\hat{z}_1)\hat{z}_2\big) + \big(M^{-1}(z_1)\alpha(z_1) - M^{-1}(\hat{z}_1)\alpha(\hat{z}_1)\big)\Big].$$

Now by using the definition of $\alpha(z_1)$, consistency of norm, and (2.2.8), the infinitesimal generator reduces to

$$\mathcal{D}V_1(z_1,\hat{z}_1) \leq (L_1-\kappa_1)\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}} + L_2\big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\big)^{\frac{\mathfrak{m}}{2}-1}\|z_1-\hat{z}_1\|\|z_2-\hat{z}_2\|. \tag{2.2.12}$$

To handle the second term, we use Young's inequality [KKK95] as

$$ab \leq \frac{\varepsilon^{s_1}}{s_1}|a|_1^s + \frac{1}{s_2\varepsilon_2^s}|b|_2^s, \tag{2.2.13}$$

where $\varepsilon > 0$, constants $s_1, s_2 > 1$ satisfying condition $(s_1-1)(s_2-1) = 1$, and $a, b \in \mathbb{R}$. Now by using the consistency of norms and applying Young's inequality (2.2.13), we can reduce the second term in (2.2.12) to

$$\begin{aligned} L_2\big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\big)^{\frac{\mathfrak{m}}{2}-1}\|z_1-\hat{z}_1\|\|z_2-\hat{z}_2\| &= L_2\|z_1-\hat{z}_1\|^{\mathfrak{m}-1}\|z_2-\hat{z}_2\| \\ &\leq \frac{L_2\varepsilon_1^{r_1}}{r_1}\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}} + \frac{L_2}{s_1\varepsilon_1^{s_1}}\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}, \end{aligned} \tag{2.2.14}$$

where $r_1 = \frac{\mathfrak{m}}{\mathfrak{m}-1}$, $s_1 = \mathfrak{m}$, and ε_1 is any positive constant. After substituting inequality (2.2.14) in (2.2.12), one obtains

$$\mathcal{D}V_1(z_1,\hat{z}_1) \leq (L_1 + \frac{L_2\varepsilon_1^{r_1}}{r_1} - \kappa_1)\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}} + \frac{L_2}{s_1\varepsilon_1^{s_1}}\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}. \tag{2.2.15}$$

One can readily verify that V_1 is a $\delta_\exists$-ISS-$\mathrm{M}_\mathfrak{m}$ function for $\tilde{\zeta}_1$-subsystem with respect to z_2 as the input provided that $L_1 + \frac{L_2\varepsilon_1^{r_1}}{r_1} - \kappa_1 < 0$. Function V_1 satisfies the conditions in Definition 2.2.4 with $\underline{\alpha}(y) = \overline{\alpha}(y) = \frac{1}{\mathfrak{m}}y$, $\mathbf{d}$ is the natural Euclidean metric, $\kappa = \mathfrak{m}(\kappa_1 - L_1 - \frac{L_2\varepsilon_1^{r_1}}{r_1})$, and $\overline{\varphi}(y) = \frac{L_2}{s_1\varepsilon_1^{s_1}}y$, for any $y \in \mathbb{R}_0^+$.

Now consider a Lyapunov function $V_2(z_2,\hat{z}_2)$, $\forall z_2, \hat{z}_2 \in \mathbb{R}^n$, for the $\tilde{\zeta}_2$-subsystem as

$$V_2(z_2,\hat{z}_2) = \frac{1}{\mathfrak{m}}\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}.$$

The respective infinitesimal generator is given by

$$\begin{aligned}
\mathcal{D}V_2(z_2,\hat{z}_2) &= (z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}\\
&\Big[\big(Gu+\mathfrak{N}+\kappa_1\frac{\mathrm{d}\,M}{\mathrm{d}\,t}z_1+\kappa_1(z_2-\kappa_1 Mz_1)\big)-\big(\hat{G}\hat{u}+\hat{\mathfrak{N}}+\kappa_1\frac{\mathrm{d}\,\hat{M}}{\mathrm{d}\,t}\hat{z}_1+\kappa_1(\hat{z}_2-\kappa_1\hat{M}\hat{z}_1)\big)\Big]\\
&+\frac{1}{2}\operatorname{Tr}\Big(\big(g(z_1)-g(\hat{z}_1)\big)\big(g(z_1)-g(\hat{z}_1)\big)^T\partial_{z_2z_2}V_2(z_2,\hat{z}_2)\Big)\\
&+\frac{1}{\mathfrak{m}}\sum_{i=1}^{\tilde{r}}\lambda_i\Big(\big((z_2+r(z_1)e_i)-(\hat{z}_2+r(\hat{z}_1)e_i)\big)^T\big((z_2+r(z_1)e_i)-(\hat{z}_2+r(\hat{z}_1)e_i)\big)\Big)^{\frac{\mathfrak{m}}{2}}\\
&-\frac{1}{\mathfrak{m}}\sum_{i=1}^{\tilde{r}}\lambda_i\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}},
\end{aligned}\tag{2.2.16}$$

where $G=G(z_1)$, $\mathfrak{N}=\mathfrak{N}(z_1,z_2+\alpha(z_1))$, $M=M(z_1)$, $\hat{G}=G(\hat{z}_1)$, $\hat{\mathfrak{N}}=\mathfrak{N}(\hat{z}_1,\hat{z}_2+\alpha(\hat{z}_1))$, and $\hat{M}=M(\hat{z}_1)$. The same abbreviation will be used in the rest of the proof. The first term can be simply handled by selecting proper control input u and the second term can be reduced using consistency of norm, Lipschitz assumption on the diffusion term $g(\cdot)$ and the Young's inequality as follows

$$\begin{aligned}
&\frac{1}{2}\operatorname{Tr}\Big(\big(g(z_1)-g(\hat{z}_1)\big)\big(g(z_1)-g(\hat{z}_1)\big)^T\partial_{z_2z_2}V_2(z_2,\hat{z}_2)\Big)\\
&=\frac{1}{2}\operatorname{Tr}\Bigg(\big(g(z_1)-g(\hat{z}_1)\big)\big(g(z_1)-g(\hat{z}_1)\big)^T\Big[\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}I_n\\
&\quad+(\mathfrak{m}-2)(z_2-\hat{z}_2)(z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-2}\Big]\Bigg)\\
&\le\frac{\mathfrak{m}-1}{2}\|g(z_1)-g(\hat{z}_1)\|_F^2\|z_2-\hat{z}_2\|^{\mathfrak{m}-2}\le\frac{\min\{n,\breve{r}\}(\mathfrak{m}-1)}{2}\|g(z_1)-g(\hat{z}_1)\|^2\|z_2-\hat{z}_2\|^{\mathfrak{m}-2}\\
&\le\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2}\|z_1-\hat{z}_1\|^2\|z_2-\hat{z}_2\|^{\mathfrak{m}-2}\\
&\le\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2}\Big[\frac{\varepsilon_2^{r_2}}{r_2}\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}+\frac{1}{s_2\varepsilon_2^{s_2}}\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}\Big],
\end{aligned}\tag{2.2.17}$$

where $s_2=\frac{\mathfrak{m}}{\mathfrak{m}-2}$, $r_2=\frac{\mathfrak{m}}{2}$, and ε_2 is any positive constant. With the help of Jenson's inequality for convex functions [AS03] and of Lipschitz assumption on the reset term $r(\cdot)$ (cf. Definition 2.2.1), the third term in (2.2.16) can be reduced as

$$\begin{aligned}
&\frac{1}{\mathfrak{m}}\sum_{i=1}^{\tilde{r}}\lambda_i\Big[\|(z_2-\hat{z}_2)+\big(r(z_1)e_i-r(\hat{z}_1)e_i\big)\|^{\mathfrak{m}}-\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}\Big]\\
&\qquad\le\frac{1}{\mathfrak{m}}\sum_{i=1}^{\tilde{r}}\lambda_i\Big[2^{\mathfrak{m}-1}\|z_2-\hat{z}_2\|^{\mathfrak{m}}+2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\|z_1-\hat{z}_1\|^{\mathfrak{m}}-\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}\Big]
\end{aligned}$$

$$\leq \Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}}, \quad (2.2.18)$$

where $\lambda = \sum_{i=1}^{\tilde{r}}\lambda_i$. Finally, the infinitesimal generator (2.2.16) corresponding to $V_2(z_2,\hat{z}_2)$ can be reduced with the help of (2.2.17) and (2.2.18) to

$$\begin{aligned}\mathcal{D}V_2(z_2,\hat{z}_2) \leq{}& \Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}\Big(\frac{\min\{n,\breve{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2}+\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}}\Big)\\
&+(z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}\\
&\Big[\Big(Gu+\mathfrak{N}+\kappa_1\frac{\mathrm{d}\,M}{\mathrm{d}\,t}z_1+\kappa_1(z_2-\kappa_1Mz_1)+\Big(\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)z_2\Big)\\
&-\Big(\hat{G}\hat{u}+\hat{\mathfrak{N}}+\kappa_1\frac{\mathrm{d}\,\hat{M}}{\mathrm{d}\,t}\hat{z}_1+\kappa_1(\hat{z}_2-\kappa_1\hat{M}\hat{z}_1)+\Big(\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)\hat{z}_2\Big)\Big].\end{aligned} \quad (2.2.19)$$

Now consider a Lyapunov function V for the overall system (2.2.11) as $V(z,\hat{z}) = V_1(z_1,\hat{z}_1)+V_2(z_2,\hat{z}_2)$ and the respective infinitesimal generator can be obtained by using (2.2.15) and (2.2.19) as

$$\begin{aligned}\mathcal{D}V(z,\hat{z}) \leq{}& \Big(L_1+\frac{L_2\varepsilon_1^{r_1}}{r_1}+\frac{\min\{n,\breve{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2}+\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}}-\kappa_1\Big)\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}\\
&+(z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}\\
&\Big[\Big(Gu+\mathfrak{N}+\kappa_1\frac{\mathrm{d}\,M}{\mathrm{d}\,t}z_1+\kappa_1(z_2-\kappa_1Mz_1)+\Big(\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{L_2}{s_1\varepsilon_1^{s_1}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)z_2\Big)\\
&-\Big(\hat{G}\hat{u}+\hat{\mathfrak{N}}+\kappa_1\frac{\mathrm{d}\,\hat{M}}{\mathrm{d}\,t}\hat{z}_1+\kappa_1(\hat{z}_2-\kappa_1\hat{M}\hat{z}_1)+\Big(\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{L_2}{s_1\varepsilon_1^{s_1}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)\hat{z}_2\Big)\Big].\end{aligned} \quad (2.2.20)$$

If we choose the state feedback control law $u(z_1,z_2)$ as

$$\begin{aligned}u(z_1,z_2) ={}& G^{-1}\Big(-\mathfrak{N}-\kappa_1\frac{\mathrm{d}\,M}{\mathrm{d}\,t}z_1+\kappa_1^2Mz_1\\
&-\Big(2\kappa_1+\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{L_2}{s_1\varepsilon_1^{s_1}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)z_2+\overline{u}\Big),\end{aligned}$$

where $\hat{u}$ is a new control input with respect to which the closed-loop system will be shown to be $\delta_\exists$-ISS-$\mathrm{M}_\mathfrak{m}$. After using $u(z_1,z_2)$, the inequality (2.2.20) reduces to

$$\begin{aligned}\mathcal{D}V(z,\hat{z}) \leq{}& -\Big(\kappa_1-\Big(L_1+\frac{L_2\varepsilon_1^{r_1}}{r_1}+\frac{\min\{n,\breve{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2}+\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}}\Big)\Big)\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}\\
&-\kappa_1\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}+(z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}(\overline{u}-\hat{\overline{u}}).\end{aligned} \quad (2.2.21)$$

Now the third term can further be reduced by applying Young's inequality to

$$\begin{aligned}(z_2-\hat{z}_2)^T\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}-1}(\overline{u}-\hat{\overline{u}}) &\leq \|z_2-z_2\|^{\mathfrak{m}-1}\|\overline{u}-\hat{\overline{u}}\| \\ &\leq \frac{\varepsilon_1^{r_1}}{r_1}\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}+\frac{1}{s_1\varepsilon_1^{s_1}}\|\overline{u}-\hat{\overline{u}}\|^{\mathfrak{m}},\end{aligned} \tag{2.2.22}$$

where the parameters ε_1, s_1 and r_1 are the same as the ones in (2.2.14). Using (2.2.22), inequality (2.2.21) reduces to

$$\mathcal{D}V(z,\hat{z}) \leq -c_1\Big((z_1-\hat{z}_1)^T(z_1-\hat{z}_1)\Big)^{\frac{\mathfrak{m}}{2}}-c_2\Big((z_2-\hat{z}_2)^T(z_2-\hat{z}_2)\Big)^{\frac{\mathfrak{m}}{2}}+c_3\|\overline{u}-\hat{\overline{u}}\|^{\mathfrak{m}}, \tag{2.2.23}$$

where $c_1=\Big(\kappa_1-\big(L_1+\frac{L_2\varepsilon_1^{r_1}}{r_1}+\frac{\min\{n,\breve{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2}+\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}}\big)\Big)$, $c_2=\Big(\kappa_1-\frac{\varepsilon_1^{r_1}}{r_1}\Big)$, $c_3=\frac{1}{s_1\varepsilon_1^{s_1}}$, all required to be positive. By choosing the design parameter κ_1 as

$$\kappa_1 > L_1+\frac{\max\{L_2,1\}\varepsilon_1^{r_1}}{r_1}+\frac{\min\{n,\breve{r}\}L_g^2\varepsilon_2^{r_2}(\mathfrak{m}-1)}{2r_2}+\frac{2^{\mathfrak{m}-1}L_r^{\mathfrak{m}}\lambda}{\mathfrak{m}},$$

one obtains c_1, c_2, $c_3>0$.
If $\kappa=\min\{kc_1,kc_2\}$, the inequality (2.2.23) can further be reduced to

$$\mathcal{D}V \leq -\kappa V(z,\hat{z})+\overline{\varphi}(\|\overline{u}-\hat{\overline{u}}\|^{\mathfrak{m}}), \tag{2.2.24}$$

where $\overline{\varphi}(y)=c_3y, \forall y\in\mathbb{R}_0^+$, which satisfies condition (iii) in Definition 2.2.4. One can readily verify that conditions (i) and (ii) in Definition 2.2.4 are satisfied by defining metric $\mathbf{d}$ as the natural Euclidean one, and defining $\underline{\alpha}(y)=\frac{1}{2^{\frac{\mathfrak{m}}{2}-1}\mathfrak{m}}y$, and $\overline{\alpha}(y)=\frac{1}{\mathfrak{m}}y, \forall y\in\mathbb{R}_0^+$. Now with the help of Theorem 2.2.5, one obtains

$$\mathbb{E}[\|\zeta_{z\overline{v}}(t)-\zeta_{\hat{z}\hat{\overline{v}}}(t)\|^{\mathfrak{m}}] \leq \beta(\mathbb{E}[\|z-\hat{z}\|^{\mathfrak{m}}],t)+\gamma(\mathbb{E}[\|\overline{v}-\hat{\overline{v}}\|_\infty^{\mathfrak{m}}]), \tag{2.2.25}$$

where $\zeta_{z\hat{v}}(t)$ denotes the value of the solution process of $\hat{\Sigma}_s$ in (2.2.11) at time $t\in\mathbb{R}_0^+$ under the input signal $\hat{v}$ and from the initial condition $\zeta_{z\hat{v}}(0)=z$ $\mathbb{P}$-a.s. The function $\beta\in\mathcal{KL}$, and the function $\gamma\in\mathcal{K}_\infty$ can be defined as

$$\beta(y,t)=\underline{\alpha}^{-1}(\overline{\alpha}(y)\mathsf{e}^{-\kappa t})=2^{\frac{\mathfrak{m}}{2}-1}\mathsf{e}^{-\kappa t}y, \quad \gamma(y)=\underline{\alpha}^{-1}(\frac{\overline{\varphi}(y)}{\kappa})=\frac{2^{\frac{\mathfrak{m}}{2}-1}\mathfrak{m}}{\kappa}c_3y, \tag{2.2.26}$$

for all $y\in\mathbb{R}_0^+$. Now by applying the change of coordinate $\zeta=\psi(\xi)$, where $\xi=[\mathsf{q}^T\ \mathsf{p}^T]^T$, the control law υ reduces to

$$\begin{aligned}\upsilon=G^{-1}(\mathsf{q})\Big(&-\mathfrak{N}(\mathsf{q},\mathsf{p})+\kappa_1\frac{\mathrm{d}\,M(\mathsf{q})}{\mathrm{d}\,t}\mathsf{q}-\kappa_1^2M(\mathsf{q})\mathsf{q} \\ &-\Big(\frac{(2^{\mathfrak{m}-1}-1)\lambda}{\mathfrak{m}}+\frac{L_2}{s_1\varepsilon_1^{s_1}}+\frac{\min\{n,\breve{r}\}L_g^2(\mathfrak{m}-1)}{2s_2\varepsilon_2^{s_2}}\Big)\big(\mathsf{p}+\kappa_1M(\mathsf{q})\mathsf{q}\big)+\overline{\upsilon}\Big),\end{aligned} \tag{2.2.27}$$

and (2.2.25) can be rewritten as

$$\mathbb{E}[\|\psi(\xi_{x\overline{\upsilon}}(t)) - \psi(\xi_{\hat{x}\hat{\overline{\upsilon}}}(t))\|^{\mathfrak{m}}] \leq \beta(\mathbb{E}[\|\psi(x) - \psi(\hat{x})\|^{\mathfrak{m}}], t) + \gamma(\mathbb{E}[\|\overline{\upsilon} - \hat{\overline{\upsilon}}\|_{\infty}^{\mathfrak{m}}]), \quad (2.2.28)$$

where $x = [q^T \ p^T]^T$. By defining a metric[1] $\mathbf{d}(x, \hat{x}) = \|\psi(x) - \psi(\hat{x})\|$, we can rewrite (2.2.28) as

$$\mathbb{E}[(\mathbf{d}(\xi_{x\overline{\upsilon}}(t),\xi_{\hat{x}\hat{\overline{\upsilon}}}(t)))^{\mathfrak{m}}] \leq \beta(\mathbb{E}[(\mathbf{d}(x, \hat{x}))^{\mathfrak{m}}], t) + \gamma(\mathbb{E}[\|\overline{\upsilon} - \hat{\overline{\upsilon}}\|_{\infty}^{\mathfrak{m}}]), \quad (2.2.29)$$

which satisfies condition (2.2.2) for original Σ_s. Hence, Σ_s in (2.2.7) equipped with the feedback control law (2.2.27) is $\delta_{\exists}$-ISS-$\mathrm{M}_{\mathfrak{m}}$ for any $\mathfrak{m} > 2$. □

The next corollary provides the same results as the ones in Theorem 2.2.7 but for $\mathfrak{m} = 2$.

Corollary 2.2.8. *Consider the stochastic control system Σ_s in (2.2.7). The state feedback control law*

$$\upsilon = G^{-1}(\mathsf{q})\Big(- \mathfrak{N}(\mathsf{q}, \mathsf{p}) - \kappa_1 \frac{\mathrm{d}\, M(\mathsf{q})}{\mathrm{d}\, t}\mathsf{q} + \kappa_1^2 M(\mathsf{q})\mathsf{q} - \Big(2\kappa_1 + \frac{\lambda}{2} + \frac{L_2}{2\varepsilon_1^2}\Big)\big(\mathsf{p} + \kappa_1 M(\mathsf{q})\mathsf{q}\big) + \overline{\upsilon}\Big),$$

renders the closed-loop stochastic control system $\delta_{\exists}$-ISS-M_2 with respect to input $\hat{\upsilon}$, for all

$$\kappa_1 > L_1 + \frac{\max\{L_2, 1\}\varepsilon_1^2}{2} + \frac{\min\{n, \breve{r}\}L_g^2}{2} + L_r^2\lambda,$$

where ε_1 is any positive constant which can be chosen arbitrarily, and L_1, L_2, L_g, and L_r are the Lipschitz constants introduced in (2.2.8) and Definition 2.2.1, respectively.

Proof. The corollary is a particular case of Theorem 2.2.7. The proof is almost similar to that of Theorem 2.2.7 by substituting $\mathfrak{m} = 2$. The only difference is the trace term (2.2.17) in $\tilde{\zeta}_2$-subsystem which is now given by

$$\begin{aligned}&\frac{1}{2}\operatorname{Tr}\Big(\big(g(z_1) - g(\hat{z}_1)\big)\big(g(z_1) - g(\hat{z}_1)\big)^T \partial_{z_2 z_2} V_2(z_2, \hat{z}_2)\Big)\\ &\quad\leq \frac{1}{2}\operatorname{Tr}\Big(\big(g(z_1) - g(\hat{z}_1)\big)\big(g(z_1) - g(\hat{z}_1)\big)^T\Big) \leq \frac{\min\{n, \breve{r}\}L_g^2}{2}(z_1 - \hat{z}_1)^T(z_1 - \hat{z}_1).\end{aligned}$$

The rest of the proof follows similar to that of Theorem 2.2.5. □

Remark 2.2.9. *Assume that for all $x, \hat{x} \in \mathbb{R}^n$, the change of coordinate map ψ in (2.2.10) satisfies*

$$\underline{\chi}(\|x - \hat{x}\|^{\mathfrak{m}}) \leq \|\psi(x) - \psi(\hat{x})\|^{\mathfrak{m}} \leq \overline{\chi}(\|x - \hat{x}\|^{\mathfrak{m}}),$$

for some convex function $\underline{\chi} \in \mathcal{K}_\infty$ and concave function $\overline{\chi} \in \mathcal{K}_\infty$. Then, inequality (2.2.29) for the original system Σ_s reduces to

$$\mathbb{E}[\|\xi_{x\overline{\upsilon}}(t) - \xi_{\hat{x}\hat{\overline{\upsilon}}}(t)\|^{\mathfrak{m}}] \leq \hat{\beta}(\mathbb{E}[\|x - \hat{x}\|^{\mathfrak{m}}], t) + \hat{\gamma}(\mathbb{E}[\|\overline{\upsilon} - \hat{\overline{\upsilon}}\|_{\infty}^{\mathfrak{m}}]),$$

[1] Since ψ is a bijective function, $\mathbf{d}$ satisfies all the requirements of a metric.

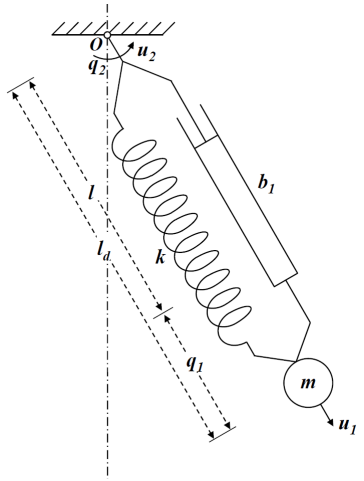

Figure 2.1: Controlled spring pendulum.

for the $\mathcal{KL}$ function $\hat{\beta}(y,t) = \underline{\chi}^{-1}(2\beta(\overline{\chi}(y),t))$ and the $\mathcal{K}_\infty$ function $\hat{\gamma}(y) = \underline{\chi}^{-1}(2\gamma(y))$, for any $y,t \in \mathbb{R}_0^+$. Note that if $\underline{\chi}^{-1}$ satisfies the triangle inequality (i.e., $\underline{\chi}^{-1}(a+b) \leq \underline{\chi}^{-1}(a) + \underline{\chi}^{-1}(b)$), one can remove the coefficients 2 in the expressions of $\hat{\beta}$ and $\hat{\gamma}$. Particularly, if the inertia matrix (M) is constant, one has

$$\left\| \begin{bmatrix} q - \hat{q} \\ (p + \kappa_1 M q) - (\hat{p} + \kappa_1 M \hat{q}) \end{bmatrix} \right\| = \left\| A \begin{bmatrix} q - \hat{q} \\ p - \hat{p} \end{bmatrix} \right\| = \|A(x - \hat{x})\|,$$

where A is a constant matrix given by

$$A = \begin{bmatrix} I_n & 0_n \\ \kappa_1 M & I_n \end{bmatrix}.$$

Therefore, one obtains

$$(\lambda_{\min}(A^T A))^{\frac{\mathfrak{m}}{2}} \|x - \hat{x}\|^{\mathfrak{m}} \leq \|\psi(x) - \psi(\hat{x})\|^{\mathfrak{m}} = \|A(x - \hat{x})\|^{\mathfrak{m}} \leq (\lambda_{\max}(A^T A))^{\frac{\mathfrak{m}}{2}} \|x - \hat{x}\|^{\mathfrak{m}},$$

where $\lambda_{\min}(A^T A)$ and $\lambda_{\max}(A^T A)$ denote minimum and maximum eigenvalues of $A^T A$, respectively.

2.2.4 Case Study

To verify the efficacy of the control design framework proposed in this section, we illustrate the results on a spring pendulum attached to stochastically vibrating ceiling and subject to random jumps such as sudden jerks due to interaction with environmental disturbances. The nonlinear dynamics of the considered system is borrowed from [WCS12], now affected by jumps and schematically shown in Figure 2.1. Let us define the generalized coordinate vector as $\mathsf{q} = [\mathsf{q}_1 \ \mathsf{q}_2]^T$, where q_1 represents change of arm length as a difference between the dynamic length (l_d) and static length (l) of a spring

pendulum; and q_2 is the angle of pendulum with vertical axis. The corresponding generalized momenta vector is given by $\mathsf{p} = [m\frac{\mathrm{d}\,\mathsf{q}_1}{\mathrm{d}\,t} \;\; m(l+\mathsf{q}_1)^2\frac{\mathrm{d}\,\mathsf{q}_2}{\mathrm{d}\,t}]^T$, where m is the mass of the ball, which gives the inertia matrix $M(q)$ as

$$M(q) = \begin{bmatrix} m & 0 \\ 0 & m(l+q_1)^2 \end{bmatrix}. \tag{2.2.30}$$

The Hamiltonian function $\mathcal{H}(q,p)$ is given by the total energy of the system as

$$\mathcal{H}(q,p) = \frac{p_1^2}{2m} + \frac{p_2^2}{2m(l+q_1)^2} + \frac{k_s q_1^2}{2} + m g_a (l+q_1)(1-\cos q_2),$$

where k_s is an elasticity coefficient of spring and g_a is the acceleration due to gravity. Now $\mathfrak{N}(q,p) = -\frac{\partial \mathcal{H}}{\partial q}(q,p) + b(q,p)$ can be calculated as

$$\mathfrak{N}(q,p) = \begin{bmatrix} \frac{p_2^2}{m(l+q_1)^3} - k_s q_1 - m g_a (1-\cos q_2) \\ -m g_a (l+q_1)\sin q_2 \end{bmatrix} + \begin{bmatrix} -\frac{b_1 p_1}{m} \\ -\frac{b_2 p_2}{m} \end{bmatrix}, \tag{2.2.31}$$

where b_1 is a damping coefficient of piston and b_2 is an air damping coefficient. By considering a 2-dimensional Brownian motion, the diffusion function $g(q)$ can be determined with the help of notion of relative kinematics by considering point O in Figure 2.1 stochastically vibrating [WCS12] which is given by

$$g(q) = \begin{bmatrix} -m\sin q_2 & m\cos q_2 \\ -m(l+q_1)\cos q_2 & -m(l+q_1)\sin q_2 \end{bmatrix}.$$

To introduce abrupt jumps in the system, we consider a one-dimensional Poison process with the rate $\lambda = 1$ and linear reset function $r(\mathsf{q}) = \mathsf{q}$. The term $\frac{\mathrm{d}\,M(\mathsf{q})}{\mathrm{d}\,t}$ can be obtained as

$$\frac{\mathrm{d}\,M(\mathsf{q})}{\mathrm{d}\,t} = \frac{\partial M(\mathsf{q})}{\partial q^T} \times \Big(\frac{\mathrm{d}\,\mathsf{q}}{\mathrm{d}\,t} \otimes I_2\Big) = \begin{bmatrix} 0 & 0 \\ 0 & \frac{2(1+\mathsf{q}_1)\mathsf{p}_1}{m} \end{bmatrix}. \tag{2.2.32}$$

As control input $\upsilon = [\upsilon_1 \; \upsilon_2]^T$ itself acting on the mass, one gets $G(q) = I_2$.

Now with the help of (2.2.30), (2.2.31), (2.2.32), Theorem 2.2.7, and fixing $\mathfrak{m} = 2$, we can obtain the final state feedback control input υ for the considered system as follows

$$\begin{aligned}
\upsilon_1(\mathsf{q},\mathsf{p}) =& -\frac{\mathsf{p}_2^2}{m(l+\mathsf{q}_1)^3} + k_s\mathsf{q}_1 + m g_a(1-\cos\mathsf{q}_2) + \frac{b_1\mathsf{p}_1}{m} - \kappa_1\Big(\kappa_1 + \frac{\lambda}{2} + \frac{L_2}{2\varepsilon_1^2}\Big)m\mathsf{q}_1 \\
&- \Big(2\kappa_1 + \frac{\lambda}{2} + \frac{L_2}{2\varepsilon_1^2}\Big)\mathsf{p}_1 + \overline{\upsilon}_1, \\
\upsilon_2(\mathsf{q},\mathsf{p}) =& m g_a(l+\mathsf{q}_1)\sin\mathsf{q}_2 + \frac{b_2\mathsf{p}_2}{m} + \frac{2\kappa_1(1+\mathsf{q}_1)\mathsf{p}_1\mathsf{q}_2}{m} - \kappa_1\Big(\kappa_1 + \frac{\lambda}{2} + \frac{L_2}{2\varepsilon_1^2}\Big)m\mathsf{q}_2(l+\mathsf{q}_1)^2 \\
&- \Big(2\kappa_1 + \frac{\lambda}{2} + \frac{L_2}{2\varepsilon_1^2}\Big)\mathsf{p}_2 + \overline{\upsilon}_2,
\end{aligned}$$

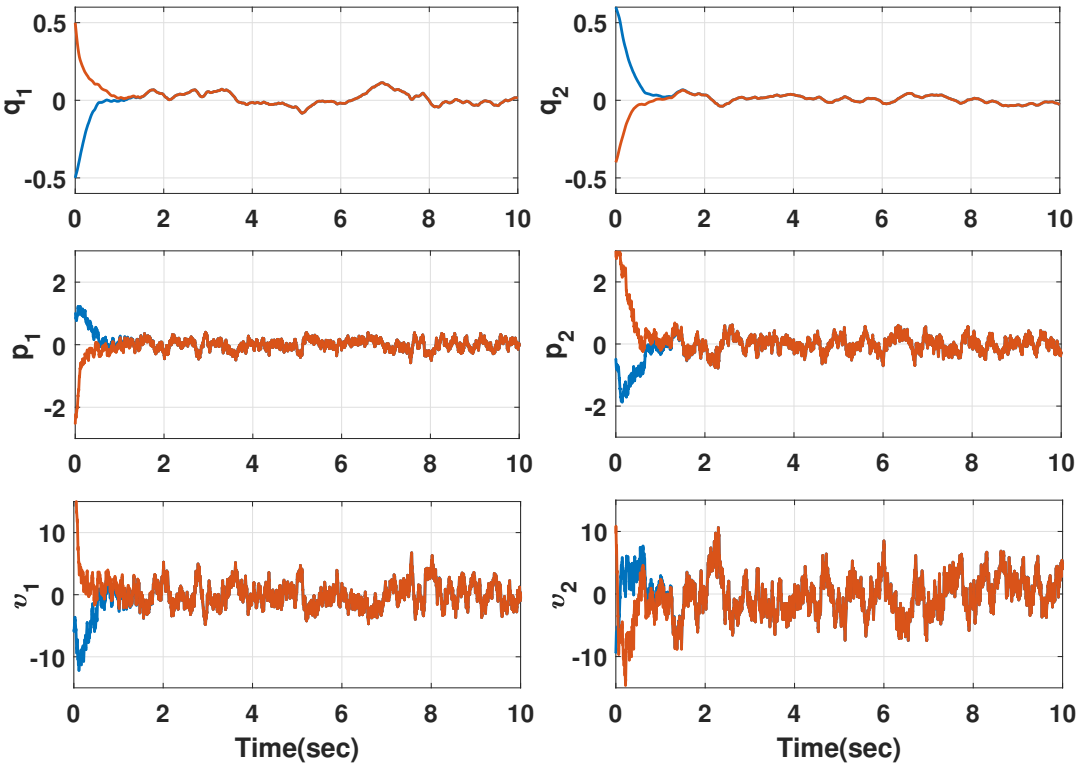

Figure 2.2: Two trajectories q (top two plots), two trajectories p (middle two plots) started from two different initial conditions $[q_1, q_2, p_1, p_2]^T = [0.5, -0.4, -2.5, 3]^T$ and $[\hat{q}_1, \hat{q}_2, \hat{p}_1, \hat{p}_2]^T = [-0.5, 0.6, 1, -0.5]^T$, and the two corresponding input trajectories υ_1 and υ_2 (bottom two plots).

rendering the closed-loop system $\delta_\exists$-ISS-M_2 with respect to input $[\overline{\upsilon}_1\ \overline{\upsilon}_2]^T$ for any arbitrarily chosen $\varepsilon_1 > 0$ and appropriately chosen κ_1. For the simulation purpose, we consider system parameters as $m = 0.8$, $l = 1.5$, $g_a = 9.8$, $k_s = 15$, $b_1 = 1$, and $b_2 = 1$; all the constants and the variables are considered in SI units; the Lipschitz constants are computed as $L_1 = 1, L_2 = 2, L_g = 1$, and $L_r = 1$, and the design parameters are chosen as $\varepsilon_1 = 0.5$ and $\kappa_1 = 4$. We choose inputs $\overline{\upsilon}_1(t) = \overline{\upsilon}_2(t) = 0.5 \sin t$. Figure 2.2 shows the evolution of the closed-loop trajectories q and p in the presence of Brownian noise and Poisson jumps started from two different initial conditions $[q_1, q_2, p_1, p_2]^T$=[0.5, -0.4, -2.5, 3]^T and $[\hat{q}_1, \hat{q}_2, \hat{p}_1, \hat{p}_2]^T$=[$-0.5$, 0.6, 1, -0.5]^T and the evolution of the corresponding input trajectories υ_1 and υ_2. Figure 2.2 shows that indeed, by virtue of the $\delta_\exists$-ISS-M_2 property, both trajectories converge to each other. To verify the bound on $\mathbb{E}[\|\zeta_{z\overline{\upsilon}}(t) - \zeta_{\hat{z}\hat{\overline{\upsilon}}}(t)\|^2]$ as given in (2.2.25), we simulated the closed-loop system for 5000 realizations, two fixed initial conditions, and the same input for both trajectories (i.e., $\overline{\upsilon} = \hat{\overline{\upsilon}}$). The inequality (2.2.25) reduces to

$$\mathbb{E}[\|\zeta_{z\overline{\upsilon}}(t) - \zeta_{\hat{z}\hat{\overline{\upsilon}}}(t)\|^2] \leq \beta(\|z - \hat{z}\|^2, t), \tag{2.2.33}$$

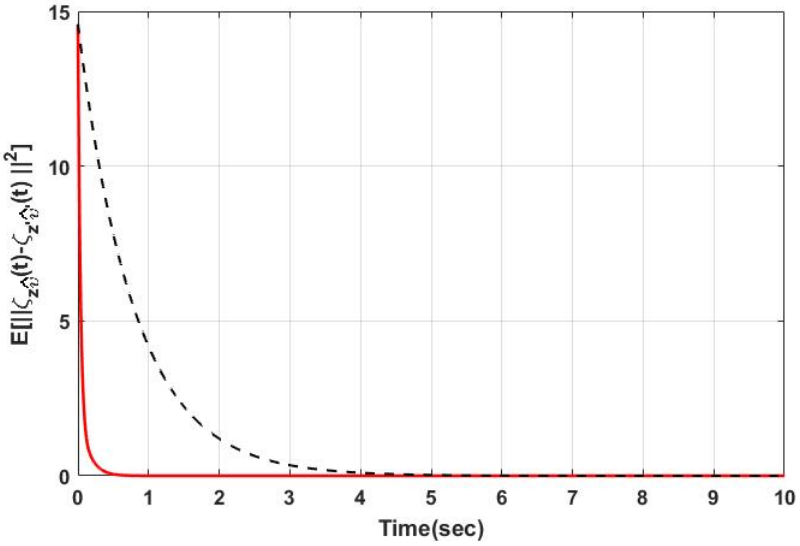

Figure 2.3: The average value of the squared distance of two trajectories of $\hat{\Sigma}_s$ started from two different initial conditions $z = [0.5, -0.4, -0.9, -2.12]^T$ and $\hat{z} = [-0.5, 0.6, -0.6, 1.42]^T$. The black doted curve indicates corresponding bound given by (2.2.33).

where the function $\beta \in \mathcal{KL}$ is given in (2.2.26) and computed as $\beta(y,t) = \mathsf{e}^{-\kappa t}y$ with $\kappa = 1.25$. The average value of the squared distance of two trajectories of $\hat{\Sigma}_s$ started from two different initial conditions $z = [0.5,\ -0.4,\ -0.9,\ -2.12]^T$ and $\hat{z} = [-0.5,\ 0.6,\ -0.6,\ 1.42]^T$ together with computed theoretical bound are shown in Figure 2.3. One can readily verify that the simulated distance is always lower than the computed theoretical one in (2.2.33).

2.3 Incremental Stability of Retarded Jump-Diffusion Systems

In this section, we provide results on the characterization of a notion of incremental input-to-state stability for retarded jump-diffusion systems as defined next.

2.3.1 Retarded Jump-Diffusion Systems (RJDS)

Definition 2.3.1. *A retarded jump-diffusion system (RJDS) is a tuple* $\Sigma_R = (\mathbb{R}^n, \mathcal{X}, \mathsf{U}, \mathcal{U}, \overline{f}, \overline{g}, \overline{r})$, *where:*

- $\mathbb{R}^n$ *is the Euclidean space;*
- $\mathcal{X}$ *is a subset of* $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$, *for some* $\tau \in \mathbb{R}_0^+$;
- $\mathsf{U} \subseteq \mathbb{R}^m$ *is the input set;*
- $\mathcal{U}$ *is a subset of the set of all measurable, locally essentially bounded functions of time from* $\mathbb{R}_0^+$ *to* U;

- $\overline{f} : \mathcal{X} \times \mathsf{U} \to \mathbb{R}^n$, *satisfies the following Lipschitz assumption: there exist constants* $\overline{L}_{\overline{f}}, \overline{L}_u \in \mathbb{R}^+$, *such that* $\|\overline{f}(\mathsf{x}_t, u) - \overline{f}(\hat{\mathsf{x}}_t, \hat{u})\| \leq \overline{L}_{\overline{f}} \|\mathsf{x}_t - \hat{\mathsf{x}}_t\|_{[-\tau,0]} + \overline{L}_u \|u - \hat{u}\|$ *for all* $\mathsf{x}_t, \hat{\mathsf{x}}_t \in \mathcal{X}$ *and all* $u, \hat{u} \in \mathsf{U}$;
- $\overline{g} : \mathcal{X} \to \mathbb{R}^{n \times \check{r}}$ *satisfies the following Lipschitz assumption: there exists a constant* $\overline{L}_{\overline{g}} \in \mathbb{R}_0^+$ *such that* $\|\overline{g}(\mathsf{x}_t) - \overline{g}(\hat{\mathsf{x}}_t)\| \leq \overline{L}_{\overline{g}} \|\mathsf{x}_t - \hat{\mathsf{x}}_t\|_{[-\tau,0]}$ *for all* $\mathsf{x}_t, \hat{\mathsf{x}}_t \in \mathcal{X}$;
- $\overline{r} : \mathcal{X} \to \mathbb{R}^{n \times \tilde{r}}$ *satisfies the following Lipschitz assumption: there exists a constant* $\overline{L}_{\overline{r}} \in \mathbb{R}_0^+$ *such that* $\|\overline{r}(\mathsf{x}_t) - \overline{r}(\hat{\mathsf{x}}_t)\| \leq \overline{L}_{\overline{r}} \|\mathsf{x}_t - \hat{\mathsf{x}}_t\|_{[-\tau,0]}$ *for all* $\mathsf{x}_t, \hat{\mathsf{x}}_t \in \mathcal{X}$.

An $\mathbb{R}^n$-valued continuous-time process ξ is said to be a *solution process* for Σ_R if there exists $\upsilon \in \mathcal{U}$ satisfying

$$\mathrm{d}\,\xi(t) = \overline{f}(\xi_t, \upsilon(t))\,\mathrm{d}\,t + \overline{g}(\xi_t)\,\mathrm{d}\,W_t + \overline{r}(\xi_t)\,\mathrm{d}\,P_t, \tag{2.3.1}$$

$\mathbb{P}$-almost surely ($\mathbb{P}$-a.s.), where $\overline{f}$, $\overline{g}$, and $\overline{r}$ are the drift, diffusion, and reset terms, respectively, and $\xi_t := \{\xi(t+\theta) | -\tau \leq \theta \leq 0\}$. We emphasize that postulated assumptions on $\overline{f}$, $\overline{g}$, and $\overline{r}$ ensure the existence and uniqueness of the solution process ξ on $t \geq -\tau$ [ØS05, Theorem 1.19]. Throughout the subsection we use the notation $\xi_{\zeta,\upsilon}(t)$ to denote the value of a solution process starting from initial condition $\zeta = \{\xi(\theta) | -\tau \leq \theta \leq 0\} \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau, 0]; \mathbb{R}^n)$ $\mathbb{P}$-a.s. and under the input signal υ at time t. We also use the notation $\xi_{t,\zeta,\upsilon}$ to denote the solution process starting from initial condition $\zeta = \{\xi(\theta) | -\tau \leq \theta \leq 0\} \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau, 0]; \mathbb{R}^n)$ $\mathbb{P}$-a.s. and under the input signal υ. Note that for any $t \in \mathbb{R}_0^+$, $\xi_{\zeta,\upsilon}(t)$ is a random variable taking values in $\mathbb{R}^n$ and $\xi_{t,\zeta,\upsilon}$ is a random variable taking values in $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$. Here, we assume that the Poisson process and the Brownian motion are independent of each other. The Poisson process $P_s := [P_s^1; \ldots; P_s^{\tilde{r}}]$ models $\tilde{r}$ kinds of events whose occurrences are assumed to be independent of each other and have the constant rates of λ_i for each P_s^i, $i \in [1; \tilde{r}]$. Now we will introduce delayed jump-diffusion system (Σ_D)(DJDS) as a special case of retarded jump-diffusion system which is given by

$$\mathrm{d}\,\xi(t) = F(\xi(t), \xi(t-\tau_1), \upsilon(t))\,\mathrm{d}\,t + G(\xi(t), \xi(t-\tau_2))\,\mathrm{d}\,W_t + R(\xi(t), \xi(t-\tau_3))\,\mathrm{d}\,P_t, \tag{2.3.2}$$

where $F : \mathbb{R}^n \times \mathbb{R}^n \times \mathsf{U} \to \mathbb{R}^n$, $G : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n \times \check{r}}$, and $R : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n \times \tilde{r}}$ are the drift, diffusion, and reset terms, respectively. The constants τ_1, τ_2, and τ_3 are the state delay in the drift, diffusion, and reset terms, respectively.

2.3.2 Incremental Stability for RJDS

Here, we introduce a notion of incremental stability for RJDS (resp. DJDS).

Definition 2.3.2. *An RJDS* Σ_R *(resp. DJDS* Σ_D*) is incrementally input-to-state stable in the* $\mathfrak{m}^{th}$ *moment, where* $\mathfrak{m} \geq 1$, *denoted by (*δ*-ISS-*$M_\mathfrak{m}$*), if there exist a function* $\beta \in \mathcal{KL}$ *and a function* $\gamma \in \mathcal{K}_\infty$ *such that for any* $t \in \mathbb{R}_0^+$, *any two initial conditions* $\zeta, \hat{\zeta} \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau, 0]; \mathbb{R}^n)$, *and any* $\upsilon, \hat{\upsilon} \in \mathcal{U}$ *the following condition is satisfied:*

$$\mathbb{E}[\|\xi_{\zeta,\upsilon}(t) - \xi_{\hat{\zeta},\hat{\upsilon}}(t)\|^\mathfrak{m}] \leq \beta(\mathbb{E}[\|\zeta - \hat{\zeta}\|^\mathfrak{m}_{[-\tau,0]}], t) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty). \tag{2.3.3}$$

One can readily verify that in the absence of delay, Definition 2.3.2 reduces to that of δ-ISS-$\mathrm{M}_{\mathfrak{m}}$ for stochastic control systems in [ZMEM^{+}14, Definition 3.1].

For later use, we provide the infinitesimal generators (denoted by operator $\mathcal{D}$) for an RJDS Σ_R and a DJDS Σ_D using Itô's differentiation [JP09, equation (23)]. Let function $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_0^+$ be twice differentiable on $\mathbb{R}^n \times \mathbb{R}^n \setminus \triangle$. The infinitesimal generator of V associated with an RJDS Σ_R in (2.3.1) is an operator, denoted by $\mathcal{D}V$, from $\mathcal{C}([-\tau, 0]; \mathbb{R}^n) \times \mathcal{C}([-\tau, 0]; \mathbb{R}^n)$ to $\mathbb{R}$, and $\forall t \in \mathbb{R}_0^+, \forall \mathsf{x}_t, \hat{\mathsf{x}}_t \in \mathcal{C}([-\tau, 0]; \mathbb{R}^n)$ and $\forall u, \hat{u} \in \mathsf{U}$ it is given by

$$\mathcal{D}V(\mathsf{x}_t, \hat{\mathsf{x}}_t, u, \hat{u}) := \begin{bmatrix}\partial_x V & \partial_{\hat{x}} V\end{bmatrix} \begin{bmatrix}\overline{f}(\mathsf{x}_t, u) \\ \overline{f}(\hat{\mathsf{x}}_t, \hat{u})\end{bmatrix} + \frac{1}{2}\mathsf{Tr}\left(\begin{bmatrix}\overline{g}(\mathsf{x}_t) \\ \overline{g}(\hat{\mathsf{x}}_t)\end{bmatrix}\begin{bmatrix}\overline{g}^T(\mathsf{x}_t)\, \overline{g}^T(\hat{\mathsf{x}}_t)\end{bmatrix}\begin{bmatrix}\partial_{x,x} V & \partial_{x,\hat{x}} V \\ \partial_{\hat{x},x} V & \partial_{\hat{x},\hat{x}} V\end{bmatrix}\right) + \sum_{i=1}^{\tilde{r}} \lambda_i\big(V(\mathsf{x}_t(0) + \overline{r}(\mathsf{x}_t)e_i, \hat{\mathsf{x}}_t(0) + \overline{r}(\hat{\mathsf{x}}_t)e_i) - V(\mathsf{x}_t(0), \hat{\mathsf{x}}_t(0))\big). \quad (2.3.4)$$

The infinitesimal generator of V associated with a DJDS Σ_D in (2.3.2) is an operator, denoted by $\mathcal{D}V$, from $\mathbb{R}^{8n}$ to $\mathbb{R}$ and $\forall x, \hat{x}, y, \hat{y}, z, \hat{z}, p, \hat{p} \in \mathbb{R}^n$, and $\forall u, \hat{u} \in \mathsf{U}$ it is given by

$$\begin{aligned}&\mathcal{D}V(x, \hat{x}, y, \hat{y}, z, \hat{z}, p, \hat{p}, u, \hat{u}) \\ &\quad := \begin{bmatrix}\partial_x V \partial_{\hat{x}} V\end{bmatrix} \begin{bmatrix}F(x, y, u) \\ F(\hat{x}, \hat{y}, \hat{u})\end{bmatrix} + \frac{1}{2}\mathsf{Tr}\left(\begin{bmatrix}G(x, z) \\ G(\hat{x}, \hat{z})\end{bmatrix}\begin{bmatrix}G^T(x, z)\, G^T(\hat{x}, \hat{z})\end{bmatrix}\begin{bmatrix}\partial_{x,x} V & \partial_{x,\hat{x}} V \\ \partial_{\hat{x},x} V & \partial_{\hat{x},\hat{x}} V\end{bmatrix}\right) \\ &\qquad + \sum_{i=1}^{\tilde{r}} \lambda_i\big(V(x + R(x, p)e_i, \hat{x} + R(\hat{x}, \hat{p})e_i) - V(x, \hat{x})\big). \end{aligned} \quad (2.3.5)$$

The symbols ∂_x and $\partial_{x,\hat{x}}$ in (2.3.4) and (2.3.5) represent first and second-order partial derivatives with respect to x (1st argument) and $\hat{x}$ (2nd argument), respectively. Note that we dropped the arguments of $\partial_x V$, $\partial_{\hat{x}} V$, $\partial_{x,x} V$, $\partial_{\hat{x},x} V$, $\partial_{x,\hat{x}} V$, and $\partial_{\hat{x},\hat{x}} V$ in (2.3.4) and (2.3.5) for the sake of simplicity.

Now we describe δ-ISS-$\mathrm{M}_{\mathfrak{m}}$ in terms of existence of so-called δ-ISS-$\mathrm{M}_{\mathfrak{m}}$ Lyapunov functions for RJDS and DJDS using Razumikhin-type condition as defined next.

Definition 2.3.3. *Consider an RJDS Σ_R and a continuous function $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_0^+$ that is twice differentiable on $\mathbb{R}^n \times \mathbb{R}^n \setminus \triangle$. The function V is called a δ-ISS-$M_{\mathfrak{m}}$ Lyapunov function for Σ_R for $\mathfrak{m} \geq 1$, if there exist functions $\underline{\alpha}, \overline{\alpha}, \overline{\varphi} \in \mathcal{K}_\infty$, such that:*

(i) $\underline{\alpha}$ *(resp.* $\overline{\alpha}$ *) is a convex (resp. concave) function;*

(ii) $\forall x, \hat{x} \in \mathbb{R}^n$, $\underline{\alpha}(\|x - \hat{x}\|^{\mathfrak{m}}) \leq V(x, \hat{x}) \leq \overline{\alpha}(\|x - \hat{x}\|^{\mathfrak{m}})$;

(iii) $\forall u, \hat{u} \in \mathsf{U}$ *and* $\forall t \geq 0$,

$$\mathbb{E}[\mathcal{D}V(\phi, \hat{\phi}, u, \hat{u})] \leq -\mathbb{E}[\varkappa(\phi(0), \hat{\phi}(0))] + \overline{\varphi}(\|u - \hat{u}\|), \quad (2.3.6)$$

for all $\phi, \hat{\phi} \in \mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau, 0]; \mathbb{R}^n)$ *satisfying*

$$\mathbb{E}[V(\phi(\theta), \hat{\phi}(\theta))] \leq \mathbb{E}[\tilde{q}(\phi(0), \hat{\phi}(0))], \ \forall \theta \in [-\tau, 0]; \quad (2.3.7)$$

where $\varkappa : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$ *is a nonnegative function such that there exists a function* $\tilde{\varkappa} \in \mathcal{K}_\infty$ *satisfying* $\varkappa(x,\hat{x}) \geq \tilde{\varkappa}(\|x-\hat{x}\|^{\mathfrak{m}})$ *and* $\lim_{\|s\|\to\infty} \frac{\tilde{\varkappa}(\|s\|^{\mathfrak{m}})}{\overline{\alpha}(\|s\|^{\mathfrak{m}})} > 0$*;* $\tilde{q} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ *is a function such that* $\tilde{q}(x,\hat{x}) - V(x,\hat{x}) \geq \overline{q}(\|x-\hat{x}\|)$*, where* $\overline{q}$ *is a* $\mathcal{K}_\infty$ *function satisfying* $\lim_{\|s\|\to\infty} \frac{\overline{q}(\|s\|)}{\overline{\alpha}(\|s\|^{\mathfrak{m}})} > 0$*.*

Definition 2.3.4. *Consider a DJDS* Σ_D *and a continuous function* $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_0^+$ *that is twice differentiable on* $\mathbb{R}^n \times \mathbb{R}^n \setminus \triangle$*. Function* V *is called a* δ*-ISS-*$M_{\mathfrak{m}}$ *Lyapunov function for* Σ_D *for* $\mathfrak{m} \geq 1$*, if there exist constants* κ_0*,* κ_1*,* κ_2*,* κ_3 *such that* $\kappa_0 \geq \sum_{i=1}^3 \kappa_i \geq 0$*, a nonnegative function* $\psi : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$*, functions* $\underline{\alpha}, \overline{\alpha}, \overline{\varphi} \in \mathcal{K}_\infty$*, and function* $\hat{\varkappa} \in \mathcal{K}$ *such that: conditions* (i) *and* (ii) *in Definition 2.3.3 hold and* $\forall x, \hat{x}, y, \hat{y}, z, \hat{z}, p, \hat{p} \in \mathbb{R}^n$ *and* $\forall u, \hat{u} \in \mathsf{U}$*,*

$$\begin{aligned}&\mathcal{D}V(x,\hat{x},y,\hat{y},z,\hat{z},p,\hat{p},u,\hat{u})\\ &\quad\leq -\kappa_0 V(x,\hat{x}) - \psi(x,\hat{x}) + \kappa_1 V(y,\hat{y}) + \kappa_2 V(z,\hat{z}) + \kappa_3 V(p,\hat{p}) + \overline{\varphi}(\|u-\hat{u}\|),\end{aligned}$$

and $\psi(x,\hat{x}) \geq \hat{\varkappa}(\|x-\hat{x}\|^{\mathfrak{m}})$ *and* $\lim_{\|s\|\to\infty} \frac{\hat{\varkappa}(\|s\|^{\mathfrak{m}})}{\overline{\alpha}(\|s\|^{\mathfrak{m}})} > 0$*.*

Now we provide the description of δ-ISS-$\mathrm{M}_{\mathfrak{m}}$ for an RJDS Σ_R in terms of existence of δ-ISS-$\mathrm{M}_{\mathfrak{m}}$ Lyapunov functions in the following theorem.

Theorem 2.3.5. *An RJDS* Σ_R *is* δ*-ISS-*$M_{\mathfrak{m}}$ *if it admits a* δ*-ISS-*$M_{\mathfrak{m}}$ *Lyapunov function as in Definition 2.3.3.*

Proof. The proof is inspired by the proof of Theorem 3.1 in [HM09]. Denote $\tilde{\varphi} = \overline{\varphi}(\|\upsilon - \hat{\upsilon}\|_\infty)$ and $\overline{V}_0 = \underline{\alpha}(\mathbb{E}[\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}])$, $\forall \upsilon, \hat{\upsilon} \in \mathcal{U}$ and $\forall \zeta, \hat{\zeta} \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau,0];\mathbb{R}^n)$. By using Lemma 3.2 and 3.3 in [HM09], there exist a constant $a_q > 0$ and a function $\mu_\varkappa \in \mathcal{K}_\infty$ such that $\forall t \geq 0$, $\mathbb{E}[\varkappa(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \geq 2\tilde{\varphi}$ and $\mathbb{E}[\tilde{q}(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] - \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \geq a_q$, whenever $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \geq \mu_\varkappa^{-1}(2\tilde{\varphi})$. Without loss of generality, assume $\mu_\varkappa^{-1}(2\tilde{\varphi}) < \underline{\alpha}(\sup_{-\tau\leq\theta\leq 0} \mathbb{E}[\|\zeta(\theta) - \hat{\zeta}(\theta)\|^{\mathfrak{m}}]) \leq \overline{V}_0$. Let J be the minimal nonnegative integer such that $M_0 = \mu_\varkappa^{-1}(2\tilde{\varphi}) + Ja_q > \overline{V}_0$. Let $\hat{\tau} = \max\{\tau, M_0/\tilde{\varphi}\}$ and $t_j = j\hat{\tau}$ for $j \in [0;J]$. In order to prove the theorem, we need to show

$$\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq \min\{\overline{V}_0, M_j\}, \forall t \geq t_j, \tag{2.3.8}$$

where $M_j = \mu_\varkappa^{-1}(2\tilde{\varphi}) + (J-j)a_q$ and $j \in [0;J]$.

First, we show that $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq \overline{V}_0, \forall t \geq t_0$. Suppose that $t_a := \inf\{t > t_0 \mid \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] > \overline{V}_0\} < \infty$. Since $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))]$ is continuous in time $t \geq 0$, there exist a pair of constants t_b and t_c such that $t_0 \leq t_b \leq t_a < t_c$ and

$$\begin{aligned}\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] &= \overline{V_0}, \qquad t = t_b;\\ \overline{V_0} < \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] &< \overline{V_0} + a_q, \quad t_b < t \leq t_c.\end{aligned} \tag{2.3.9}$$

However, by generalized Itô's formula [Sko09] and condition (2.3.6) in Definition 2.3.3, we have

$$\begin{aligned}\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] &= \mathbb{E}[V(\xi_{\zeta,\upsilon}(t_b), \xi_{\hat{\zeta},\hat{\upsilon}}(t_b))] + \int_{t_b}^t \mathbb{E}[\mathcal{D}V(\xi_{s,\zeta,\upsilon}, \xi_{s,\hat{\zeta},\hat{\upsilon}})]\,\mathrm{d}\,s \qquad (2.3.10)\\ &\leq \overline{V_0} - \tilde{\varphi}(t - t_b) \leq \overline{V_0}\end{aligned}$$

fot all $t \in (t_b, t_c]$, which contradicts (2.3.9). Thus the inequality $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq \overline{V}_0$ must be true for all $t \geq t_0$. Now we show that $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq M_1, \forall t \geq t_1$. Let $t_m := \inf\{t \geq t_0 \mid \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq M_1\} < \infty$. If $t_m > t_1$, then $\forall t \in [t_0, t_1]$, we have

$$\mathbb{E}[\tilde{q}(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \geq \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] + a_q > M_1 + a_q > \overline{V}_0 \geq \mathbb{E}[V(\xi_{\zeta,\upsilon}(t+\theta), \xi_{\hat{\zeta},\hat{\upsilon}}(t+\theta))], \tag{2.3.11}$$

for all $\theta \in [-\tau, 0]$. Using condition (2.3.6) in Definition 2.3.3, inequality (2.3.11) implies

$$\mathbb{E}[\mathcal{D}V(\xi_{t,\zeta,\upsilon}, \xi_{t,\hat{\zeta},\hat{\upsilon}})] \leq -\tilde{\varphi}, \ \forall t \in [t_0, t_1].$$

Consequently, by generalized Itô's formula, we have $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq \overline{V}_0 - \tilde{\varphi}\hat{\tau} < 0$, which contradicts the property of $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \geq 0, \ \forall t \geq 0$. Hence, we must have $t_m \leq t_1$. Let

$$\bar{t}_a := \inf\{t > t_m \mid \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] > M_1\} < \infty.$$

Again as $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))]$ is continuous in $t \geq 0$, there exists constants $\bar{t}_b$ and $\bar{t}_c$ such that $t_1 \leq \bar{t}_b \leq \bar{t}_a < \bar{t}_c$ and

$$\begin{aligned} \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] &= M_1, && t = \bar{t}_b; \\ M_1 < \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] &< M_1 + a_q, && \bar{t}_b < t \leq \bar{t}_c. \end{aligned}$$

By using similar reasoning as before, generalized Itô's formula [Sko09] and condition (2.3.6) in Definition 2.3.3, the assumption results in contradiction, thus we have (2.3.8) for $j = 1$. Now define $t_j := \inf\{t \geq t_{j-1} \mid \mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq M_j\} < \infty$ for $j = 2, 3, \ldots, J$. By similar type of reasoning, we get $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq M_j, \forall t \geq t_j$. Particularly, $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq M_J = \mu_{\varkappa}^{-1}(2\tilde{\varphi}), \forall t \geq t_J$. By following Jensen's inequality, one obtains

$$\mathbb{E}[\|\xi_{\zeta,\upsilon}(t) - \xi_{\hat{\zeta},\hat{\upsilon}}(t)\|^{\mathfrak{m}}] \leq \gamma(\|\upsilon - \hat{\upsilon}\|_{\infty}), \ \forall t \geq t_J, \tag{2.3.12}$$

where $\gamma(s) = \underline{\alpha}^{-1}(\mu_{\varkappa}^{-1}(2\overline{\varphi}(s)))$ for all $s \in \mathbb{R}_0^+$. Now choose a function $\overline{\beta} \in \mathcal{KL}$ such that $\overline{\beta}(\overline{V}_0, t) \geq 2\overline{V}_0 - \frac{\overline{V}_0}{t_J}t, \forall t \in [0, t_J]$. So we have $\mathbb{E}[V(\xi_{\zeta,\upsilon}(t), \xi_{\hat{\zeta},\hat{\upsilon}}(t))] \leq \overline{\beta}(\overline{V}_0, t), \forall t \in [0, t_J]$ which implies

$$\mathbb{E}[\|\xi_{\zeta,\upsilon}(t) - \xi_{\hat{\zeta},\hat{\upsilon}}(t)\|^{\mathfrak{m}}] \leq \beta(\mathbb{E}[\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}], t), \ \forall t \in [0, t_J], \tag{2.3.13}$$

where $\beta(s,t) = \underline{\alpha}^{-1}(\overline{\beta}(\underline{\alpha}(s), t))$ for any $s, t \in \mathbb{R}_0^+$. From (2.3.12) and (2.3.13), one can readily verify inequality (2.3.3) which implies that Σ_R is δ-ISS-$\mathrm{M}_{\mathfrak{m}}$. □

The next corollary proposes similar results as in the previous theorem but for DJDS.

Corollary 2.3.6. *A DJDS Σ_D is δ-ISS-$M_{\mathfrak{m}}$ if it admits a δ-ISS-$M_{\mathfrak{m}}$ Lyapunov function as in Definition 2.3.4.*

Proof. Let $\varkappa(x,\hat{x}) = \frac{1}{1+\kappa_0}\psi(x,\hat{x})$ for all $x,\hat{x}\in\mathbb{R}^n$. Now by considering Definition 2.3.4, we have

$$
\begin{aligned}
&\mathbb{E}[\mathcal{D}V(\phi,\hat{\phi},u,\hat{u})] = \mathbb{E}[\mathcal{D}V(\phi(0),\hat{\phi}(0),\phi(-\tau_1),\hat{\phi}(-\tau_1),\phi(-\tau_2),\hat{\phi}(-\tau_2),\phi(-\tau_3),\hat{\phi}(-\tau_3))]\\
&\leq -\kappa_0\mathbb{E}[V(\phi(0),\hat{\phi}(0))] - \mathbb{E}[\psi(\phi(0),\hat{\phi}(0))] + \kappa_1\mathbb{E}[V(\phi(-\tau_1),\hat{\phi}(-\tau_1))]\\
&\quad + \kappa_2\mathbb{E}[V(\phi(-\tau_2),\hat{\phi}(-\tau_2))] + \kappa_3\mathbb{E}[V(\phi(-\tau_3),\hat{\phi}(-\tau_3))] + \overline{\varphi}(\|u-\hat{u}\|)\\
&\leq -\kappa_0\big(\mathbb{E}[V(\phi(0),\hat{\phi}(0))] + \mathbb{E}[\varkappa(\phi(0),\hat{\phi}(0))]\big) + \kappa_1\mathbb{E}[V(\phi(-\tau_1),\hat{\phi}(-\tau_1))]\\
&\quad + \kappa_2\mathbb{E}[V(\phi(-\tau_2),\hat{\phi}(-\tau_2))] + \kappa_3\mathbb{E}[V(\phi(-\tau_3),\hat{\phi}(-\tau_3))] - \mathbb{E}[\varkappa(\phi(0),\hat{\phi}(0))] + \overline{\varphi}(\|u-\hat{u}\|)\\
&\leq -(\kappa_0 - \sum_{i=1}^{3}\kappa_i)\Big(\mathbb{E}[V(\phi(0),\hat{\phi}(0))] + \mathbb{E}[\varkappa(\phi(0),\hat{\phi}(0))]\Big) - \mathbb{E}[\varkappa(\phi(0),\hat{\phi}(0))] + \overline{\varphi}(\|u-\hat{u}\|)\\
&\leq -\mathbb{E}[\varkappa(\phi(0),\hat{\phi}(0))] + \overline{\varphi}(\|u-\hat{u}\|),
\end{aligned}
$$

for all $t\geq 0$ and $\phi,\hat{\phi}\in\mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau,0];\mathbb{R}^n)$ satisfying condition (2.3.7) in Definition 2.3.3 with function $\tilde{q}(\phi(0),\hat{\phi}(0)) := V(\phi(0),\hat{\phi}(0)) + \varkappa(\phi(0),\hat{\phi}(0))$. Moreover, functions $\tilde{\varkappa}(s) = \overline{q}(s) = \frac{\hat{\varkappa}(s)}{1+\kappa_0}$, $\forall s\in\mathbb{R}_0^+$, satisfy properties required in condition (iii) in Definition 2.3.3. Therefore, V satisfies all the conditions in Definition 2.3.3. Thus by following Theorem 2.3.5, we obtain that Σ_D is δ-ISS-$\mathrm{M}_{\mathfrak{m}}$. □

In the following lemma, we provide a similar result as in Corollary 2.3.6 but tailored to linear delayed jump-diffusion systems in which sufficient conditions boil down to a matrix inequality.

Lemma 2.3.7. *Consider a DJDS Σ_D as given in (2.3.2), where for all $x,y,z,p\in\mathbb{R}^n$ and $u\in\mathsf{U}$, $F(x,y,u) := A_1x + A_2y + Bu$, for some $A_1,A_2\in\mathbb{R}^{n\times n}$ and $B\in\mathbb{R}^{n\times m}$, $G(x,z) := [G_1x\ G_2x\ \cdots\ G_{\check{r}}x] + [\overline{G}_1z\ \overline{G}_2z\ \cdots\ \overline{G}_{\check{r}}z]$ and $R(x,p) := [R_1x\ R_2x\ \cdots\ R_{\tilde{r}}x] + [\overline{R}_1p\ \overline{R}_2p\ \cdots\ \overline{R}_{\tilde{r}}p]$, for some $G_i,\overline{G}_i,R_i,\overline{R}_i\in\mathbb{R}^{n\times n}$. Then, system Σ_D is δ-ISS-M_2 if there exist constants $c_1,c_2,c_3,c_4,c_5\in\mathbb{R}^+$ satisfying $c_1 > \sum_{i=2}^{4}c_i$ and*

$$
\begin{aligned}
&\begin{bmatrix}
\Delta & PA_2 & \sum_{i=1}^{\check{r}} G_i^TP\overline{G}_i & \sum_{i=1}^{\tilde{r}}\lambda_i(P\overline{R}_i + R_i^TP\overline{R}_i) & PB\\
A_2^TP & 0 & 0 & 0 & 0\\
\sum_{i=1}^{\check{r}}\overline{G}_i^TPG_i & 0 & \sum_{i=1}^{\check{r}}\overline{G}_i^TP\overline{G}_i & 0 & 0\\
\sum_{i=1}^{\tilde{r}}\lambda_i(\overline{R}_i^TP + \overline{R}_i^TPR_i) & 0 & 0 & \sum_{i=1}^{\tilde{r}}\lambda_i\overline{R}_i^TP\overline{R}_i & 0\\
B^TP & 0 & 0 & 0 & 0
\end{bmatrix}\\
&\preceq \begin{bmatrix}
-c_1P & 0 & 0 & 0 & 0\\
0 & c_2P & 0 & 0 & 0\\
0 & 0 & c_3P & 0 & 0\\
0 & 0 & 0 & c_4P & 0\\
0 & 0 & 0 & 0 & c_5I_m
\end{bmatrix},
\end{aligned}
\tag{2.3.14}
$$

where P is a symmetric positive definite matrix and $\Delta = PA_1 + A_1^T P + \sum_{i=1}^{\check{r}} G_i^T P G_i + \sum_{i=1}^{\tilde{r}} \lambda_i (PR_i + R_i^T P + R_i^T P R_i)$.

Proof. Consider a function $V : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}_0^+$ given by

$$V(x, \hat{x}) := \frac{1}{2}(x - \hat{x})^T P (x - \hat{x}), \quad \forall x, \hat{x} \in \mathbb{R}^n, \tag{2.3.15}$$

where P is a symmetric positive definite matrix. One can readily verify that function V in (2.3.15) satisfies properties (i) and (ii) in Definition 2.3.3 with functions $\underline{\alpha}(s) := \frac{1}{2}\lambda_{\min}(P)s^2$ and $\overline{\alpha}(s) := \frac{1}{2}\lambda_{\max}(P)s^2$ for all $s \in \mathbb{R}_0^+$ and $\mathfrak{m} = 2$. By considering the infinitesimal generator in (2.3.5) associated with the considered linear delayed jump-diffusion system, Lipschitz assumptions, Young's inequality, consistency of norms, and (2.3.14), one can obtain the following chains of inequalities

$$\begin{aligned}
&\mathcal{D}V(x, \hat{x}, y, \hat{y}, z, \hat{z}, p, \hat{p}, u, \hat{u}) = (x - \hat{x})^T P (A_1(x - \hat{x}) + A_2(y - \hat{y}) + B(u - \hat{u})) \\
&\quad + \frac{1}{2} \sum_{i=1}^{\check{r}} \Big(G_i(x - \hat{x}) + \overline{G}_i(z - \hat{z}) \Big)^T P \Big(G_i(x - \hat{x}) + \overline{G}_i(z - \hat{z}) \Big) \\
&\quad + \frac{1}{2} \sum_{i=1}^{\tilde{r}} \lambda_i \Big[\Big((x - \hat{x}) + R_i(x - \hat{x}) + \overline{R}_i(p - \hat{p}) \Big)^T \\
&\quad P \Big((x - \hat{x}) + R_i(x - \hat{x}) + \overline{R}_i(p - \hat{p}) \Big) - (x - \hat{x})^T P (x - \hat{x}) \Big] \\
&\leq (x - \hat{x})^T P (A_1(x - \hat{x}) + A_2(y - \hat{y}) + B(u - \hat{u})) \\
&\quad + \frac{1}{2} \sum_{i=1}^{\check{r}} \Big[(x - \hat{x})^T G_i^T P G_i (x - \hat{x}) + (x - \hat{x})^T G_i^T P \overline{G}_i (z - \hat{z}) + (z - \hat{z})^T \overline{G}_i^T P G_i (x - \hat{x}) \\
&\quad + (z - \hat{z})^T \overline{G}_i^T P \overline{G}_i (z - \hat{z}) \Big] + \frac{1}{2} \sum_{i=1}^{\tilde{r}} \lambda_i \Big[(x - \hat{x})^T P (R_i(x - \hat{x}) + \overline{R}_i(p - \hat{p})) + (R_i(x - \hat{x}) \\
&\quad + \overline{R}_i(p - \hat{p}))^T P (x - \hat{x}) + (R_i(x - \hat{x}) + \overline{R}_i(p - \hat{p}))^T P (R_i(x - \hat{x}) + \overline{R}_i(p - \hat{p})) \Big] \\
&\leq (x - \hat{x})^T P (A_1(x - \hat{x}) + A_2(y - \hat{y}) + B(u - \hat{u})) \\
&\quad + \frac{1}{2} \sum_{i=1}^{\check{r}} \Big[(x - \hat{x})^T G_i^T P G_i (x - \hat{x}) + (x - \hat{x})^T G_i^T P \overline{G}_i (z - \hat{z}) \\
&\quad + (z - \hat{z})^T \overline{G}_i^T P G_i (x - \hat{x}) + (z - \hat{z})^T \overline{G}_i^T P \overline{G}_i (z - \hat{z}) \Big] + \frac{1}{2} \sum_{i=1}^{\tilde{r}} \lambda_i \Big[(x - \hat{x})^T P (R_i(x - \hat{x}) \\
&\quad + \overline{R}_i(p - \hat{p})) + ((x - \hat{x})^T R_i^T + (p - \hat{p})^T \overline{R}_i^T) P (x - \hat{x}) + (x - \hat{x})^T R_i^T P R_i (x - \hat{x}) \\
&\quad + (x - \hat{x})^T R_i^T P \overline{R}_i (p - \hat{p}) + (p - \hat{p})^T \overline{R}_i^T P R_i (x - \hat{x}) + (p - \hat{p})^T \overline{R}_i^T P \overline{R}_i (p - \hat{p})) \Big]
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{1}{2}\begin{bmatrix} x-\hat{x} \\ y-\hat{y} \\ z-\hat{z} \\ p-\hat{p} \\ u-\hat{u} \end{bmatrix}^T \begin{bmatrix} \Delta & PA_2 & \sum\limits_{i=1}^{\check{r}} G_i^T P\overline{G}_i & \sum\limits_{i=1}^{\tilde{r}} \lambda_i(P\overline{R}_i + R_i^T P\overline{R}_i) & PB \\ A_2^T P & 0 & 0 & 0 & 0 \\ \sum\limits_{i=1}^{\check{r}} \overline{G}_i^T P G_i & 0 & \sum\limits_{i=1}^{\check{r}} \overline{G}_i^T P\overline{G}_i & 0 & 0 \\ \sum\limits_{i=1}^{\tilde{r}} \lambda_i(\overline{R}_i^T P + \overline{R}_i^T P R_i) & 0 & 0 & \sum\limits_{i=1}^{\tilde{r}} \lambda_i \overline{R}_i^T P\overline{R}_i & 0 \\ B^T P & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x-\hat{x} \\ y-\hat{y} \\ z-\hat{z} \\ p-\hat{p} \\ u-\hat{u} \end{bmatrix} \\
&\leq \frac{1}{2}\big(- c_1(x-\hat{x})^T P(x-\hat{x}) + c_2(y-\hat{y})^T P(y-\hat{y}) + c_3(z-\hat{z})^T P(z-\hat{z}) \\
&\quad + c_4(p-\hat{p})^T P(p-\hat{p}) + c_5\|u-\hat{u}\|^2\big) \\
&\leq - c_1 V(x,\hat{x}) + c_2 V(y,\hat{y}) + c_3 V(z,\hat{z}) + c_4 V(p,\hat{p}) + \frac{c_5}{2}\|u-\hat{u}\|^2.
\end{aligned}
$$

Thus by following the proof of Corollary 2.3.6 with $\kappa_0 = \sum_{i=2}^4 c_i$, $\psi(x,\hat{x}) = (c_1 - \kappa_0)V(x,\hat{x})$, $\varkappa(x,\hat{x}) = \frac{1}{1+\kappa_0}\psi(x,\hat{x}) = \kappa V(x,\hat{x})$, where $\kappa = \frac{c_1-\kappa_0}{1+\kappa_0}$, $\forall t \geq 0$, $\forall \zeta, \hat{\zeta} \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau,0];\mathbb{R}^n)$, and $\forall \phi, \hat{\phi} \in \mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau,0];\mathbb{R}^n)$ satisfying (2.3.7), one obtains

$$
\mathbb{E}[\mathcal{D}V(\phi,\hat{\phi},u,\hat{u})] \leq -\kappa\mathbb{E}[V(\xi_{\zeta,\upsilon}(t),\xi_{\hat{\zeta},\hat{\upsilon}}(t))] + \frac{c_5}{2}\|u-\hat{u}\|^2. \tag{2.3.16}
$$

By using generalized Ito's formula, (2.3.16), and condition (ii) in Definition 2.3.3, we have

$$
\begin{aligned}
\mathbb{E}[V(\xi_{\zeta,\upsilon}(t),\xi_{\hat{\zeta},\hat{\upsilon}}(t))] &= \mathbb{E}[V(\xi_{\zeta,\upsilon}(0),\xi_{\hat{\zeta},\hat{\upsilon}}(0)) + \int_0^t \mathcal{D}V(\phi,\hat{\phi},u,\hat{u})\,\mathrm{d}\,s] \\
&= \mathbb{E}[V(\xi_{\zeta,\upsilon}(0),\xi_{\hat{\zeta},\hat{\upsilon}}(0))] + \int_0^t \mathbb{E}[\mathcal{D}V(\phi,\hat{\phi},u,\hat{u})]\,\mathrm{d}\,s \\
&\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\xi_{\zeta,\upsilon}(0) - \xi_{\hat{\zeta},\hat{\upsilon}}(0)\|^2] + \int_0^t \mathbb{E}[\mathcal{D}V(\phi,\hat{\phi})]\,\mathrm{d}\,s \\
&\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}] + \int_0^t \Big(- \kappa\mathbb{E}[V(\xi_{\zeta,\upsilon}(s),\xi_{\hat{\zeta},\hat{\upsilon}}(s))] + \frac{c_5}{2}\|\upsilon(s)-\hat{\upsilon}(s)\|^2\Big)\,\mathrm{d}\,s \\
&\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}] + -\kappa\int_0^t \mathbb{E}[V(\xi_{\zeta,\upsilon}(s),\xi_{\hat{\zeta},\hat{\upsilon}}(s))]\,\mathrm{d}\,s + \frac{c_5}{2}\|\upsilon-\hat{\upsilon}\|^2_\infty t,
\end{aligned}
$$

which, by virtue of Gronwall's inequality, leads to

$$
\begin{aligned}
\mathbb{E}[V(\xi_{\zeta,\upsilon}(t),\xi_{\hat{\zeta},\hat{\upsilon}}(t))] &\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}]\mathsf{e}^{-\kappa t} + \frac{c_5 t\mathsf{e}^{-\kappa t}}{2}\|\upsilon-\hat{\upsilon}\|^2_\infty \\
&\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}]\mathsf{e}^{-\kappa t} + \frac{c_5}{2\mathsf{e}\kappa}\|\upsilon-\hat{\upsilon}\|^2_\infty.
\end{aligned}
$$

Now by using condition (ii) in Definition 2.3.3, one obtains

$$
\begin{aligned}
\frac{\lambda_{\min}(P)}{2}\mathbb{E}[\|\xi_{\zeta,\upsilon}(t) - \xi_{\hat{\zeta},\hat{\upsilon}}(t)\|^2] &\leq \mathbb{E}[V(\xi_{\zeta,\upsilon}(t),\xi_{\hat{\zeta},\hat{\upsilon}}(t))] \\
&\leq \frac{\lambda_{\max}(P)}{2}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}]\mathsf{e}^{-\kappa t} + \frac{c_5}{2\mathsf{e}\kappa}\|\upsilon-\hat{\upsilon}\|^2_\infty,
\end{aligned}
$$

and, hence,

$$\mathbb{E}[\|\xi_{\zeta,\upsilon}(t)-\xi_{\hat{\zeta},\hat{\upsilon}}(t)\|^2]\leq\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}\mathbb{E}[\|\zeta-\hat{\zeta}\|^2_{[-\tau,0]}]\mathsf{e}^{-\kappa t}+\frac{c_5}{\lambda_{\min}(P)\mathsf{e}\kappa}\|\upsilon-\hat{\upsilon}\|^2_\infty.$$

Therefore, by introducing functions β and γ as

$$\beta(s,t)=\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}\mathsf{e}^{-\kappa t}s,\gamma(s)=\frac{c_5}{\lambda_{\min}(P)\mathsf{e}\kappa}s^2, \tag{2.3.17}$$

for any $s,t\in\mathbb{R}^+_0$, inequality (2.3.3) is satisfied. □

Remark 2.3.8. *For fixed values of c_i, $i=[1;5]$, the inequality* (2.3.14) *boils down to a linear matrix inequality (LMI) which can be solved efficiently using semidefinite programming. One may also solve a bilinear matrix inequality (BMI) (locally) using a $V-K$ iteration [GB94]. That is, for fixed values c_i, $i=[1;5]$, we find matrix P satisfying the LMI, and then for a fixed P we find constants c_i, $i=[1;5]$, to maximize the value of $c_1-\sum_{i=2}^4 c_i$, and we iterate until there is no improvement in the value of $c_1-\sum_{i=2}^4 c_i$.*

3 Controller Synthesis for Retarded Jump-Diffusion Systems

This chapter is concerned with the automated, correct-by-construction, controller synthesis scheme for a class of infinite-dimensional stochastic systems, namely, retarded jump-diffusion systems. Under the assumption of incremental stability as discussed in the previous chapter, first, we construct finite abstractions approximately bisimilar to non-stochastic retarded systems corresponding to the original systems. Then, we provide a result on quantifying the distance between the output trajectory of the obtained finite abstraction and that of the original retarded jump-diffusion system in a probabilistic setting. Using the proposed result, one can refine the control policy synthesized using finite abstractions to the original systems while providing a guarantee on the probability of satisfaction of high-level complex specification.

3.1 Introduction

Finite (a.k.a. symbolic) abstraction techniques have gained significant attentions in the last few years since they provide tools for automated, correct-by-construction, controller synthesis for several classes of control systems. In particular, such abstractions provide approximate models that are related to concrete systems by aggregating concrete states and inputs to the symbolic ones. Having such finite abstractions, one can make use of the existing automata-theoretic techniques [MPS95] to synthesize hybrid controllers enforcing rich complex specifications (usually expressed as linear temporal logic formulae or as automata on infinite strings) over the original systems.

3.1.1 Related Literature

Bisimilar finite abstractions for non-stochastic and stochastic systems

In the past few years, there have been several results providing bisimilar finite abstractions for various continuous-time continuous-space non-stochastic as well as stochastic systems. The results include construction of approximately bisimilar abstractions for incrementally stable control systems [PGT08], switched systems [GPT09], stochastic control systems [ZMEM$^+$14], and randomly switched stochastic systems [ZA14]. However, the abstractions obtained in these results are based on state-space quantization

which suffer severely from the *curse of dimensionality*, i.e., the computational complexity increases exponentially with respect to the state-space dimension of the concrete system.

Finite abstractions without state-space discretization

To alleviate the issue of the curse of dimensionality, [LCGG13] proposed an alternative approach for constructing approximately bisimilar abstractions for incrementally stable non-stochastic switched systems without discretizing the state-space. The concept is further extended to provide finite abstractions for incrementally stable stochastic switched systems [ZAG15], stochastic control systems [ZTA16], and infinite dimensional non-stochastic control systems [Gir14]. For a comparison between state-space discretization based and free approaches, we refer the interested readers to the discussion in [ZTA16, Section 5.4].

Finite abstractions for retarded systems

Retarded stochastic systems are widely used to model various processes in finance, ecology, medical, and engineering (see examples in [Sha13, BSP08, KM13]). However, the construction of symbolic models for such classes of systems is still unaddressed due to underlying challenges such as infinite-dimensional functional state-space and dependency on state history. Recent results by [PPDBT10] and [PPDB15] provide the construction of abstractions for incrementally stable non-stochastic time-delayed systems by spline-based approximation of functional spaces. However, the proposed results are complex from the implementation point of view and also suffer from the curse of dimensionality with respect to the state-space dimension of the concrete system. This motivates our work in this chapter to provide a scheme for the construction of finite abstractions for a class of infinite-dimensional stochastic systems, namely, retarded jump-diffusion systems without discretizing the state-space.

3.1.2 Contributions

Under the incremental stability property over retarded jump-diffusion systems, we provide a construction of finite abstractions which are approximately bisimilar to the corresponding non-stochastic version of retarded systems. Then, under some mild assumptions over incremental Lyapunov functions, we obtain a lower bound on the probability such that the distance between output trajectories of the obtained finite abstraction and those of the original retarded jump-diffusion system remains close over a finite time horizon. One can leverage the proposed probability closeness to synthesize a control policy using constructed finite abstractions and refine it back to the original system while providing a guarantee on the probability of satisfaction over the original system. Further, we demonstrate the effectiveness of the proposed results by synthesizing a controller keeping temperatures in a comfort zone in a ten-room building modeled as a linear delayed jump-diffusion system.

3.2 Preliminaries

3.2.1 Non-stochastic Retarded System

In order to provide results on the construction of finite abstraction and given a RJDS Σ_R (defined in 2.3.1), we introduce the corresponding non-stochastic retarded systems (denoted by $\overline{\Sigma}_R$) obtained by removing diffusion and reset terms (that is, $\overline{g}$ and $\overline{r}$ in (2.3.1)). From now onwards, we use notation $\overline{\xi}_{\zeta,\upsilon}(t)$ to denote the value of a trajectory of $\overline{\Sigma}_R$ in $\mathbb{R}^n$ and $\overline{\xi}_{t,\zeta,\upsilon}$ to denote the solution of $\overline{\Sigma}_R$ in $\mathcal{C}([-\tau,0];\mathbb{R}^n)$ at time $t \in \mathbb{R}_0^+$ started from the non-stochastic initial condition $\zeta \in \mathcal{C}^b_{\mathcal{F}_0}([-\tau,0];\mathbb{R}^n)$, where $\mathcal{F}_0$ is the trivial sigma-algebra, and under input signal υ. Now, we provide a technical lemma which is used later to show a relation between non-stochastic retarded systems $\overline{\Sigma}_R$ and their symbolic models.

Lemma 3.2.1. *Consider an incrementally input-to-state stable non-stochastic retarded systems $\overline{\Sigma}_R$ corresponding to a δ-ISS-$M_{\mathfrak{m}}$ RJDS Σ_R for $\mathfrak{m} \geq 1$, that is for any $t \in \mathbb{R}_0^+$, any $\zeta, \hat{\zeta} \in \mathcal{C}([-\tau,0];\mathbb{R}^n)$, and any $\upsilon, \hat{\upsilon} \in \mathcal{U}$, it satisfies*

$$\|\overline{\xi}_{\zeta,\upsilon}(t) - \overline{\xi}_{\hat{\zeta},\hat{\upsilon}}(t)\|^{\mathfrak{m}} \leq \beta(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, t) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty), \tag{3.2.1}$$

where β and γ are the functions appearing in (2.3.3). Then there exists a function $\tilde{\beta} \in \mathcal{KL}$ such that the following inequality holds:

$$\|\overline{\xi}_{t,\zeta,\upsilon} - \overline{\xi}_{t,\hat{\zeta},\hat{\upsilon}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \tilde{\beta}(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, t) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty), \tag{3.2.2}$$

where $\tilde{\beta}(s,t) = \mathsf{e}^{-(t-\tau)}s + \beta(s, \max\{0, t-\tau\})$.

Proof. The proof is inspired by the proof of Theorem 3 in [PPDBT10]. From inequality (3.2.1), we obtain the following inequalities:

$$\|\xi_{t,\zeta,\upsilon} - \xi_{t,\hat{\zeta},\hat{\upsilon}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \beta(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, t-\tau) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty), \ \forall t \geq \tau, \tag{3.2.3}$$

and

$$\|\xi_{t,\zeta,\upsilon} - \xi_{t,\hat{\zeta},\hat{\upsilon}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]} + \beta(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, 0) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty), \ \forall t \in [0,\tau). \tag{3.2.4}$$

Moreover, we also have

$$\mathsf{e}^{-(t-\tau)}\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]} \geq \|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, \quad \forall t \in [0,\tau). \tag{3.2.5}$$

Inequalities (3.2.3) and (3.2.4) along with (3.2.5) yield

$$\|\xi_{t,\zeta,\upsilon} - \xi_{t,\hat{\zeta},\hat{\upsilon}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \mathsf{e}^{-(t-\tau)}\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]} + \beta(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, \max\{0, t-\tau\}) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty),$$

for all $t \geq 0$. One can rewrite the last inequality as

$$\|\xi_{t,\zeta,\upsilon} - \xi_{t,\hat{\zeta},\hat{\upsilon}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \tilde{\beta}(\|\zeta - \hat{\zeta}\|^{\mathfrak{m}}_{[-\tau,0]}, t) + \gamma(\|\upsilon - \hat{\upsilon}\|_\infty),$$

for all $t \geq 0$, where $\tilde{\beta}(s,t) := \mathsf{e}^{-(t-\tau)}s + \beta(s, \max\{0, t-\tau\})$ is a $\mathcal{KL}$ function. □

3.2.2 Systems and Approximate Equivalence Relations

First, We recall the notion of *system* introduced in [Tab09] which later serves as a unified modeling framework for both retarded jump-diffusion systems Σ_D and their finite abstractions.

Definition 3.2.2. *A systems is a tuple $S = (X, X_0, U, \longrightarrow, Y, H)$ where X is a set of states (possibly infinite), $X_0 \subseteq X$ is a set of initial states, U is a set of inputs (possibly infinite), $\longrightarrow \subseteq X \times U \times X$ is a transition relation, Y is a set of outputs, and $H : X \to Y$ is an output map.*

We denote $x \xrightarrow{u} x'$ as an alternative representation for a transition $(x, u, x') \in \longrightarrow$, where state x' is called a u-successor (or simply successor) of state x, for some input $u \in U$. Moreover, a system S is said to be

- *metric*, if the output set Y is equipped with a metric $\mathbf{d} : Y \times Y \to \mathbb{R}_0^+$.
- *finite* (or *symbolic*), if X and U are finite.
- *deterministic*, if there exists at most a u-successor of x, for any $x \in X$ and $u \in U$.
- *nonblocking*, if for any $x \in X$, there exists some u-successor of x, for some $u \in U$.

For a system S, the finite state-run generated from initial state $x_0 \in X_0$ is a finite sequence of transitions:

$$x_0 \xrightarrow{u_0} x_1 \xrightarrow{u_1} \cdots \xrightarrow{u_{k-2}} x_{k-1} \xrightarrow{u_{k-1}} x_k, \tag{3.2.6}$$

such that $x_i \xrightarrow{u_i} x_{i+1}$, for $i \in [0; k-1]$. The associated finite output-run is given by $y_i = H(x_i)$, for $i \in [0; k-1]$. These finite runs can be directly extended to infinite runs as well.

Now, we provide the notion of approximate (bi)simulation relation between two systems, introduced in [GP07], which is later used for analyzing and synthesizing controllers for retarded jump-diffusion systems Σ_D.

Definition 3.2.3. *Let $S_1 = (X_1, X_{10}, U_1, \underset{1}{\longrightarrow}, Y_1, H_1)$ and $S_2 = (X_2, X_{20}, U_2, \underset{2}{\longrightarrow}, Y_2, H_2)$ be two metric systems having the same output sets $Y_1 = Y_2$ and metric $\mathbf{d}$. For $\varepsilon \in \mathbb{R}_0^+$, a relation $\mathcal{R} \subseteq X_1 \times X_2$ is said to be an ε-approximate bisimulation relation between S_1 and S_2 if it satisfies the following conditions:*

(i) $\forall (x_1, x_2) \in \mathcal{R}$, *we have* $\mathbf{d}(H_1(x_1), H_2(x_2)) \leq \varepsilon$;

(ii) $\forall (x_1, x_2) \in \mathcal{R}$, $x_1 \xrightarrow[1]{u_1} x_1'$ *in* S_1 *implies* $x_2 \xrightarrow[2]{u_2} x_2'$ *in* S_2 *satisfying* $(x_1', x_2') \in \mathcal{R}$;

(iii) $\forall (x_1, x_2) \in \mathcal{R}$, $x_2 \xrightarrow[2]{u_2} x_2'$ *in* S_2 *implies* $x_1 \xrightarrow[1]{u_1} x_1'$ *in* S_1 *satisfying* $(x_1', x_2') \in \mathcal{R}$.

If we remove condition (iii), then $\mathcal{R} \subseteq X_1 \times X_2$ is said to be an ε-approximate simulation relation from S_1 to S_2.

The system S_1 is ε-approximate bisimilar to S_2, denoted by $S_1 \cong_{\mathcal{S}}^{\varepsilon} S_2$, if there exists an ε-approximate bisimulation relation $\mathcal{R}$ between S_1 and S_2 such that: $\forall x_{10} \in X_{10}$, $\exists x_{20} \in X_{20}$ with $(x_{10}, x_{20}) \in \mathcal{R}$ and $\forall x_{20} \in X_{20}$, $\exists x_{10} \in X_{10}$ with $(x_{10}, x_{20}) \in \mathcal{R}$.

In order to present the main results in this chapter, we need to employ the notion of system as an abstract representation of a retarded jump-diffusion system. First, we define a metric system associated with the retarded jump-diffusion system Σ_R, denoted by $S(\Sigma_R) = (X, X_0, U, \longrightarrow, Y, H)$, where

- X is the set of all $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$-valued random variables defined on the probability space $(\Omega, \mathcal{F}, \mathbb{P})$;
- X_0 is a subset of $\mathcal{C}^b_{\mathcal{F}_0}([-\tau, 0]; \mathbb{R}^n)$;
- $U = \mathcal{U}$;
- $\zeta \xrightarrow{\upsilon} \zeta'$ if ζ and ζ' are measurable in $\mathcal{F}_t$ and $\mathcal{F}_{t+h}$, respectively, for some $t \in \mathbb{R}^+_0$ and $h \in \mathbb{R}^+$, and there exists a solution $\xi_t \in \mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau, 0]; \mathbb{R}^n)$ of Σ_R satisfying $\xi_t = \zeta$ and $\xi_{h,\zeta,\upsilon} = \zeta'$ $\mathbb{P}$-a.s.;
- $Y = X$;
- $H(\zeta) = \zeta$.

From now on, we restrict our attention to the sampled-data system, where control signals (in Σ_R) are piecewise-constant over intervals of length $h \in \mathbb{R}^+$, i.e.

$$\mathcal{U}_h = \{\upsilon \in \mathcal{U} \mid \upsilon(t) = \upsilon(ih), t \in [ih, (i+1)h), i \in \mathbb{N}_0\}.$$

The metric systems associated with the sampled-data retarded jump-diffusion systems can be defined as $S_h(\Sigma_R) = (X_h, X_{h0}, U_h, \underset{h}{\longrightarrow}, Y_h, H_h)$, where $X_h = X$, $X_{h0} = X_0$, $U_h = \mathcal{U}_h$, $Y_h = Y$, $H_h = H$, and $\zeta_h \xrightarrow[h]{\upsilon_h} \zeta'_h$ if ζ_h and ζ'_h are measurable in $\mathcal{F}_{ih}$ and $\mathcal{F}_{(i+1)h}$, respectively, for some $i \in \mathbb{N}_0$, and there exists a solution $\xi_t \in \mathbf{L}^{\mathfrak{m}}_{\mathcal{F}_t}([-\tau, 0]; \mathbb{R}^n)$ of Σ_R satisfying $\xi_t = \zeta_h$ and $\xi_{h,\zeta_h,\upsilon_h} = \zeta'_h$ $\mathbb{P}$-a.s. In other words, a finite state-run of $S_h(\Sigma_R)$, represented by $\zeta_0 \xrightarrow[h]{\upsilon_0} \zeta_1 \xrightarrow[h]{\upsilon_1} \cdots \xrightarrow[h]{\upsilon_{k-1}} \zeta_k$, where $\upsilon_i \in U_h$ and $\zeta_{i+1} = \xi_{h,\zeta_i,\upsilon_i}$ $\mathbb{P}$-a.s. for $i \in [0; k-1]$, captures solutions of RJDS Σ_R at the sampling times $t = 0, h, \ldots, kh$, started from $\zeta_0 \in X_0$ and resulting from control input υ obtained by the concatenation of input signals $\upsilon_i \in \mathcal{U}_h$. Moreover, the corresponding finite output-run is $\{y_0, y_1, \ldots, y_k\}$. Similarly, we consider metric systems corresponding to non-stochastic sampled-data retarded systems denoted by $S_h(\overline{\Sigma}_R) = (\overline{X}_h, \overline{X}_{h0}, \overline{U}_h, \underset{h}{\longrightarrow}, \overline{Y}_h, \overline{H}_h)$ where $\overline{X}_h = \mathcal{C}([-\tau, 0]; \mathbb{R}^n)$, $\overline{X}_{h0} \subseteq \overline{X}_h$, $\overline{U}_h = \mathcal{U}_h$, $\overline{Y}_h = \overline{X}_h$, $\overline{H}_h(\zeta) = \zeta$, and $\zeta_h \xrightarrow[h]{\upsilon_h} \zeta'_h$ if $\zeta'_h = \overline{\xi}_{h,\zeta_h,\upsilon_h}$. For later use, we represent an $\mathbb{R}^n$-valued output at the k^{th} sampling instance starting from initial state ζ under input signal υ_h by $\overline{\xi}_{\zeta,\upsilon_h}(kh)$.

3.3 Finite Dimensional Abstractions

In this section, we introduce a *finite dimensional abstraction* for $S_h(\overline{\Sigma}_R)$. Consider metric systems associated with the sampled-data retarded systems $S_h(\overline{\Sigma}_R)$ and consider triple $\rho = (h, N, \zeta_s)$ of parameters, where $h \in \mathbb{R}^+$ is the sampling time, $N \in \mathbb{N}$ is a temporal horizon, and $\zeta_s \in \mathcal{C}([-\tau, 0]; \mathbb{R}^n)$ is a source state. Let us define a metric system as

$$S_\rho(\overline{\Sigma}_R) = (X_\rho, X_{\rho 0}, U_\rho, \underset{\rho}{\longrightarrow}, Y_\rho, H_\rho),$$

where

- $X_\rho = \mathsf{U}$, $X_{\rho 0} = X_\rho$, $U_\rho = \mathsf{U}$, $Y_\rho = Y_h$;
- $x_\rho \xrightarrow[\rho]{u_\rho} x'_\rho$, where $x_\rho = (u_1, u_2, \ldots, u_N) \in X_\rho$, if and only if $x'_\rho = (u_2, \ldots, u_N, u_\rho)$;
- $H_\rho(x_\rho) = \overline{\xi}_{Nh,\zeta_s,x_\rho}$.

Here, we abuse notation by identifying $x_\rho = (u_1, u_2, \ldots, u_N) \in [\mathsf{U}]_\eta^N$ as an input curve $\upsilon : [0, Nh) \to [\mathsf{U}]_\eta$ such that $\upsilon(t) = u_k$ for any $t \in [(k-1)h, kh)$ for $k \in [1; N]$ in $\overline{\xi}_{Nh,\zeta_s,x_\rho}$. We use similar notations in the rest of the chapter as well. Notice that the system $S_\rho(\overline{\Sigma}_R)$ is deterministic, non-blocking, and *finite dimensional* (but not necessarily symbolic unless U is a finite set). Note that H_ρ is the output map from non-stochastic state $x_\rho \in X_\rho$ to a $\mathcal{C}([-\tau, 0]; \mathbb{R}^n)$-valued solution process $\overline{\xi}_{Nh,\zeta_s,x_\rho}$ and corresponding $\mathbb{R}^n$-valued solution is represented by $\overline{\xi}_{\zeta_s,x_\rho}(Nh)$.

The next theorem provides the results on the construction of finite dimensional abstractions which are approximately bisimilar to $S_h(\overline{\Sigma}_R)$.

Theorem 3.3.1. *Consider a retarded system $\overline{\Sigma}_R$ corresponding to δ-ISS-$M_\mathfrak{m}$ RJDS Σ_R for $\mathfrak{m} \geq 1$. Given any $\varepsilon > 0$, let the sampling time h, temporal horizon N, and source state ζ_s be such that*

$$\tilde{\beta}(\varepsilon, h)) + \tilde{\beta}(\mathcal{Z}(\zeta_s), Nh) \leq \varepsilon, \tag{3.3.1}$$

where $\mathcal{Z}(\zeta_s) = \sup_{u_1 \in \mathsf{U}} \|\overline{\xi}_{h,\zeta_s,u_1} - \zeta_s\|^\mathfrak{m}_{[-\tau,0]}$. Then, the relation

$$\mathcal{R}_1 = \{(\zeta, x_\rho) \in \overline{X}_h \times X_\rho \mid \|\overline{H}_h(\zeta) - H_\rho(x_\rho)\|^\mathfrak{m}_{[-\tau,0]} \leq \varepsilon\},$$

is an ε-approximate bisimulation relation between $S_h(\overline{\Sigma}_R)$ and $S_\rho(\overline{\Sigma}_R)$.

Proof. Consider any $(\zeta, x_\rho) \in \mathcal{R}_1$, where $\zeta \in \overline{X}_h$ and $x_\rho = (u_1, u_2, \ldots, u_N) \in X_\rho$. Then we have $\|\overline{H}_h(\zeta) - H_\rho(x_\rho)\|^\mathfrak{m}_{[-\tau,0]} \leq \varepsilon$. Thus condition (i) in Definition 3.2.3 holds. Now we show that condition (ii) in Definition 3.2.3 holds. Consider any $\upsilon_h : [0, h[\to u_h$ for some $u_h \in \mathsf{U}$ and $\zeta' = \overline{\xi}_{h,\zeta,\upsilon_h}$. Consider $u_\rho = u_h$ and $x'_\rho = (u_2, \ldots, u_N, u_\rho)$ and let $\overline{x}_\rho = (u_1, u_2, \ldots, u_N, u_\rho)$ denote input sequence in U^{N+1}. With the help of triangle inequality and (3.2.2) one obtains the following chains of inequalities:

$$\begin{aligned}
&\|\overline{H}_h(\zeta') - H_\rho(x'_\rho)\|^\mathfrak{m}_{[-\tau,0]} = \|\overline{H}_h(\zeta') - H_\rho(\overline{x}_\rho) + H_\rho(\overline{x}_\rho) - H_\rho(x'_\rho)\|^\mathfrak{m}_{[-\tau,0]} \\
&= \|\overline{\xi}_{h,\zeta,\upsilon_h} - \overline{\xi}_{(N+1)h,\zeta_s,\overline{x}_\rho} + \overline{\xi}_{(N+1)h,\zeta_s,\overline{x}_\rho} - \overline{\xi}_{Nh,\zeta_s,x'_\rho}\|^\mathfrak{m}_{[-\tau,0]} \\
&\leq \|\overline{\xi}_{h,\zeta,\upsilon_h} - \overline{\xi}_{(N+1)h,\zeta_s,\overline{x}_\rho}\|^\mathfrak{m}_{[-\tau,0]} + \|\overline{\xi}_{(N+1)h,\zeta_s,\overline{x}_\rho} - \overline{\xi}_{Nh,\zeta_s,x'_\rho}\|^\mathfrak{m}_{[-\tau,0]} \\
&\leq \|\overline{\xi}_{h,\zeta,\upsilon_h} - \overline{\xi}_{h,\overline{\xi}_{Nh,\zeta_s,x_\rho},u_\rho}\|^\mathfrak{m}_{[-\tau,0]} + \|\overline{\xi}_{Nh,\overline{\xi}_{h,\zeta_s,u_1},x'_\rho} - \overline{\xi}_{Nh,\zeta_s,x'_\rho}\|^\mathfrak{m}_{[-\tau,0]} \\
&\leq \tilde{\beta}(\|\zeta - \overline{\xi}_{Nh,\zeta_s,x_\rho}\|^\mathfrak{m}_{[-\tau,0]}, h) + \tilde{\beta}(\|\overline{\xi}_{h,\zeta_s,\upsilon_1} - \zeta_s\|^\mathfrak{m}_{[-\tau,0]}, Nh) \leq \tilde{\beta}(\varepsilon, h) + \tilde{\beta}(\mathcal{Z}(\zeta_s), Nh) \leq \varepsilon.
\end{aligned}$$

Hence, $(\zeta', x'_\rho) \in \mathcal{R}_1$. Thus condition (ii) in Definition 3.2.3 holds. In a similar way, one can show that condition (iii) in Definition 3.2.3 holds which completes the proof. □

Note that in the above theorem, given any $\varepsilon > 0$, one can select temporal horizon N to be sufficiently large to enforce term $\tilde{\beta}(\mathcal{Z}(\zeta_s), Nh)$ to be sufficiently small. This results in $\tilde{\beta}(\varepsilon, h) < \varepsilon$ which enforces a lower bound for the sampling time h. Now we establish the results on the existence of finite dimensional abstraction $S_\rho(\overline{\Sigma}_R)$ such that $S_h(\overline{\Sigma}_R) \cong^{\varepsilon}_{\mathcal{S}} S_\rho(\overline{\Sigma}_R)$ given the result in Theorem 3.3.1.

Theorem 3.3.2. *Consider the results in Theorem 3.3.1. If we select*

$$X_{h0} \subseteq \{\zeta \in \mathcal{C}([-\tau, 0]; \mathbb{R}^n) \,|\, \|\overline{H}_h(\zeta) - H_\rho(x_{\rho 0})\|^{\mathfrak{m}}_{[-\tau,0]} \leq \varepsilon, \exists x_{\rho 0} \in X_{\rho 0}\},$$

then we have $S_h(\overline{\Sigma}_R) \cong^{\varepsilon}_{\mathcal{S}} S_\rho(\overline{\Sigma}_R)$.

Proof. For every $\zeta \in \overline{X}_{h0}$, there always exists $x_{\rho 0} \in X_{\rho 0}$ such that

$$\|\overline{H}_h(\zeta) - H_\rho(x_{\rho 0})\|^{\mathfrak{m}}_{[-\tau,0]} \leq \varepsilon.$$

Hence $(\zeta, x_{\rho 0}) \in \mathcal{R}_1$. In a similar way, we can show that for every $x_{\rho 0} \in X_{\rho 0}$ there exists $\zeta \in \overline{X}_{h0}$ such that $(\zeta, x_{\rho 0}) \in \mathcal{R}_1$, which completes the proof. □

3.4 Finite Abstractions

In this section, we provide a finite (a.k.a. symbolic) abstraction for $S_h(\overline{\Sigma}_R)$ by quantizing input set U. Let us consider tuple $\overline{\rho} = (h, N, \zeta_s, \eta)$, where $\eta > 0$ is a quantization parameter and the quantized input set is denoted by $[\mathsf{U}]_\eta$. Now, we can define the corresponding finite systems as

$$S_{\overline{\rho}}(\overline{\Sigma}_R) = (X_{\overline{\rho}}, X_{\overline{\rho}0}, U_{\overline{\rho}}, \underset{\overline{\rho}}{\longrightarrow}, Y_{\overline{\rho}}, H_{\overline{\rho}}),$$

where

- $X_{\overline{\rho}} = [\mathsf{U}]^N_\eta$, $X_{\overline{\rho}0} = X_{\overline{\rho}}$, $U_{\overline{\rho}} = [\mathsf{U}]_\eta$, $Y_{\overline{\rho}} = Y_h$;
- $x_{\overline{\rho}} \xrightarrow[\overline{\rho}]{\overline{u}_{\overline{\rho}}} x'_{\overline{\rho}}$, where $x_{\overline{\rho}} = (\overline{u}_1, \overline{u}_2, \ldots, \overline{u}_N) \in X_{\overline{\rho}}$, if and only if $x'_{\overline{\rho}} = (\overline{u}_2, \ldots, \overline{u}_N, \overline{u}_{\overline{\rho}})$;
- $H_{\overline{\rho}}(x_{\overline{\rho}}) = \overline{\xi}_{Nh, \zeta_s, x_{\overline{\rho}}}$.

A finite state-run of $S_{\overline{\rho}}(\overline{\Sigma}_R)$ is represented by $x_{\overline{\rho}}(0) \xrightarrow[\overline{\rho}]{\upsilon_0} x_{\overline{\rho}}(1) \xrightarrow[\overline{\rho}]{\upsilon_1} \cdots \xrightarrow[\overline{\rho}]{\upsilon_{k-1}} x_{\overline{\rho}}(k)$, where $\upsilon_i \in U_{\overline{\rho}}$. For later use, we denote by $\tilde{Y}_{y_0, \upsilon_{\overline{\rho}}} : \mathbb{N}_0 \to \mathbb{R}^n$ an $\mathbb{R}^n$-valued output-run of $S_{\overline{\rho}}(\overline{\Sigma}_R)$ starting from $y_0 = \overline{\xi}_{\zeta_s, x_{\overline{\rho}}(0)}(Nh)$ under input signal $\upsilon_{\overline{\rho}}$. In order to provide an approximate bisimulation relation between sampled retarded systems and symbolic models, we need the following technical lemma.

Lemma 3.4.1. *Consider a retarded system* $\overline{\Sigma}_R$ *corresponding to* δ*-ISS-*$M_{\mathfrak{m}}$ *RJDS* Σ_R *for* $\mathfrak{m} \geq 1$ *and a quantization parameter* η *such that* $0 < \eta \leq span(\mathsf{U})$. *Then the relation* $\mathcal{R}_2$ *given by*

$$\mathcal{R}_2 = \{(x_\rho, x_{\overline{\rho}}) \in X_\rho \times X_{\overline{\rho}} \mid x_\rho = (u_1, u_2, \ldots, u_N), x_{\overline{\rho}} = ([u_1]_\eta, [u_2]_\eta, \ldots, [u_N]_\eta)\}, \tag{3.4.1}$$

is a $\gamma(\eta)$*-approximate bisimulation relation between* $S_\rho(\overline{\Sigma}_R)$ *and* $S_{\overline{\rho}}(\overline{\Sigma}_R)$, *and* $S_\rho(\overline{\Sigma}_R) \cong^{\gamma(\eta)}_{\mathcal{S}} S_{\overline{\rho}}(\overline{\Sigma}_R)$.

Proof. Let $(x_\rho, x_{\overline{\rho}}) \in \mathcal{R}_2$, then $\|u_i - [u_i]_\eta\| \leq \eta$ for $i \in [1; N]$ implies that $\|x_\rho - x_{\overline{\rho}}\|_\infty \leq \eta$. By using (3.2.2), one obtains

$$\|H_\rho(x_\rho) - H_{\overline{\rho}}(x_{\overline{\rho}})\|^{\mathfrak{m}} = \|\overline{\xi}_{Nh,\zeta_s,x_\rho} - \overline{\xi}_{Nh,\zeta_s,x_{\overline{\rho}}}\|^{\mathfrak{m}}_{[-\tau,0]} \leq \gamma(\|x_\rho - x_{\overline{\rho}}\|_\infty) \leq \gamma(\eta).$$

Then, the first condition in Definition 3.2.3 holds. Now, consider any $(x_\rho, x_{\overline{\rho}}) \in \mathcal{R}_2$, where $x_\rho = (u_1, u_2, \ldots, u_N)$ and $x_{\overline{\rho}} = (\overline{u}_1, \overline{u}_2, \ldots, \overline{u}_N)$. Let $u \in U_\rho$ and consider $x_\rho \xrightarrow[\rho]{u} x'_\rho := (u_2, \ldots, u_N, u)$ in $S_\rho(\overline{\Sigma}_R)$. Choose $\overline{u} = [u]_\eta$ and consider $x_{\overline{\rho}} \xrightarrow[\overline{\rho}]{\overline{u}} x'_{\overline{\rho}} := (\overline{u}_2, \ldots, \overline{u}_N, \overline{u})$ in $S_{\overline{\rho}}(\overline{\Sigma}_R)$. It is obvious that $(x'_\rho, x'_{\overline{\rho}}) \in \mathcal{R}_2$ and, hence, condition (ii) in Definition 3.2.3 holds. Similarly, condition (iii) in Definition 3.2.3 holds which shows $\mathcal{R}_2$ is a $\gamma(\eta)$-approximate bisimulation relation between $S_\rho(\overline{\Sigma}_R)$ and $S_{\overline{\rho}}(\overline{\Sigma}_R)$. For any $x_{\rho 0} := (u_1, u_2, \ldots, u_N) \in X_{\rho 0}$, there always exists $x_{\overline{\rho}0} := ([u_1]_\eta, [u_2]_\eta, \ldots, [u_N]_\eta) \in X_{\overline{\rho}0}$ and, hence, $(x_{\rho 0}, x_{\overline{\rho}0}) \in \mathcal{R}_2$. Note that the existence of such $x_{\overline{\rho}0}$ is guaranteed by U being a finite union of boxes and by the inequality $\eta \leq span(\mathsf{U})$. Moreover, for any $x_{\overline{\rho}0} \in X_{\overline{\rho}0}$ and by choosing $x_{\rho 0} = x_{\overline{\rho}0}$, one readily gets $(x_{\rho 0}, x_{\overline{\rho}0}) \in \mathcal{R}_2$ and, hence, $S_\rho(\overline{\Sigma}_R) \cong_{\mathcal{S}}^{\gamma(\eta)} S_{\overline{\rho}}(\overline{\Sigma}_R)$. □

Now we provide the main results of this section on establishing an approximate bisimulation relation between $S_h(\overline{\Sigma}_R)$ and $S_{\overline{\rho}}(\overline{\Sigma}_R)$, which is an immediate consequence of the transitivity property of approximate bisimulation relations [GP07, Proposition 4] as recalled next.

Proposition 3.4.2. *Consider metric systems S_1, S_2, and S_3 such that $S_1 \cong_{\mathcal{S}}^{\delta_1} S_2$ and $S_2 \cong_{\mathcal{S}}^{\delta_2} S_3$, for some $\delta_1, \delta_2 \in \mathbb{R}_0^+$. Then, we have $S_1 \cong_{\mathcal{S}}^{\delta_1+\delta_2} S_3$.*

Now we provide the first main result of this section.

Theorem 3.4.3. *Consider a retarded system $\overline{\Sigma}_R$ corresponding to δ-ISS-$M_{\mathfrak{m}}$ RJDS Σ_R for $\mathfrak{m} \geq 1$. Given any $\varepsilon > 0$ and a quantization parameter $0 < \eta \leq span(\mathsf{U})$, consider the results in Theorem 3.3.1 and Lemma 3.4.1. Then, the relation $\mathcal{R}$ given by*

$$\mathcal{R} = \big\{(x_h, x_{\overline{\rho}}) \in X_h \times \overline{X}_{\overline{\rho}} \mid \exists x_\rho \in X_\rho, (x_h, x_\rho) \in \mathcal{R}_1 \text{ and } (x_\rho, x_{\overline{\rho}}) \in \mathcal{R}_2\big\},$$

is an $(\varepsilon + \gamma(\eta))$-approximate bisimulation relation between $S_h(\overline{\Sigma}_R)$ and $S_{\overline{\rho}}(\overline{\Sigma}_R)$.

Note that relations $\mathcal{R}_1$ and $\mathcal{R}_2$ in Theorem 3.4.3 have been defined in Theorem 3.3.1 and Lemma 3.4.1, respectively.

Having the result in Theorem 3.4.3, now we provide the main result of the chapter that quantifies the closeness of the output trajectories of sampled retarded jump-diffusion systems $S_h(\Sigma_R)$ and those of symbolic models $S_{\overline{\rho}}(\overline{\Sigma}_R)$. In order to prove this result, we raise a supplementary assumption on δ-ISS-$M_{\mathfrak{m}}$ Lyapunov functions V.

Assumption 3.4.4. *For a δ-ISS-$M_{\mathfrak{m}}$ Lyapunov function V as in Definition 2.3.3, the following conditions hold:*

(i) for function ω in Definition 2.3.3 there exists $\kappa \in \mathbb{R}^+$ such that $\omega(x, \hat{x}) \geq \kappa V(x, \hat{x})$ for all $x, \hat{x} \in \mathbb{R}^n$;

(ii) there exists some constant $\mathfrak{K} \in \mathbb{R}_0^+$ *such that*

$$\frac{1}{2}\mathsf{Tr}(\overline{g}(\zeta)\overline{g}^T(\hat{\zeta})\partial_{\zeta(0),\zeta(0)}V(\zeta(0),\hat{\zeta}(0)))-\frac{1}{2}\mathsf{Tr}\left(\begin{bmatrix}\overline{g}(\zeta)\\ \overline{g}(\hat{\zeta})\end{bmatrix}\begin{bmatrix}\overline{g}^T(\zeta) & \overline{g}^T(\hat{\zeta})\end{bmatrix}\mathbb{H}(V)(\zeta(0),\hat{\zeta}(0))\right)$$
$$+\sum_{i=1}^{\bar{r}}\lambda_i\Big(V\big(\zeta(0)+\overline{r}(\zeta)e_i,\hat{\zeta}(0)\big)-V\big(\zeta(0)+\overline{r}(\zeta)e_i,\hat{\zeta}(0)+\overline{r}(\hat{\zeta})e_i\big)\Big)\leq\mathfrak{K}, \qquad (3.4.2)$$

for all $\zeta,\hat{\zeta}\in\mathcal{C}([-\tau,0];\mathbb{R}^n)$*, where* $\mathbb{H}(V)(x,\hat{x})$ *denotes the Hessian matrix of* V *at* $x,\hat{x}\in\mathbb{R}^n$*.*

Note that condition (ii) in Assumption 3.4.4 is not restrictive, provided that V is restricted to a compact subset of $\mathbb{R}^n\times\mathbb{R}^n$, the Hessian matrix $\mathbb{H}(V)(x,\hat{x})$ of V is a positive semidefinite matrix in $\mathbb{R}^{2n\times 2n}$, and $\partial_{x,x}V(x,\hat{x})\leq\overline{P}$ for some positive semidefinite matrix $\overline{P}$ in $\mathbb{R}^{n\times n}$. With these conditions and Lipschitz assumptions on $\overline{g}$ and $\overline{r}$, one can always find $\mathfrak{K}\in\mathbb{R}_0^+$ satisfying (3.4.2).

Theorem 3.4.5. *Consider an RJDS* Σ_R *admitting a* δ*-ISS-*$M_{\mathfrak{m}}$ *Lyapunov function* V *for* $\mathfrak{m}\geq 1$ *as in Definition 2.3.3 and satisfying conditions in Assumption 3.4.4. For any abstraction parameters* $\overline{\rho}=(h,N,\zeta_s,\eta)$ *and any* $\varepsilon\in\mathbb{R}^+$ *satisfying* (3.3.1)*, consider finite abstraction* $S_{\overline{\rho}}(\overline{\Sigma}_R)$ *and results in Theorem 3.4.3. For any* $\mathbb{R}^n$*-valued output-run* ξ_{ζ,υ_h} *of* $S_h(\Sigma_R)$*, there exists an* $\mathbb{R}^n$*-valued output-run* $\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}$ *of* $S_{\overline{\rho}}(\overline{\Sigma}_R)$*, and vice versa, such that following inequalities hold*

$$\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\|\xi_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\geq\epsilon+\varepsilon+\gamma(\eta)\mid\zeta\right\}\leq 1-\mathsf{e}^{-\frac{\mathfrak{K}T_dh}{\underline{\alpha}(\epsilon)}},\quad \text{if } \underline{\alpha}(\epsilon)\geq\frac{\mathfrak{K}}{\kappa}, \qquad (3.4.3)$$

$$\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\|\xi_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\geq\epsilon+\varepsilon+\gamma(\eta)\mid\zeta\right\}\leq\frac{(\mathsf{e}^{\kappa T_dh}-1)\mathfrak{K}}{\kappa\underline{\alpha}(\epsilon)\mathsf{e}^{\kappa T_dh}},\quad \text{if } \underline{\alpha}(\epsilon)\leq\frac{\mathfrak{K}}{\kappa}. \qquad (3.4.4)$$

Proof. From the result in Theorem 3.4.3, we have $S_h(\overline{\Sigma}_R)\cong^{\varepsilon+\gamma(\eta)}_{\mathcal{S}} S_{\overline{\rho}}(\overline{\Sigma}_R)$ which implies $\|\overline{\xi}_{h,\zeta,\upsilon_h}-\overline{\xi}_{Nh,\zeta_s,x_{\overline{\rho}}}\|^{\mathfrak{m}}_{[-\tau,0]}\leq\varepsilon+\gamma(\eta)$ and consequently

$$\sup_{k\in\mathbb{N}_0}\|\overline{\xi}_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\leq\varepsilon+\gamma(\eta). \qquad (3.4.5)$$

Hence, one obtains the following chain of (in)equalities:

$$
\begin{aligned}
&\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\|\xi_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\geq\epsilon+\varepsilon+\gamma(\eta)\mid\zeta\right\}\\
&=\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\|\xi_{\zeta,\upsilon_h}(kh)-\overline{\xi}_{\zeta,\upsilon_h}(kh)+\overline{\xi}_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\geq\epsilon+\varepsilon+\gamma(\eta)\,|\,\zeta\right\}\\
&\leq\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\{\|\xi_{\zeta,\upsilon_h}(kh)-\overline{\xi}_{\zeta,\upsilon_h}(kh)\|^{\mathfrak{m}}+\|\overline{\xi}_{\zeta,\upsilon_h}(kh)-\tilde{Y}_{\overline{\xi}_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^{\mathfrak{m}}\}\geq\epsilon+\varepsilon+\gamma(\eta)\,|\,\zeta\right\}\\
&\leq\mathbb{P}\left\{\sup_{0\leq k\leq T_d}\underline{\alpha}(\|\xi_{\zeta,\upsilon_h}(kh)-\overline{\xi}_{\zeta,\upsilon_h}(kh)\|^{\mathfrak{m}})\geq\underline{\alpha}(\epsilon)\mid\zeta\right\}\\
&\leq\mathbb{P}\left\{\sup_{0\leq k\leq T_d}V(\xi_{\zeta,\upsilon_h}(kh),\overline{\xi}_{\zeta,\upsilon_h}(kh))\geq\underline{\alpha}(\epsilon)\mid\zeta\right\}. \qquad (3.4.6)
\end{aligned}
$$

From the properties of V in Definition 2.3.3 and Assumption 3.4.4, one has

$$
\begin{aligned}
\mathcal{D}V(\xi_{t,\zeta,\upsilon},\overline{\xi}_{t,\zeta,\upsilon}) =&\begin{bmatrix}\partial_x V & \partial_{\hat{x}}V\end{bmatrix}\begin{bmatrix}\overline{f}(\xi_{t,\zeta,\upsilon},\upsilon(t))\\ \overline{f}(\overline{\xi}_{t,\zeta,\upsilon},\upsilon(t))\end{bmatrix}+\frac{1}{2}\mathsf{Tr}(\overline{g}(\xi_{t,\zeta,\upsilon})\overline{g}^T(\xi_{t,\zeta,\upsilon})\partial_{x,x}V)\\
&+\sum_{i=1}^{\tilde{r}}\lambda_i\Big(V(\xi_{\zeta,\upsilon}(t)+\overline{r}(\xi_{t,\zeta,\upsilon})e_i,\overline{\xi}_{\zeta,\upsilon}(t))-V(\xi_{\zeta,\upsilon}(t),\overline{\xi}_{\zeta,\upsilon}(t))\Big)\\
=&\begin{bmatrix}\partial_x V & \partial_{\hat{x}}V\end{bmatrix}\begin{bmatrix}\overline{f}(\xi_{t,\zeta,\upsilon},\upsilon(t))\\ \overline{f}(\overline{\xi}_{t,\zeta,\upsilon},\upsilon(t))\end{bmatrix}+\frac{1}{2}\mathsf{Tr}\Big(\begin{bmatrix}\overline{g}(\xi_{t,\zeta,\upsilon})\\ \overline{g}(\overline{\xi}_{t,\zeta,\upsilon})\end{bmatrix}\begin{bmatrix}\overline{g}^T(\xi_{t,\zeta,\upsilon}) & \overline{g}^T(\overline{\xi}_{t,\zeta,\upsilon})\end{bmatrix}\mathbb{H}(V)\Big)\\
&+\sum_{i=1}^{\tilde{r}}\lambda_i\Big(V(\xi_{\zeta,\upsilon}(t)+\overline{r}(\xi_{t,\zeta,\upsilon})e_i,\overline{\xi}_{\zeta,\upsilon}(t)+\overline{r}(\overline{\xi}_{t,\zeta,\upsilon})e_i)-V(\xi_{\zeta,\upsilon}(t),\overline{\xi}_{\zeta,\upsilon}(t))\Big)\\
&+\frac{1}{2}\mathsf{Tr}(\overline{g}(\xi_{t,\zeta,\upsilon})\overline{g}^T(\xi_{t,\zeta,\upsilon})\partial_{x,x}V)+\sum_{i=1}^{\tilde{r}}\lambda_i V(\xi_{\zeta,\upsilon}(t)+\overline{r}(\xi_{t,\zeta,\upsilon})e_i,\overline{\xi}_{\zeta,\upsilon}(t))\\
&-\frac{1}{2}\mathsf{Tr}\Big(\begin{bmatrix}\overline{g}(\xi_{t,\zeta,\upsilon})\\ \overline{g}(\overline{\xi}_{t,\zeta,\upsilon})\end{bmatrix}\begin{bmatrix}\overline{g}^T(\xi_{t,\zeta,\upsilon}) & \overline{g}^T(\overline{\xi}_{t,\zeta,\upsilon})\end{bmatrix}\mathbb{H}(V)\Big)\\
&-\sum_{i=1}^{\tilde{r}}\lambda_i V(\xi_{\zeta,\upsilon}(t)+\overline{r}(\xi_{s,\zeta,\upsilon})e_i,\overline{\xi}_{\zeta,\upsilon}(t)+\overline{r}(\overline{\xi}_{t,\zeta,\upsilon})e_i)\\
&\leq-\kappa V(\xi_{\zeta,\upsilon}(t),\overline{\xi}_{\zeta,\upsilon}(t))+\mathfrak{K}, \qquad (3.4.7)
\end{aligned}
$$

for any $t\in\mathbb{R}_0^+$, $\upsilon_h\in\mathcal{U}_h$ and $\zeta\in\mathcal{C}^b_{\mathcal{F}_0}([-\tau,0];\mathbb{R}^n)$. Using inequalities (3.4.6), (3.4.7), and the result in [Kus67, Theorem 1, pp. 79], one obtains the relations (3.4.3) and (3.4.4). In a similar way, we can prove the other direction. □

Inequalities (3.4.3) and (3.4.4) lower bound the probability such that the distance between output trajectories of the finite abstraction and those of the corresponding

sampled retarded jump-diffusion system remains close over finite time horizon. One can leverage the result in Theorem 3.4.5 to synthesize control policies for finite abstractions and refine them to the original systems while providing guarantee on the probability of satisfaction. Similar relations were established in [JP09], [ZMEM+14], and [LSZ19] for stochastic hybrid systems, stochastic control systems, and interconnected stochastic control systems and their (in)finite abstractions. For a detailed discussion on how inequalities (3.4.3) and (3.4.4) can be used to provide a lower bound on the probability of satisfying the specification for the original systems, we kindly refer the interested readers to [LSZ19, Section 6].

3.5 Case Study

To show the effectiveness of the proposed results, we consider a simple thermal model of ten-room building as shown schematically in Figure 3.1. The model used here is similar to that used in [ZAG15]. In addition, we modified arrangement of the rooms to increase state-space dimensions and considered that the dynamic is affected by delays in states and by jumps (modeling door and window opening). The dynamic of the considered delayed jump-diffusion system Σ_D is given by the following delayed stochastic differential equations:

$$
\begin{aligned}
\mathrm{d}\,\xi_i(t) =& (\alpha_r(\xi_2(t)-\xi_i(t))+\alpha_e(T_e-\xi_i(t))-\alpha_{\tau_1}\xi_i(t-\tau_1))\,\mathrm{d}\,t+(\overline{g}\xi_i(t)+\overline{g}_{\tau_2}\xi_i(t-\tau_2))\,\mathrm{d}\,W_t^i \\
&+\overline{r}\xi_i(t)\,\mathrm{d}\,P_t^i, \quad i=\{1,7,9\}, \\
\mathrm{d}\,\xi_2(t) =& (\alpha_r(\xi_1(t)-4\xi_2(t)+\xi_7(t)+\xi_9(t)+\xi_3(t))+\alpha_{e_H}(T_e-\xi_2(t))-\alpha_{\tau_1}\xi_2(t-\tau_1)+\alpha_H(T_h-\xi_2(t))\upsilon_1)\,\mathrm{d}\,t \\
&+(\overline{g}\xi_2(t)+\overline{g}_{\tau_2}\xi_2(t-\tau_2))\,\mathrm{d}\,W_t^2+\overline{r}\xi_2(t)\,\mathrm{d}\,P_t^2, \\
\mathrm{d}\,\xi_j(t) =& (\alpha_r(\xi_{j-1}(t)-2\xi_j(t)+\xi_{j+1}(t))+\alpha_e(T_e-\xi_j(t))-\alpha_{\tau_1}\xi_j(t-\tau_1))\,\mathrm{d}\,t+(\overline{g}\xi_j(t)+\overline{g}_{\tau_2}\xi_j(t-\tau_2))\,\mathrm{d}\,W_t^j \\
&+\overline{r}\xi_j(t)\,\mathrm{d}\,P_t^j, \quad j=\{3,4\}, \\
\mathrm{d}\,\xi_5(t) =& (\alpha_r(\xi_4(t)-4\xi_5(t)+\xi_8(t)+\xi_{10}(t)+\xi_6(t))+\alpha_{e_H}(T_e-\xi_5(t))-\alpha_{\tau_1}\xi_5(t-\tau_1)+\alpha_H(T_h-\xi_5(t))\upsilon_2)\,\mathrm{d}\,t \\
&+(\overline{g}\xi_5(t)+\overline{g}_{\tau_2}\xi_5(t-\tau_2))\,\mathrm{d}\,W_t^5+\overline{r}\xi_5(t)\,\mathrm{d}\,P_t^5, \\
\mathrm{d}\,\xi_l(t) =& (\alpha_r(\xi_5(t)-\xi_l(t))+\alpha_e(T_e-\xi_l(t))-\alpha_{\tau_1}\xi_l(t-\tau_1))\,\mathrm{d}\,t+(\overline{g}\xi_l(t)+\overline{g}_{\tau_2}\xi_l(t-\tau_2))\,\mathrm{d}\,W_t^l+\overline{r}\xi_l(t)\,\mathrm{d}\,P_t^l, \\
& l=\{6,8,10\},
\end{aligned}
$$

where the terms W_t^i and P_t^i, $i \in [1;10]$, denote the standard Brownian motion and

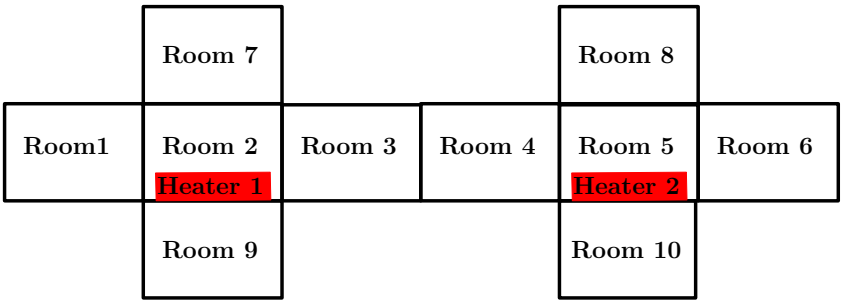

Figure 3.1: A schematic of ten-room building.

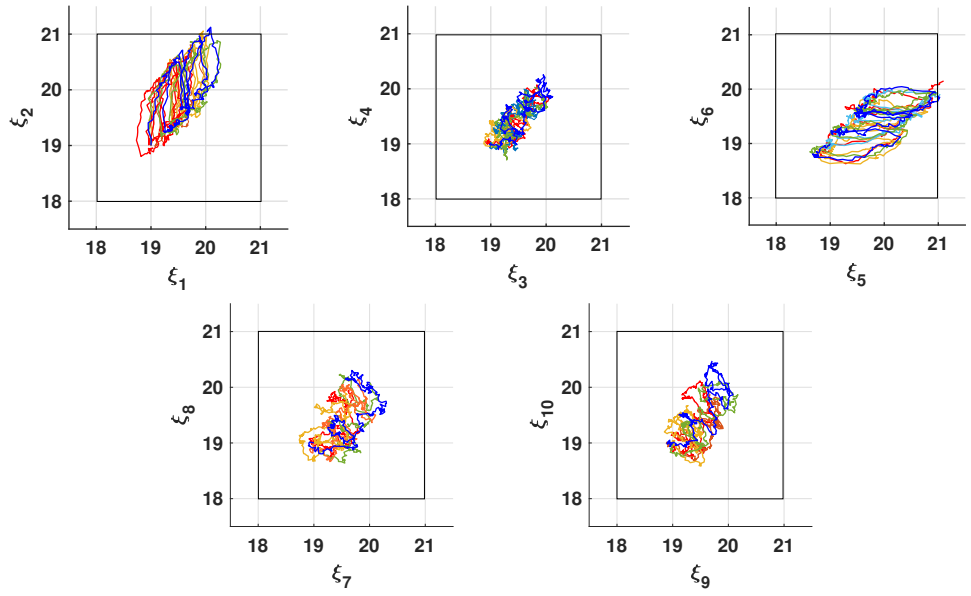

Figure 3.2: A few realizations of the solution process $\xi_{\zeta,\upsilon}$ with initial condition $\zeta \equiv [19, 19, 19, 19, 19, 19, 19, 19, 19, 19]^T$.

Poisson process with rate $\lambda_i = 0.1$, respectively; ξ_i, $i \in [1; 10]$, denote the temperature in each room; $T_e = 15$ (degree Celsius) is the external temperature; $T_{H_1} = T_{H_2} = 100$ are the temperatures of two heaters; $\tau_1 = 10$ time units and $\tau_2 = 5$ time units are state delays in drift and diffusion terms, respectively; and the control inputs υ_1, υ_2 are amounted to 1 if corresponding heaters are on and to 0 if corresponding heaters are off. Here, we assume that at most one heater is on at each time instance which results in the finite input set $\mathsf{U} = \{(1, 0), (0, 1), (0, 0)\}$. The system parameters are chosen as $\alpha_r = 5 \times 10^{-2}$, $\alpha_{eH} = 8 \times 10^{-3}$, $\alpha_e = 5 \times 10^{-3}$, $\alpha_H = 3.6 \times 10^{-3}$, $\alpha_{\tau_1} = 1 \times 10^{-4}$, $\overline{g} = 2 \times 10^{-3}$, $\overline{g}_{\tau_2} = 1 \times 10^{-4}$, and $\overline{r} = 1 \times 10^{-5}$. In this example, we work on the subset $\mathsf{D} = [17, 22]^{10}$ of state space of Σ_D. Using Lyapunov function $V(x, \hat{x}) = (x-\hat{x})^T P(x-\hat{x})$, for all $x, \hat{x} \in \mathsf{D}$, where

$$P = \begin{bmatrix}
1.1126 & -0.1205 & 0.0031 & 0.0004 & -0.0001 & 0.0000 & 0.0036 & 0.0000 & 0.0036 & 0.0000 \\
-0.1205 & 1.5002 & -0.1232 & 0.0026 & 0.0006 & -0.0001 & -0.1205 & -0.0001 & -0.1205 & -0.0001 \\
0.0031 & -0.1232 & 1.2308 & -0.1182 & 0.0026 & 0.0004 & 0.0031 & 0.0004 & 0.0031 & 0.0004 \\
0.0004 & 0.0026 & -0.1182 & 1.2308 & -0.1232 & 0.0031 & 0.0004 & 0.0031 & 0.0004 & 0.0031 \\
-0.0001 & 0.0006 & 0.0026 & -0.1232 & 1.5002 & -0.1205 & -0.0001 & -0.1205 & -0.0001 & -0.1205 \\
0.0000 & -0.0001 & 0.0004 & 0.0031 & -0.1205 & 1.1126 & 0.0000 & 0.0036 & 0.0000 & 0.0036 \\
0.0036 & -0.1205 & 0.0031 & 0.0004 & -0.0001 & 0.0000 & 1.1126 & 0.0000 & 0.0036 & 0.0000 \\
0.0000 & -0.0001 & 0.0004 & 0.0031 & -0.1205 & 0.0036 & 0.0000 & 1.1126 & 0.0000 & 0.0036 \\
0.0036 & -0.1205 & 0.0031 & 0.0004 & -0.0001 & 0.0000 & 0.0036 & 0.0000 & 1.1126 & 0.0000 \\
0.0000 & -0.0001 & 0.0004 & 0.0031 & -0.1205 & 0.0036 & 0.0000 & 0.0036 & 0.0000 & 1.1126
\end{bmatrix},$$

one can readily obtain the functions $\underline{\alpha}(s) = 0.5029s$, $\overline{\alpha}(s) = 0.8197s$, for $\mathfrak{m} = 2$ satisfying conditions (i) and (ii) in Definition 2.3.3. Using the results in Lemma 2.3.7, one obtains function $\beta(s, t) = \mathsf{e}^{-\kappa t}s$, $\forall s \in \mathbb{R}_0^+$, with $\kappa = 0.6667$.

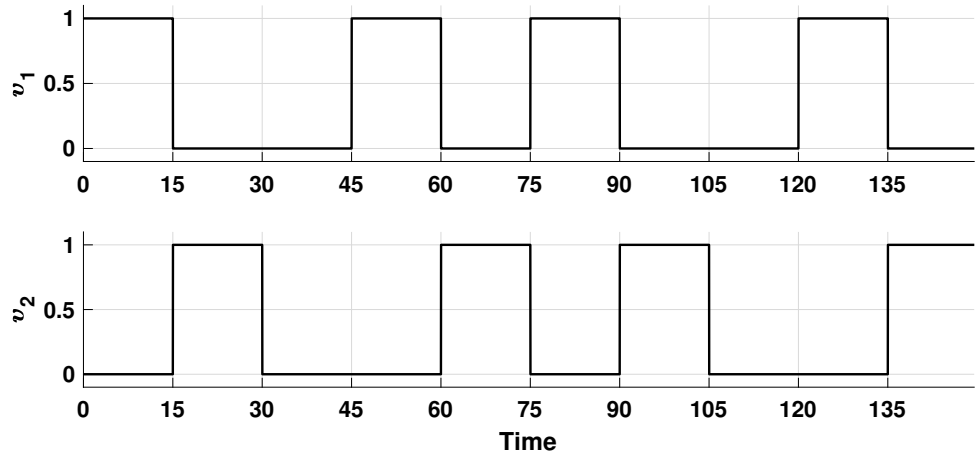

Figure 3.3: The evolution of input signals υ_1 and υ_2.

By considering $\mathfrak{m} = 2$, a constant function $\zeta_s \equiv [17, 17, 17, 17, 17, 17, 17, 17, 17, 17]^T \in \mathcal{C}([-\tau, 0]; \mathbb{R}^n)$, where $\tau = 10$, a precision $\varepsilon = 0.05$ and by fixing sampling time $h = 15$ time units, one can obtain temporal horizon $N = 9$ for $S_\rho(\overline{\Sigma}_D)$, satisfying inequality (2.2.21) in Theorem 3.3.1. Therefore, the resulting cardinality of the set of states of $S_\rho(\overline{\Sigma}_D)$ is $3^9 = 19683$ and number of transitions are $3^{10} = 59049$. The CPU time taken for computing finite abstraction with $N = 9$ is accounted to 0.015. From inequalities (3.4.3) and (3.4.4), one can observe that higher precision of the abstraction helps to improve the bound on the closeness of the trajectories. To get higher precision, one can increasing the value of N. For example, for $N = 11$ one gets an abstraction with precision $\varepsilon = 0.008$. However, it increases the size of the abstraction and the computation time which are 531441 and 0.413 seconds, respectively. Remark that since the input set is finite, the finite dimensional abstraction $S_\rho(\overline{\Sigma}_D)$ is also symbolic.

Now consider the objective to design a controller enforcing the trajectories of Σ_D to stay within comfort zone $\mathsf{W} = [18, 21]^{10}$. This corresponds to the LTL specification $\Box\mathsf{W}$. The computation of the symbolic model $S_\rho(\overline{\Sigma}_D)$ and the controller synthesis have been performed using tool QUEST [JZst] (discussed in Chapter 4) on a computer with CPU 3.5GHz Intel Core i7. The CPU time taken to synthesize controller for $N = 9$ and $N = 11$ is accounted to 2.07 and 2.56 seconds, respectively.

For the given dynamics and δ-ISS-M_2 Lyapunov function V on D, we have $\mathfrak{K} = 0.0012$ in Assumption 3.4.4. For $\epsilon = 0.8$, using the result in Theorem 3.4.5, we guarantee that the the distance between output of $S_h(\Sigma_D)$ (i.e. sampled Σ_D) and that of $S_\rho(\overline{\Sigma}_D)$ will not exceed $\epsilon + \varepsilon = 0.85$ over the discrete time horizon $\{0, 15, \ldots, 150\}$ with probability at least 75.8%. To get a better lower bound for the aforementioned probability, one can reduce the time horizon or increase ϵ. For example, if we consider discrete time horizon $\{0, 15, \ldots, 75\}$ and $\epsilon = 1$, the lower bound on the probability will be 89.5%.

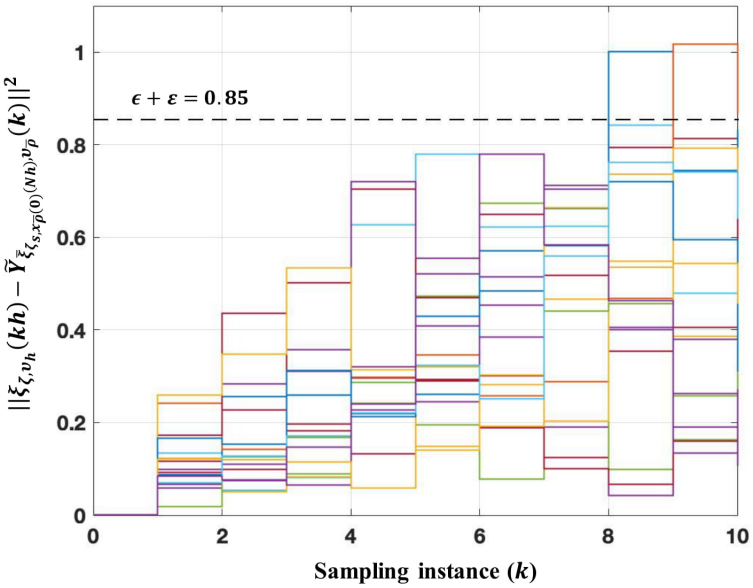

Figure 3.4: A few realizations of $\|\xi_{\zeta,\upsilon_h}(kh) - \tilde{Y}_{\xi_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^2$ for $T_d = 10$ at sampling instances.

Figure 3.2 shows a few realizations of the closed-loop solution process $\xi_{\zeta,\upsilon}$ starting form initial condition $\zeta \equiv [19, 19, 19, 19, 19, 19, 19, 19, 19, 19]^T \in X_{h0}$. The synthesized control policies υ_1 and υ_2 are shown in Figure 3.3. The obtained probability (i.e., at least 75.8%) is also empirically verified by computing distance between output trajectories of $S_h(\Sigma_R)$ and of $S_\rho(\overline{\Sigma}_R)$ (i.e., $\|\xi_{\zeta,\upsilon_h}(kh) - \tilde{Y}_{\xi_{\zeta_s,x_{\overline{\rho}}(0)}(Nh),\upsilon_{\overline{\rho}}}(k)\|^2$) using 5000 runs. Several realizations are shown in Figure 3.4.

4 QUEST: A Tool for State-Space Discretization-free Synthesis of Symbolic Controllers

In this chapter, we develop a software tool, called QUEST, for automated controller synthesis of incrementally input-to-state stable nonlinear control systems. This tool accepts ordinary differential equations as the descriptions of the nonlinear control systems and constructs their symbolic models using a state-space discretization-free approach which can potentially alleviate the issue regarding the so-called *curse of dimensionality* while computing discrete abstractions of the systems with high-dimensional state-space. The tool supports the computation of both minimal and maximal fixed points and thus provides natively algorithms to synthesize controllers enforcing safety and reachability specifications. The tool is designed in C++. The tool is open-source and available for download together with the user manual and some examples at `www.hcs.ei.tum.de/software`.

4.1 Introduction

Controller synthesis techniques using so-called discrete abstractions provide tools for automated, correct-by-construction controller synthesis for various systems to enforce complex specifications (usually given in linear temporal logic (LTL) formulae). There

Table 4.1: Existing Tools For Controller Synthesis

Tools	**Class of control systems**	**Implementation platform**
LTLMoP [FJKG10]	Integrator dynamics	Python
TuLiP [WTO+11]	Linear control systems	Python
Pessoa [MJDT10]	Nonlinear control systems	MATLAB
CoSyMa [MGG13]	Incrementally stable switched systems	OCaml
SCOTS [RZ16b]	Nonlinear control systems	C++
SENSE [KRZ18]	Nonlinear networked control systems	C++

have been recently various software tools on the symbolic controller synthesis for various classes of control systems. Some of them are listed in Table 4.1. However, the discrete abstractions obtained in these results and corresponding tools are based on state-space discretization. The state-space discretization schemes result in an exponential increase in computational complexity with the dimension of state space in the concrete system, and, hence, these techniques suffer severely from the issue of the so-called curse of dimensionality especially for the systems with high-dimensional state-space.

In [LCGG13, Gir14, ZAG15, ZTA16, JZ19], it has been shown that one can construct discrete abstractions without state-space discretization which are approximately bisimilar to incrementally input-to-state stable nonlinear control systems. The technique uses a fixed length of quantized input sequence as a symbolic state of the abstraction which helps to alleviate the curse of dimensionality. The length of input sequences, referred to as temporal horizon N, is used as a parameter to adjust the abstraction precision; a larger value of N results in a higher precision of the abstraction, and, consequently, in a larger abstraction in terms of the number of states.

In this chapter, we introduce QUEST, an open-source software tool implementing the synthesis of symbolic controllers based on the state-space discretization-free approach proposed in [LCGG13, Gir14, ZAG15, ZTA16, JZ19]. QUEST provides algorithms for the construction of discrete abstractions which are approximately bisimilar to the original incrementally input-to-state stable dynamics without the need to discretize the state space. Moreover, it provides algorithms for synthesizing controllers enforcing some classes of LTL specifications over concrete systems using fixed-point computations.

4.2 Tool Details

QUEST is implemented in C++ and employs binary decision diagrams (BDDs) [Bry92] as an underlying data structure to store and manipulate boolean functions representing symbolic abstractions and controllers. Operations on BDDs are handled with the help of CUDD binary decision diagram library [Som04]. QUEST provides two fixed point algorithms for maximal and minimal fixed point computation as described in [Tab09] and thus, natively, provides algorithms to synthesize controllers for safety and reachability specifications. Moreover, one can use combinations of these fixed point algorithms for synthesizing controllers enforcing more complex specifications such as reach and stay.

Inputs: QUEST accepts the description of the dynamics of incrementally input-to-state stable nonlinear control systems in the form of an ordinary differential equation; see [Ang02] for characterization of incremental stability in terms of Lyapunov functions. Additionally, the user needs to provide an input set, an input set discretization parameter η, a source state x_s, a sampling time τ, and a temporal horizon N; see [ZTA16] for the role of those parameters. The computation of parameter N for a given desired abstraction precision ε is provided in [ZTA16].

Output: QUEST synthesizes controllers with the help of fixed-point computations and stores them in the form of BDD. QUEST also provides an option to simulate the closed-loop system equipped with the synthesized controller. In particular, the QUEST provides

maximally permissible controller which is nondeterministic (i.e., it is represented as list of states along with all feasible control inputs that enforce required specification). One can utilize existing tool dtControl [AJJ+20] to efficiently obtain a deterministic controller using decision tree algorithms.

4.3 Installation and System Requirements

In general, QUEST is implemented in the "header-only" style and you only need a working C++ developer environment. However, QUESTuses the CUDD library maintained by Fabio Somenzi, which can be downloaded at `http://vlsi.colorado.edu/~fabio/`.

The requirements and installation instructions are summarized as follows:

1. A working C/C++ development environment
   - Mac OS X: You should install Xcode.app including the command line tools
   - Linux: Most Linux OS includes the necessary tools already
   - Windows: You need to have MSYS-2 installed or use the latest update of Windows 10 providing support for Ubunto-on-windows.
2. A working installation of the CUDD library and enabling the following options
   - the C++ object-oriented wrapper,
   - the dddmp library, and
   - the shared library.

   The package follows the usual configure, make, and make install installation routine. We use cudd-3.0.0, with the following configuration

   ```
   $ ./configure --enable-shared --enable-obj --enable-dddmp
   --prefix=/opt/local/
   ```

   On Windows and linux, we experienced that the header files `util.h` and `config.h` were missing in `/opt/local` and we manually copied them to `/opt/local/include`.

   For further details about windows installations (which is somehow different), please refer to the `installation_notes_windows.txt` file within QUEST. You should also test the BDD installation by compiling a dummy program, e.g. test.cc as the following

   ```
   #include<iostream>
   #include "cuddObj.hh"
   #include "dddmp.h"
   int main () {
   Cudd mgr(0,0);
   BDD x = mgr.bddVar();
   }
   ```

   which should be compiled using

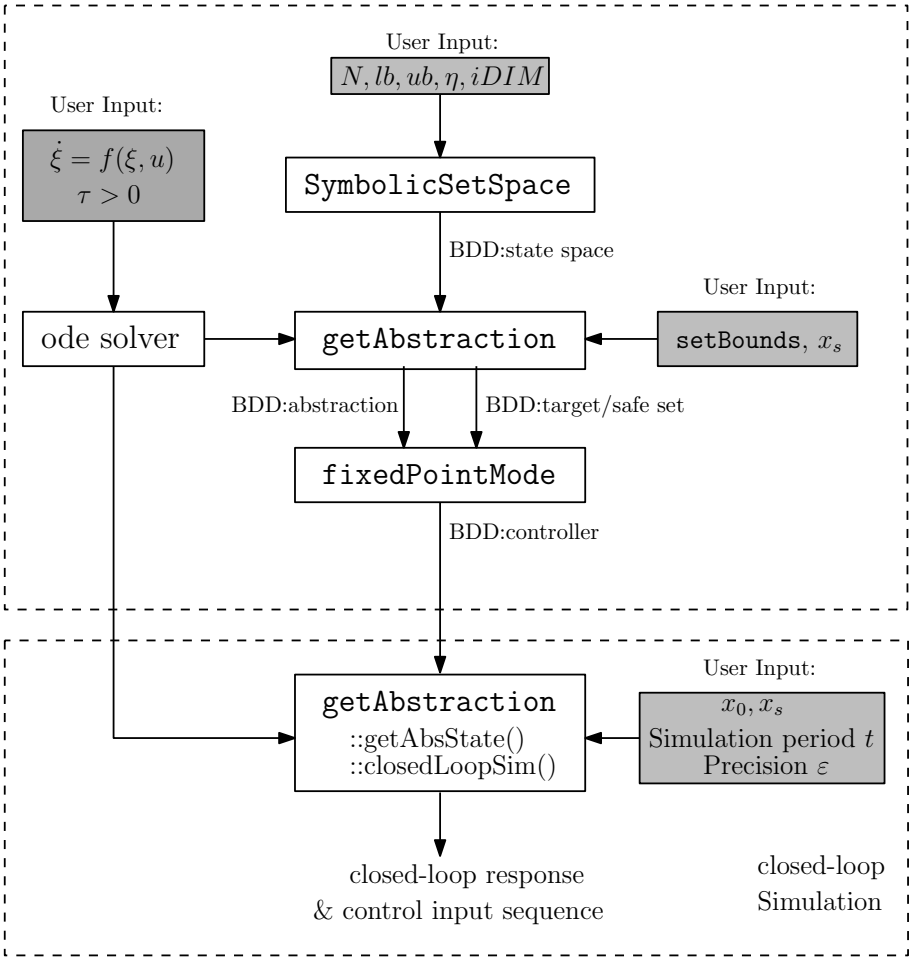

Figure 4.1: Workflow.

```
$ g++ test.cc -I/opt/local/include -L/opt/local/lib -lcudd
```

4.4 Implementation of QUEST

In this section, we describe the architecture of QUEST. The algorithm is mainly distributed among three C++ classes:

- `SymbolicSetSpace`
- `getAbstraction`
- `fixedPointMode`

4.4.1 `SymbolicSetSpace`

The `SymbolicSetSpace` is the main class in which the transition relations as described in [LCGG13, Gir14, ZAG15, ZTA16, JZ19] are computed with the help of binary de-

cision diagrams (BDDs) [Bry92] as underlying data structure. Specifically, we use the object oriented wrapper in the CUDD library [Som04]. It accepts temporal horizon N, dimension of input space $iDIM$, lower bound on input set lb, upper bound on input set ub, and discretization parameter η. The class `SymbolicSetSpace` directly constructs the transition relations as

Algorithm 1 Computation of transition relation

Require: N, lb, ub, η, $iDIM$

1: **for** $i = 1$ to $iDIM$ **do**
2: number of elements in quantized Input[i]= $(ub[i] - lb[i])/\eta[i]$
3: Number of states in abstraction= $(\Pi_{i=1}^{iDIM}$ number of elements in quantized Input[i]$)^N$
4: Let $x = (u_1, u_2, \ldots, u_N)$ be a state in abstraction, $x' = (u'_1, u'_2, \ldots, u'_N)$, and u be an input
5: **for all** x and u **do**
6: **for** $i = 1$ to $N - 1$ **do**
7: $u'_i = u_{i+1}$
8: $u'_N = u$

4.4.2 `getAbstraction`

The `getAbstraction` is a derived class of the `abstractionMode` which manages all BDD related information, such as number and indices of variables. The `getAbstraction` class provides some supporting functions that are required for overall operation of QUEST. Some of important functions are listed below:

```
getAbstraction::getAbstractSet()
/* get set of abstract states whose outputs are in safety/target region */
getAbstraction::getOutput()
/* get output H(x) corresponding to state x in the abstraction*/
getAbstraction::getAbsState()
/* get abstract state related to concrete state in the original system*/
getAbstraction::closedLoopSim()
/* closed-loop simulation and printing output response */
```

4.4.3 `fixedPointMode`

This class implements fixed point computation for the synthesis of controller. In particular, we use the methods `fixedPointMode::reach()`, `fixedPointMode::safe()` and `fixedPointMode::reachStay()` to synthesize controllers by solving fixed point computation for reachability, safety, and reach and stay specification, respectively.

The general work flow explaining use of classes with the different user inputs and the possible tool output is illustrated in Figure 4.1.

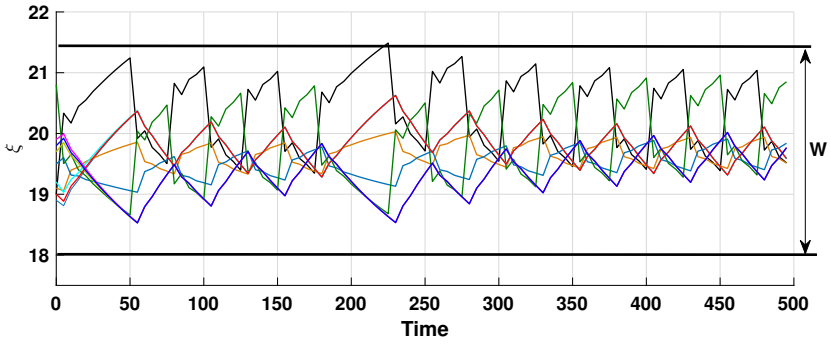

Figure 4.2: Evolution of temperatures in all rooms under synthesized controller.

4.5 Case Study

To demonstrate QUEST, we synthesize a controller regulating temperatures in a ten-room building shown schematically in Figure 3.1. QUEST accepts the dynamic given as an ordinary differential equation as shown below.

```
const int sDIM = 10;    /* System dimension */
const double T = 25;    /* Sampling time */
size_t N = 12;     /* Temporal Horizon */
typedef std::array<double,sDIM> state_type;
auto system_post = [](state_type &x, double* u) -> void {
auto rhs=[u](state_type &dx, const state_type &x) -> void {
const double a=0.05, ae2=0.005, ae5=0.005, ae=0.0033, ah=0.0036;
const double te=12;  /* External temperature */
const double th=100;  /* Heater temperature */
dx[0]=(-a-ae)*x[0]+a*x[1]+ae*te;
dx[1]=(-4*a-ae2-ah*u[0])*x[1]+a*x[0]+a*x[6]+a*x[8]+a*x[2]
          +ae2*te+ah*th*u[0];
dx[2]=(-2*a-ae)*x[2]+a*x[1]+a*x[3]+ae*te;
dx[3]=(-2*a-ae)*x[3]+a*x[2]+a*x[4]+ae*te;
dx[4]=(-4*a-ae5-ah*u[1])*x[4]+a*x[3]+a*x[7]+a*x[5]+a*x[9]
          +ae5*te+ah*th*u[1];
dx[5]=(-a-ae)*x[5]+a*x[4]+ae*te; dx[6]=(-a-ae)*x[6]+a*x[1]+ae*te;
dx[7]=(-a-ae)*x[7]+a*x[4]+ae*te; dx[8]=(-a-ae)*x[8]+a*x[1]+ae*te;
dx[9]=(-a-ae)*x[9]+a*x[4]+ae*te; };
size_t nint = 5;    /* no. of time step for ode solving */
ode_solver(rhs,x,nint,h);  /* Runga Kutte solver */ }
```

In this example, we consider that the control inputs u[0] and u[1] corresponding to heaters H_1 and H_2 are equal to 1 if the corresponding heaters are on and equal to 0 if

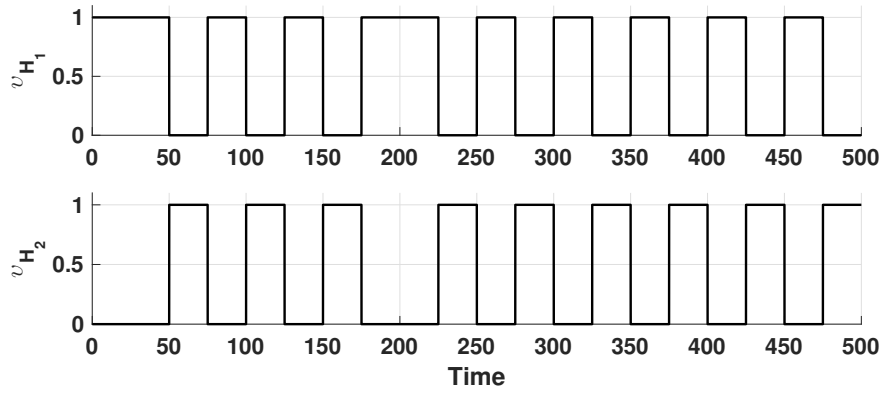

Figure 4.3: Input trajectories given by synthesized controller.

Table 4.2: Performance comparison for different values of N

N	13	12	11	10	9
Number of transitions in the abstraction	4782969	1594323	531441	177147	59049
Number of transitions in the controller	173980	55808	17888	5582	1722
Abstraction computation time (sec)	5.87	1.6	0.4	0.072	0.014
Controller computation time (sec)	319.63	96.89	29.56	9.12	2.67
Precision ε	0.05	0.1	0.22	0.5	1.4

the corresponding heaters are off. Here, we assume that at most one heater is on at each time instance. Thus, the input set of the system is given as

```
const int iDIM = 2;  /* Input dimension */
const size_t P = 3; /* Number of elements in the input set*/
double ud[P][iDIM]={{0,0},{0,1},{1,0}};
```

For this example, we consider the objective to synthesize a controller enforcing all the temperatures to stay within $\mathsf{W} = [18, 21.5]^{10}$. This corresponds to the LTL specification $\Box\mathsf{W}$ (i.e. safety specification) and is given to the tool as

```
auto setBounds = [](state_type y) -> bool {
double ul=21.8, ll=18;
/*upper and lower bound on the temperature in each room*/
bool s = true;
for(int j = 0; j < sDIM; j++){
if( y[j] >= ul || y[j] <= ll ) {s = false; break;}}
return s;}
```

We use temporal horizon $N = 12$, sampling time $\tau = 25$ time units, and source state $x_s = [17, 17, 17, 17, 17, 17, 17, 17, 17, 17]^T$ which result in precision $\varepsilon = 0.1$ for the discrete abstraction [ZTA16]. The computation of discrete abstraction and controller synthesis have been performed using QUEST on a windows computer with CPU 3.5GHz Intel Core i7. Figure 4.2 shows the evolution of the temperatures ξ in all rooms starting from initial condition $x_0 = [18.9, 19, 19.1, 19.5, 20.8, 19.7,\ 19.2, 19.9, 19, 19.8]^T$. Figure 4.3 illustrates the corresponding synthesized input trajectories υ_{H_1} and υ_{H_2}. Note that, the figures are generated using MATLAB by simulating system dynamics with the control inputs generated by QUEST. In Table 4.2, we show the effect of N on the size of the abstraction (given by the number of transitions), computation times, and precision ε. Remark that, due to the large dimension of the state-space, the existing tools such as Pessoa [MJDT10], CoSyMa [MGG13], and SCOTS [RZ16b] fail to synthesize any controller.

Part II

Controller Synthesis using Control Barrier Functions

5 Controller Synthesis for Stochastic Control Systems

This chapter provides another approach for synthesizing controller for stochastic control systems enforcing complex specifications. More specifically, we use the notion of so-called control barrier functions to synthesize controllers without any discretization.

5.1 Introduction

The controller synthesis problem for stochastic control systems against complex specifications (expressed using temporal logic formulae or automata on (in)finite strings) is very challenging. In general, the problem does not admit closed-form solutions and is hard to be solved exactly on such systems. The existing literature providing approximate or probabilistic solutions to this problem are listed below.

5.1.1 Related Literature

Synthesis using approximate finite abstractions

There have been several results in the literature utilizing approximate finite models as abstractions of the original *stochastic* dynamical systems for the formal policy synthesis. Existing results include policy synthesis for discrete-time stochastic hybrid systems [Sou14, APLS08], control of switched discrete-time stochastic systems [LAB15], and symbolic control of incrementally stable stochastic systems [ZMEM$^+$14]. These approaches rely on the discretization of the state set together with a formal upper-bound on the approximation error. These approaches suffer severely from the curse of dimensionality (i.e., computational complexity grows exponentially with the dimension of the state set). To alleviate this issue, sequential griding [SA13], state-space discretization-free abstractions [ZTA16, JZ19] (as discussed in Chapter 3), and compositional abstraction-based techniques [SAM15, LSZ18] are proposed under suitable assumptions on the system dynamics (e.g., Lipschitz continuity or incremental input-to-state stability).

Synthesis using control barrier functions

For *non-stochastic* systems, discretization-free approaches based on barrier functions were proposed for verification and synthesis to ensure safety [AXGT17, Jan18, NA18,

Pra06, WA07]. The authors in [WTL15] generalize the idea of the barrier function by combining it with the automata representation of LTL specifications for the verification of temporal property for nonlinear non-stochastic systems. The work is then extended for the verification of hybrid dynamical systems against syntactically co-safe LTL specifications [BD18] and for the synthesis of an online control strategy for multi-agent systems enforcing LTL specifications [SCE18]. There are a few recent results using barrier functions on non-stochastic systems to satisfy more general specifications. Results include the use of time-varying control barrier functions to satisfy signal temporal logic [LD19b] and control barrier function to design policies for reach and stay specification for non-stochastic switched systems [RS17]. Most of the synthesis results mentioned above consider prior knowledge of control barrier functions to provide online control strategies using quadratic programming. These results may not be suitable while dealing with constrained input sets which is the case in almost all real world applications.

For *stochastic* systems, there are very few works available in the literature to synthesize controllers against complex specifications using discretization-free approaches. The results include the synthesis of controller for continuous-time stochastic systems enforcing syntactically co-safe LTL specifications [HWM14], where the authors use automata representation corresponding to the specifications to guide a sequence of stochastic optimal control problems. The paper [FMPS18] considers synthesis for ensuring a lower bound on the probability of satisfying a specification in signal temporal logic. It encodes the requirements as chance constraints and inductively decomposes them into deterministic inequalities using the structure of the specification. Barrier functions are utilized in [HCL$^+$17, ST12, PJP07] for verification of stochastic (hybrid) systems but only with respect to the invariance property.

Our recent results in [JSZ18] present the idea of combining automata representation of a specification and control barrier function for formal verification of stochastic systems. This chapter is an extension of this work to solve the problem of controller synthesis for stochastic systems against complex temporal logic specifications.

5.1.2 Contributions

We consider temporal properties expressed in a fragment of LTL formulae, namely, LTL on finite traces, referred to as LTL$_F$ [SRK$^+$14]. We provide a systematic approach to synthesize a controller together with a lower bound on the probability that the LTL$_F$ property is satisfied over finite-time horizon. This is achieved by first decomposing specification into a sequence of simpler synthesis tasks based on the structure of the automaton associated with the negation of the specification. Then, controllers and corresponding probability bounds are obtained for these simplified synthesis tasks with the help of control barrier functions. In the final step, we combine these controllers and probability bounds to provide a hybrid control policy and a lower bound on the probability of satisfying the LTL$_F$ property.

In general, there is no guarantee that control barrier functions exist for a given stochastic system. Even if we know one exists, there is no complete algorithm for its computation. In this chapter, we provide two systematic approaches to search for control barrier

functions under suitable assumptions on the dynamics of the system and the shape of the potential control barrier functions. The first approach utilizes sum-of-squares optimization technique [Par03] and is suitable for dynamics with continuous input sets and polynomial dynamics. The second approach uses the counter-example guided inductive synthesis (CEGIS) scheme which is adapted from [RS15, RS17] and is suitable for systems with finite input sets.

5.2 Discrete-time Stochastic Control Systems

In this section, we consider discrete-time stochastic control systems (dt-SCS) that are extensively employed as models of systems under uncertainty in economics and finance [EA87] and in many engineering systems [BS96]. Examples of using dt-SCS include modelling inventory-production systems [HLL96], demand response in energy networks [Sou14], and analyzing max-plus linear systems in transportation [SAA16].

A (dt-SCS) is given by the tuple $\Sigma_s^{dt} = (X, V_{\mathsf{w}}, U, \omega, f_s)$, where X is the state set, V_{w} is the uncertainty set, and U is the input set of the system. We denote by $(X, \mathbb{B}(X))$ the measurable space with $\mathbb{B}(X)$ being the Borel sigma-algebra on the state space. Notation ω denotes a sequence of independent and identically distributed (i.i.d.) random variables on the set V_{w} as $\omega := \{\omega(k) : \Omega \to V_{\mathsf{w}},\ k \in \mathbb{N}_0\}$. The map $f_s : X \times U \times V_{\mathsf{w}} \to X$ is a measurable function characterizing the state evolution of the system. For a given initial state $\mathbf{x}(0) \in X$, the state evolution can be written as

$$\mathbf{x}(k+1) = f_s(\mathbf{x}(k), \upsilon(k), \omega(k)), \quad k \in \mathbb{N}_0. \tag{5.2.1}$$

We are interested in synthesizing a control policy υ that guarantees a potentially tight lower bound on the probability that the system Σ_s^{dt} satisfies a specification expressed as a temporal logic property. The syntax and semantics of the class of specifications dealt with in this chapter are provided in the next subsection. In this chapter, we consider *history-dependent policies* given by $\upsilon = (\upsilon_0, \upsilon_1, \ldots, \upsilon_n, \ldots)$ with functions $\upsilon_n : H_n \to U$, where H_n is a set of all n-histories h_n defined as $h_n := (\mathbf{x}(0), \upsilon(0), \mathbf{x}(1), \upsilon(1), \ldots, \mathbf{x}(n-1), \upsilon(n-1), \mathbf{x}(n))$. A subclass of policies are called *stationary* and are defined as $\mathsf{u} = (\mathsf{u}, \mathsf{u}, \ldots, \mathsf{u}, \ldots)$ with a function $\mathsf{u} : X \to U$. In stationary policies, the mapping at time n depends only on the current state x_n and does not change over time.

5.3 Preliminaries

5.3.1 Linear Temporal Logic over Finite Traces

In this subsection, we introduce linear temporal logic over finite traces, referred to as LTL_F [DGV13], which will be used later to express temporal logic specifications for our synthesis problem. Properties LTL_F use the same syntax of LTL over infinite traces given in [BKL08]. The LTL_F formulas over a set Π of atomic propositions are obtained as follows:

$$\varphi ::= \top \mid p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid \bigcirc\varphi \mid \Diamond\varphi \mid \Box\varphi \mid \varphi_1 \mathcal{U} \varphi_2,$$

where $p \in \Pi$, $\bigcirc$ is the next operator, $\Diamond$ is eventually, $\Box$ is always, and $\mathcal{U}$ is until. The semantics of LTL_F is given in terms of *finite traces*, i.e., finite words σ, denoting a finite non-empty sequence of consecutive steps over Π. We use $|\sigma|$ to represent the length of σ and σ_i as a propositional interpretation at the ith position in the trace, where $0 \leq i < |\sigma|$. Given a finite trace σ and an LTL_F formula φ, we inductively define when an LTL_F formula φ is true at the ith step ($0 \leq i < |\sigma|$) and denoted by $\sigma, i \models \varphi$, as follows:

- $\sigma, i \models \top$;
- $\sigma, i \models p$, for $p \in \Pi$ iff $p \in \sigma_i$;
- $\sigma, i \models \neg\varphi$ iff $\sigma, i \not\models \varphi$;
- $\sigma, i \models \varphi_1 \wedge \varphi_2$ iff $\sigma, i \models \varphi_1$ and $\sigma, i \models \varphi_2$;
- $\sigma, i \models \varphi_1 \vee \varphi_2$ iff $\sigma, i \models \varphi_1$ or $\sigma, i \models \varphi_2$;
- $\sigma, i \models \bigcirc\varphi$ iff $i < |\sigma| - 1$ and $\sigma, i+1 \models \varphi$;
- $\sigma, i \models \Diamond\varphi$ iff for some j such that $i \leq j < |\sigma|$, we have $\sigma, j \models \varphi$;
- $\sigma, i \models \Box\varphi$ iff for all j such that $i \leq j < |\sigma|$, we have $\sigma, j \models \varphi$;
- $\sigma, i \models \varphi_1 \mathcal{U} \varphi_2$ iff for some j such that $i \leq j < |\sigma|$, we have $\sigma, j \models \varphi_2$, and for all k s.t. $i \leq k < j$, we have $\sigma, k \models \varphi_1$.

The formula φ is true on σ, denoted by $\sigma \models \varphi$, if and only if $\sigma, 0 \models \varphi$. The set of all traces that satisfy the formula φ is called the *language* of formula φ and is denoted by $\mathcal{L}(\varphi)$. Notice that we also have the usual boolean equivalences such as $\varphi_1 \vee \varphi_2 \equiv \neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $\varphi_1 \implies \varphi_2 \equiv \neg\varphi_1 \vee \varphi_2$, $\Diamond\varphi \equiv \top\mathcal{U}\varphi$, and $\Box\varphi \equiv \neg\Diamond\neg\varphi$.

Since safety properties are one of the important class of temporal properties in many practical applications [KV99], we use a subset of LTL_F called safe-LTL_F as introduced in [SRK+14] for our case studies and is defined as follows. .

Definition 5.3.1. *An LTL_F formula is called a safe-LTL_F formula if it can be represented in positive normal form, i.e., negations only occur adjacent to atomic propositions and uses only the temporal operators next ($\bigcirc$) and always ($\Box$).*

Next, we define deterministic finite automata which later serve as equivalent representations of LTL_F formulae.

Definition 5.3.2. *A deterministic finite automaton (DFA) is a tuple $\mathcal{A} = (Q, Q_0, \Sigma, \mathfrak{d}, F)$, where Q is a finite set of states, $Q_0 \subseteq Q$ is a set of initial states, Σ is a finite set (a.k.a. alphabet), $\mathfrak{d} : Q \times \Sigma \to Q$ is a transition function, and $F \subseteq Q$ is a set of accepting states.*

We use notation $q \xrightarrow{\sigma} q'$ to denote transition $(q, \sigma, q') \in \mathfrak{d}$. A finite word $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{n-1}) \in \Sigma^n$ is accepted by DFA $\mathcal{A}$ if there exists a finite state run $\mathbf{q} = (q_0, q_1, \ldots, q_n) \in Q^{n+1}$ such that $q_0 \in Q_0$, $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $0 \leq i < n$ and $q_n \in F$.

The set of words accepted by $\mathcal{A}$ is called the accepting language of $\mathcal{A}$ and is denoted by $\mathcal{L}(\mathcal{A})$. We denote the set of successor states of a state $q \in Q$ by $\Delta(q)$.

The next result shows that every LTL_F formula can be accepted by a DFA.

Theorem 5.3.3 ([ZPV19, DGV15])**.** *Every LTL_F formula φ can be translated to a DFA $\mathcal{A}_\varphi$ that accepts the same language as φ, i.e., $\mathcal{L}(\varphi) = \mathcal{L}(\mathcal{A}_\varphi)$.*

Such $\mathcal{A}_\varphi$ in Theorem 5.3.3 can be constructed explicitly or symbolically using existing tools, such as SPOT [DLLF+16] and MONA [HJJ+95].

Remark 5.3.4. *For a given LTL_F formula φ over atomic propositions Π, the associated DFA $\mathcal{A}_\varphi$ is usually constructed over the alphabet $\Sigma = 2^\Pi$. Solution process of a system Σ_s^{dt} is also connected to the set of words by a labeling function L from the state set to the alphabet Σ. Without loss of generality, we work with the set of atomic propositions directly as the alphabet rather than its power set.*

5.3.2 Property Satisfaction by Stochastic Control Systems

For a given dt-SCS $\Sigma_s^{dt} = (X, V_{\mathsf{w}}, U, \omega, f_s)$ with dynamics (5.2.1), the system Σ_s^{dt} is connected to LTL_F formulas with the help of a measurable labeling function $L : X \to \Pi$, where Π is the set of atomic propositions.

Definition 5.3.5. *Consider a finite state sequence $\mathbf{x}_N = (\mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(N-1)) \in X^N$, $N \in \mathbb{N}$, and labeling function $L : X \to \Pi$. Then, the corresponding trace is given by $L(\mathbf{x}_N) := (\sigma_0, \sigma_1, \ldots, \sigma_{N-1}) \in \Pi^N$ if we have $\sigma_k = L(\mathbf{x}(k))$ for all $k \in [0; N-1]$.*

Note that we abuse the notation by using map $L(\cdot)$ over the domain X^N, i.e. $L(\mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(N-1)) \equiv (L(\mathbf{x}(0)), L(\mathbf{x}(1)), \ldots, L(\mathbf{x}(N-1)))$. Their distinction is clear from the context. Next, we define the probability that a dt-SCS Σ_s^{dt} satisfies LTL_F formula φ over traces of length N.

Definition 5.3.6. *Consider a dt-SCS $\Sigma_s^{dt} = (X, V_{\mathsf{w}}, U, \omega, f_s)$ and an LTL_F formula φ over Π. We denote by $\mathbb{P}_\upsilon^{x_0}\{L(\boldsymbol{x}_N) \models \varphi\}$ the probability that φ is satisfied by the state evolution of the system Σ_s^{dt} over a finite-time horizon $[0, N) \subset \mathbb{N}$ starting from initial state $\mathbf{x}(0) = x_0 \in X$ under control policy υ.*

Remark 5.3.7. *The set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$ and the labeling function $L : X \to \Pi$ provide a measurable partition of the state set $X = \cup_{i=1}^M X_i$ as $X_i := L^{-1}(p_i)$. We assume that $X_i \neq \emptyset$ for any i. This assumption is without loss of generality since all the atomic propositions p_i with $L^{-1}(p_i) = \emptyset$ can be replaced by $(\bot)$ without affecting the probability of satisfaction.*

5.3.3 Problem formulation

Problem 5.3.8. *Given a dt-SCS $\Sigma_s^{dt} = (X, V_{\mathsf{w}}, U, \omega, f_s)$ with dynamics (5.2.1), an LTL_F specification φ of length N over a set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$, a labelling function $L : X \to \Pi$, and real value $\vartheta \in (0, 1)$, compute a control policy υ (if existing) such that $\mathbb{P}_\upsilon^{x_0}\{L(\boldsymbol{x}_N) \models \varphi\} \geq \vartheta$ for all $x_0 \in L^{-1}(p_i)$ and some $i \in [0; M]$.*

Finding a solution to Problem 5.3.8 (if existing) is difficult in general. In this chapter, we give a computational method that is sound in solving the problem. Our approach is to compute a policy υ together with a lower bound $\underline{\vartheta}$. We try to find the largest lower bound, which then can be compared with ϑ and gives υ as a solution for Problem 5.3.8 if $\underline{\vartheta} \geq \vartheta$. To solve this problem, we utilize the notion of control barrier functions (discussed in Section 5.4). In general, this notion is useful for providing an upper bound on the reachability probability. The negation of LTL_F properties can be equivalently represented as a sequence of reachability problems using a DFA. Therefore, instead of computing a control policy that guarantees a lower bound $\underline{\vartheta}$ on the probability satisfaction of the LTL_F specification, we compute a policy that guarantees an upper bound on the probability satisfaction of its negation, *i.*e., $\mathbb{P}_{\upsilon}^{x_0}\{L(\mathbf{x}_N) \models \neg\varphi\} \leq \overline{\vartheta}$ for any $x_0 \in L^{-1}(p_i)$ and some $i \in \{0, 1, \ldots, M\}$. Then for the same control policy the lower bound can be easily obtained as $\underline{\vartheta} = 1 - \overline{\vartheta}$. This is done by constructing a DFA $\mathcal{A}_{\neg\varphi} = (Q, Q_0, \Pi, \mathfrak{d}, F)$ that accepts all finite words over Π satisfying $\neg\varphi$.

For the sake of illustrating the results better, we provide the following running example throughout this chapter.

Example 5.3.9. *Consider a two-dimensional dt-SCS* $\Sigma_s^{dt} = (X, V_w, U, \omega, f_s)$ *with* $X = V_w = \mathbb{R}^2$, $U = \mathbb{R}$ *and dynamics*

$$\begin{aligned} \mathbf{x}_1(k+1) &= \mathbf{x}_1(k) - 0.01\mathbf{x}_2^2(k) + 0.5\omega_1(k), \\ \mathbf{x}_2(k+1) &= -0.01\mathbf{x}_1(k)\mathbf{x}_2(k) + \upsilon(k) + 0.5\omega_2(k), \end{aligned} \tag{5.3.1}$$

where $\upsilon(\cdot)$ *is a control input and* $\omega_1(k)$, $\omega_2(k)$ *are standard normal random variables that are independent from each other for any* $k \in \mathbb{N}_0$. *The set of atomic propositions is given by* $\Pi = \{p_0, p_1, p_2, p_3\}$, *with labeling function* $L(x) = p_i$ *for any* $x \in X_i$, $i \in \{0, 1, 2, 3\}$. *The sets* X_i *are defined as*

$$\begin{aligned} X_0 &= \{(x_1, x_2) \in X \mid (x_1 + 5)^2 + x_2^2 \leq 2.5\}, \\ X_1 &= \{(x_1, x_2) \in X \mid (x_1 - 5)^2 + (x_2 - 5)^2 \leq 3\}, \\ X_2 &= \{(x_1, x_2) \in X \mid (x_1 - 4)^2 + (x_2 + 3)^2 \leq 2\}, \text{ and} \\ X_3 &= X \setminus (X_0 \cup X_1 \cup X_2). \end{aligned}$$

These sets are shown in Figure 5.1.

We are interested in computing a control policy υ *that provides a lower bound on the probability that the trajectories of* Σ_s^{dt} *of length* N *satisfies the following specification:*

- *If it starts in* X_0, *it will always stay away from* X_1 *or always stay away from* X_2. *If it starts in* X_2, *it will always stay away from* X_1.

This property can be expressed by the LTL$_F$ *formula*

$$\varphi = (p_0 \wedge (\Box\neg p_1 \vee \Box\neg p_2)) \vee (p_2 \wedge \Box\neg p_1). \tag{5.3.2}$$

The DFA corresponding to the negation of φ *in* (5.3.2) *is shown in Figure 5.2.*

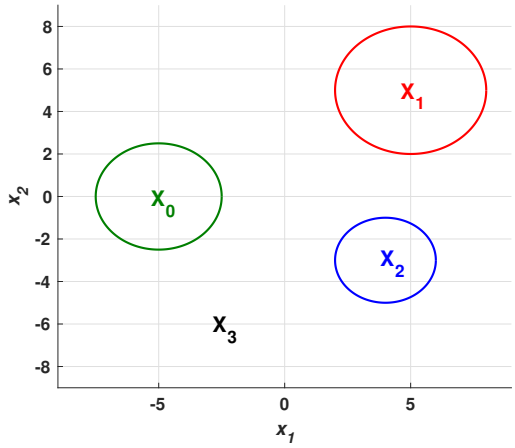

Figure 5.1: State set and regions of interest for Example 1.

5.4 Control Barrier Functions

In this section, we introduce the notion of control barrier function which will later serve as the core element for solving Problem 5.3.8.

Definition 5.4.1. *A function $\mathcal{B} : X \to \mathbb{R}_0^+$ is a control barrier function for a dt-SCS $\Sigma_s^{dt} = (X, V_w, U, \omega, f_s)$ if for any state $x \in X$, there exists an input $u \in U$ such that*

$$\mathbb{E}[\mathcal{B}(f_s(x, u, w)) \mid x, u] \leq \mathcal{B}(x) + c, \tag{5.4.1}$$

for some constant $c \geq 0$.

If the set of control inputs U is finite, one can rewrite Definition 5.4.1 as follows.

Definition 5.4.2. *A function $\mathcal{B} : X \to \mathbb{R}_0^+$ is a control barrier function for a dt-SCS $\Sigma_s^{dt} = (X, V_w, U, \omega, f_s)$ with $U = \{u_1, u_2, \ldots, u_l\}$, $l \in \mathbb{N}$, if*

$$\min_{u \in U} \mathbb{E}[\mathcal{B}(f_s(x, u, w)) \mid x, u] \leq \mathcal{B}(x) + c \quad \forall x \in X, \tag{5.4.2}$$

for some constant $c \geq 0$.

Remark 5.4.3. *Note that conditions* (5.4.1)-(5.4.2) *are relaxed versions of so-called supermartingale conditions. This is due to the positive constant c on the right-hand side. When $c = 0$, the function $\mathcal{B}(\cdot)$ becomes supermartingale for Σ_s^{dt}.*

Remark 5.4.4. *The above definitions associate a stationary policy* u *to a control barrier function. Definition 5.4.1 gives such a policy according to the existential quantifier on the input for any state $x \in X$. Definition 5.4.2 gives the policy as the arg*min *of the left-hand side of inequality* (5.4.2)*. In the case of discrete inputs, the input $v(k)$ can be selected*

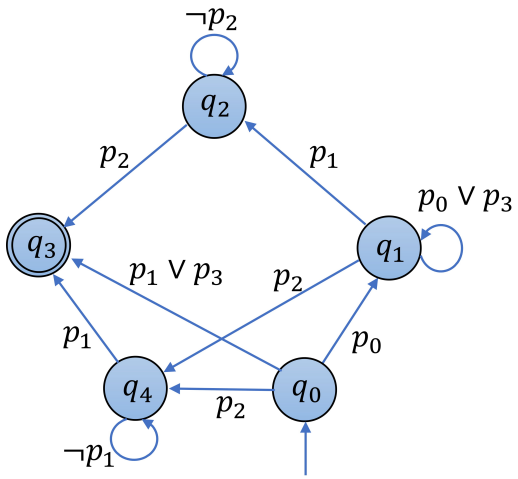

Figure 5.2: DFA $\mathcal{A}_{\neg\varphi}$ that accepts all traces satisfying $\neg\varphi$ where φ is given in (5.3.2).

as an element of $\{\upsilon(k) \in U \mid \mathbb{E}[\mathcal{B}(f_s(\mathbf{x}k), \upsilon(k))) \mid \mathbf{x}(k), \upsilon(k)] \leq \mathcal{B}(\mathbf{x}(k)) + c\}$ *for* $k \in \mathbb{N}_0$. *In other words, Definition 5.4.2 provides regions of state-space in which the particular control input is valid and is given as* $\mathsf{X}_i := \{x \in X \mid \mathbb{E}[\mathcal{B}(f_s(x, u_i)) \mid x, u_i] \leq \mathcal{B}(x) + c\}$ *for all* $i \in [1; l]$ *and* $\bigcup_i \mathsf{X}_i = X$.

We provide the following lemma and use it in the sequel. This lemma is a direct consequence of [Kus67, Theorem 3] and is also utilized in [ST12, Theorem II.1].

Lemma 5.4.5. *Consider a dt-SCS* $\Sigma_s^{dt} = (X, V_w, U, \omega, f_s)$ *and let* $\mathcal{B} : X \to \mathbb{R}_0^+$ *be a control barrier function as given in Definition 5.4.1 (or Definition 5.4.2) with constant* c *and stationary policy* u. *Then for any constant* $\overline{\lambda} > 0$ *and any initial state* $x_0 \in X$,

$$\mathbb{P}_{\mathsf{u}}^{x_0}\{\sup_{0 \leq k < T_d} \mathcal{B}(\mathbf{x}(k)) \geq \overline{\lambda} \mid \mathbf{x}(0) = x_0\} \leq \frac{\mathcal{B}(x_0) + cT_d}{\overline{\lambda}}. \tag{5.4.3}$$

Proof. The proof is similar to that of Theorem 3 in [Kus67] and is omitted here. □

Next theorem shows that a control barrier function can give an upper bound on the probability of satisfying reachability specification. This theorem is inspired by the result of [PJP07, Theorem 15] that uses supermartingales for reachability analysis of continuous-time stochastic systems.

Theorem 5.4.6. *Consider a dt-SCS* $\Sigma_s^{dt} = (X, V_w, U, \omega, f_s)$ *and sets* $X_0, X_1 \subseteq X$. *Suppose there exist a control barrier function* $\mathcal{B} : X \to \mathbb{R}_0^+$ *as defined in Definition 5.4.1 (or Definition 5.4.2) with constant* $c \geq 0$ *and stationary policy* u. *If there is a constant* $\varrho \in [0, 1]$ *such that*

$$\mathcal{B}(x) \leq \varrho \qquad \forall x \in X_0, \tag{5.4.4}$$

$$\mathcal{B}(x) \geq 1 \qquad \forall x \in X_1, \tag{5.4.5}$$

then the probability that the state evolution of Σ_s^{dt} starts from any initial state $x_0 \in X_0$ and reaches X_1 under stationary policy u within time horizon $[0, T_d) \subseteq \mathbb{N}_0$ is upper bounded by $\varrho + cT_d$.

Proof. Since $\mathcal{B}(\mathbf{x}(k))$ is a control barrier function, we conclude that (5.4.3) in Lemma 5.4.5 holds. Now using (5.4.4) and the fact that $X_1 \subseteq \{x \in X \mid \mathcal{B}(x) \geq 1\}$, we have $\mathbb{P}_{\mathsf{u}}^{x_0}\{\mathbf{x}(k) \in X_1 \text{ for some } 0 \leq k < T_d \mid \mathbf{x}(0) = x_0\} \leq \mathbb{P}_{\mathsf{u}}^{x_0}\{\sup_{0\leq k<T_d} \mathcal{B}(\mathbf{x}(k)) \geq 1 \mid \mathbf{x}(0) = x_0\} \leq \mathcal{B}(x_0) + cT_d \leq \varrho + cT_d$, which concludes the proof. □

Theorem 5.4.6 enables us to formulate an optimization problem for finding a sound solution of the policy synthesis problem 5.3.8 with reachability specifications. We can minimize the values of ϱ and c in order to find an upper bound for finite-horizon reachability that is as tight as possible.

Remark 5.4.7. *If one succeeds in finding a control barrier function $\mathcal{B}(\cdot)$ with $c = 0$ satisfying conditions of Theorem 5.4.6, the result of the theorem holds for an unbounded time horizon. However, considering relaxed supermartingale condition as discussed in Remark 5.4.3, makes it easier to find $\mathcal{B}(\cdot)$ satisfying conditions in Theorem 5.4.6 and makes our results applicable to larger classes of systems.*

In the next section, we address general LTL_F specifications and discuss how to translate the synthesis problem 5.3.8 for any LTL_F specification into the computation of a collection of control barrier functions each satisfying conditions in Theorem 5.4.6.

5.5 Decomposition into Sequential Reachability

Consider a DFA $\mathcal{A}_{\neg\varphi} = (Q, Q_0, \Pi, \mathfrak{d}, F)$ that accepts all finite words of length $n \in [0, N] \subset \mathbb{N}_0$ satisfying $\neg\varphi$.

Accepting state run of $\mathcal{A}_{\neg\varphi}$. For any $n \in \mathbb{N}_0$, sequence $\mathbf{q} = (q_0, q_1, \ldots, q_n) \in Q^{n+1}$ is called an accepting state run if $q_0 \in Q_0$, $q_n \in F$, and there exist a finite word $\sigma = (\sigma_0, \sigma_1, \ldots, \sigma_{n-1}) \in \Pi^n$ such that $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $i \in [0; n-1]$. We denote the set of such finite words by $\sigma(\mathbf{q}) \subseteq \Pi^n$ and the set of accepting state runs by $\mathcal{R}$. We also indicate the length of $\mathbf{q} \in Q^{n+1}$ by $|\mathbf{q}|$, which is $n + 1$.

Self-loops in the DFA play a central role in our decomposition. Let $Q_s \subseteq Q$ be a set of states of $\mathcal{A}_{\neg\varphi}$ having self-loops, *i.*e., $Q_s := \{q \in Q \mid \exists p \in \Pi, q \xrightarrow{p} q\}$. Let $\mathcal{R}_N$ be the set of all finite accepting state runs of lengths less than or equal to $N + 1$ excluding self-loops,

$$\mathcal{R}_N := \{\mathbf{q} = (q_0, q_1, \ldots, q_n) \in \mathcal{R} \mid n \leq N,\, q_i \neq q_{i+1},\, \forall i < n\}. \tag{5.5.1}$$

Computation of $\mathcal{R}_N$ can be done efficiently using algorithms in graph theory by viewing $\mathcal{A}_{\neg\varphi}$ as a directed graph. Consider $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ as a directed graph with vertices $\mathcal{V} = Q$ and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ such that $(q, q') \in \mathcal{E}$ if and only if $q' \neq q$ and there exist $p \in \Pi$ such that $q \xrightarrow{p} q'$. For any $(q, q') \in \mathcal{E}$, we denote the atomic proposition associated with the edge (q, q') by $\sigma(q, q')$. From the construction of the graph, it is obvious that the finite

Algorithm 2 Computation of sets $\mathcal{P}^p(\mathbf{q})$, $\mathbf{q} \in \mathcal{R}_N^p$, $p \in \Pi$

Require: $\mathcal{G}$, Q_s, N, Π
Initialize:
  $\mathcal{P}^p(\mathbf{q}) \leftarrow \emptyset, \quad \forall p \in \Pi$
Compute set $\mathcal{R}_N$ by depth first search on $\mathcal{G}$
for all $\mathbf{q} = (q_0, q_1, \ldots, q_n) \in \mathcal{R}_N$ and $p \in \Pi$ **do**
  if $p = \sigma(q_0, q_1)$ **then**
    $\mathcal{R}_N^p \leftarrow \{\mathbf{q}\}$
for all $p \in \Pi$ and $\mathbf{q} \in \mathcal{R}_N^p$ and $|\mathbf{q}| \geq 3$ **do**
  for $i = 0$ to $|\mathbf{q}| - 3$ **do**
    $\mathcal{P}_{temp}(\mathbf{q}) \leftarrow \{(q_i, q_{i+1}, q_{i+2})\}$
    if $q_{i+1} \in Q_s$ **then**
      $\mathcal{P}^p(\mathbf{q}) \leftarrow \{(q_i, q_{i+1}, q_{i+2}, N + 2 - |\mathbf{q}|)\}$
    else
      $\mathcal{P}^p(\mathbf{q}) \leftarrow \{(q_i, q_{i+1}, q_{i+2}, 1)\}$
return $\mathcal{P}^p(\mathbf{q}), \quad \forall p \in \Pi$

path in the graph of length $n+1$ starting from vertices $q_0 \in Q_0$ and ending at $q_F \in F$ is an accepting state run $\mathbf{q}$ of $\mathcal{A}_{\neg\varphi}$ without any self-loop thus belongs to $\mathcal{R}_N$. Then one can easily compute $\mathcal{R}_N$ using variants of depth first search algorithm [RNC+03]. For each $p \in \Pi$, we define a set $\mathcal{R}_N^p$ as

$$\mathcal{R}_N^p := \{\mathbf{q} = (q_0, q_1, \ldots, q_n) \in \mathcal{R}_N \mid \sigma(q_0, q_1) = p \in \Pi\}. \tag{5.5.2}$$

Note that we use the superscript $p \in \Pi$ to represent the atomic proposition corresponding to the initial region from which the state evolution starts. We use a similar notation throughout the chapter. Decomposition into sequential reachability is performed as follows. For any $\mathbf{q} = (q_0, q_1, \ldots, q_n) \in \mathcal{R}_N^p$, we define $\mathcal{P}^p(\mathbf{q})$ as a set of all state runs of length 3 augmented with a horizon,

$$\mathcal{P}^p(\mathbf{q}) := \{(q_i, q_{i+1}, q_{i+2}, T(\mathbf{q}, q_{i+1})) \mid 0 \leq i \leq n-2\}, \tag{5.5.3}$$

where the horizon is defined as $T(\mathbf{q}, q_{i+1}) = N + 2 - |\mathbf{q}|$ for $q_{i+1} \in Q_s$ and 1 otherwise. We denote $\mathcal{P}(\mathcal{A}_{\neg\varphi}) = \bigcup_{p\in\Pi} \bigcup_{\mathbf{q}\in\mathcal{R}_N^p} \mathcal{P}^p(\mathbf{q})$.

Remark 5.5.1. *Note that $\mathcal{P}^p(\mathbf{q}) = \emptyset$ for $|\mathbf{q}| = 2$. In fact, any accepting state run of length 2 specifies a subset of the state set such that the system satisfies $\neg\varphi$ whenever it starts from that subset. This gives trivial zero probability for satisfying the specification, thus neglected in the sequel.*

The computation of sets $\mathcal{P}^p(\mathbf{q})$, $\mathbf{q} \in \mathcal{R}_N^p$, $p \in \Pi$, is illustrated in Algorithm 2 and demonstrated below for our running example.

Example 5.5.2. (*Example 5.3.9 continued*) *For LTL_F formula φ given in* (5.3.2)*, Figure 5.2 shows a DFA $\mathcal{A}_{\neg\varphi}$ that accepts all words that satisfy $\neg\varphi$. From Figure 5.2, we get*

$Q_0 = \{q_0\}$, $\Pi = \{p_0, p_1, p_2, p_3\}$ and $F = \{q_3\}$. We consider traces of maximum length $N = 5$. The set of accepting state runs of lengths at most $N + 1$ without self-loops is

$$\mathcal{R}_5 = \{(q_0, q_4, q_3), (q_0, q_1, q_2, q_3), (q_0, q_1, q_4, q_3), (q_0, q_3)\}.$$

The sets $\mathcal{R}_5^p$ for $p \in \Pi$ are as follows:

$$\mathcal{R}_5^{p_0} = \{(q_0, q_1, q_2, q_3), (q_0, q_1, q_4, q_3)\}, \mathcal{R}_5^{p_1} = \{(q_0, q_3)\}, \mathcal{R}_5^{p_2} = \{(q_0, q_4, q_3)\}, \mathcal{R}_5^{p_3} = \{(q_0, q_3)\}.$$

The set of states with self-loops is $Q_s = \{q_1, q_2, q_4\}$. Then the sets $\mathcal{P}^p(\mathbf{q})$ for $\mathbf{q} \in \mathcal{R}_5^p$ are as follows:

$$\begin{aligned}
&\mathcal{P}^{p_0}(q_0, q_1, q_2, q_3) = \{(q_0, q_1, q_2, 3), (q_1, q_2, q_3, 3)\},\\
&\mathcal{P}^{p_0}(q_0, q_1, q_4, q_3) = \{(q_0, q_1, q_4, 3), (q_1, q_4, q_3, 3)\},\\
&\mathcal{P}^{p_1}(q_0, q_3) = \mathcal{P}^{p_3}(q_0, q_3) = \emptyset, \quad \mathcal{P}^{p_2}(q_0, q_4, q_3) = \{(q_0, q_4, q_3, 4)\}.
\end{aligned}$$

For every $\mathbf{q} \in \mathcal{R}_5^p$, the corresponding finite words $\sigma(\mathbf{q})$ are listed as follows:

$$\begin{aligned}
&\sigma(q_0, q_3) = \{p_1\}, \quad \sigma(q_0, q_4, q_3) = \{(p_2, p_1)\},\\
&\sigma(q_0, q_1, q_2, q_3) = \{(p_0, p_1, p_2)\}, \quad \sigma(q_0, q_1, q_4, q_3) = \{(p_0, p_2, p_1)\}.
\end{aligned}$$

□

5.6 Controller Synthesis using Control Barrier functions

Having $\mathcal{P}^p(\mathbf{q})$ defined in (5.5.3) as the set of state runs of length 3 augmented with a horizon, in this section, we provide a systematic approach to compute a policy with a (potentially tight) lower bound on the probability that the state evolutions of Σ_s^{dt} satisfies φ. Given DFA $\mathcal{A}_{\neg\varphi}$, our approach relies on performing a reachability computation over each element of $\mathcal{P}(\mathcal{A}_{\neg\varphi})$, where reachability probability is upper bounded using control barrier functions along with appropriate choices of control inputs as mentioned in Theorem 5.4.6. However, computation of control barrier functions and the policies for each element $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, can cause ambiguity while utilizing controllers in closed-loop whenever there are more than one outgoing edges from a state of the automaton. To make it more clear, consider elements $\nu_1 = (q_0, q_1, q_2, T((q_0, q_1, q_2, q_3), q_1))$ and $\nu_2 = (q_0, q_1, q_4, T((q_0, q_1, q_4, q_3), q_1))$ from Example 1, where there are two outgoing transitions from state q_1 (see Figure 5.2). This results in two different reachability problems, namely, reaching sets $L^{-1}(\sigma(q_1, q_2))$ and $L^{-1}(\sigma(q_1, q_4))$ starting from the same set $L^{-1}(\sigma(q_0, q_1))$. Thus computing different control barrier functions and corresponding controllers in such a scenario is not helpful. To resolve this ambiguity, we simply merge such reachability problems into one reachability problem by replacing the reachable set X_1 in Theorem 5.4.6 with the union of regions corresponding to the alphabets of all outgoing edges. Thus we get a common control barrier function and a corresponding controller. This enables us to partition $\mathcal{P}(\mathcal{A}_{\neg\varphi})$ and put the elements sharing a common

control barrier function and a corresponding control policy in the same partition set. These sets can be formally defined as

$$\mathfrak{S}_{(q,q',\Delta(q'))} := \{(q, q', q'', T) \in \mathcal{P}(\mathcal{A}_{\neg\varphi}) \mid q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\}.$$

The control barrier function and the controller corresponding to the partition set $\mathfrak{S}_{(q,q',\Delta(q'))}$ are denoted by $\mathcal{B}_{\mathfrak{S}_{(q,q',\Delta(q'))}}(x)$ and $\mathsf{u}_{\mathfrak{S}_{(q,q',\Delta(q'))}}(x)$, respectively. Thus, for all $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, we have

$$\mathcal{B}_\nu(x) = \mathcal{B}_{\mathfrak{S}_{(q,q',\Delta(q'))}}(x) \text{ and } \mathsf{u}_\nu(x) = \mathsf{u}_{\mathfrak{S}_{(q,q',\Delta(q'))}}(x), \quad \text{if } \nu \in \mathfrak{S}_{(q,q',\Delta(q'))}. \tag{5.6.1}$$

5.6.1 Control Policy

From the above discussion, one can readily observe that we have different control policies at different locations of the automaton which can be interpreted as a switching control policy. Next, we define the automaton representing the switching mechanism for control policies. Consider the DFA $\mathcal{A}_{\neg\varphi} = (Q, Q_0, \Pi, \mathfrak{d}, F)$ corresponding to $\neg\varphi$ as discussed in Section 5.5, then the switching mechanism is given by a DFA $\mathcal{A}_\mathfrak{m} = (Q_\mathfrak{m}, Q_{\mathfrak{m}0}, \Pi_\mathfrak{m}, \mathfrak{d}_\mathfrak{m}, F_\mathfrak{m})$, where $Q_\mathfrak{m} := Q_{\mathfrak{m}0} \cup \{(q, q', \Delta(q')) \mid q, q' \in Q \setminus F\} \cup F_\mathfrak{m}$ is the set of states, $Q_{\mathfrak{m}0} := \{(q_0, \Delta(q_0)) \mid q_0 \in Q_0\}$ is a set of initial states, $\Pi_\mathfrak{m} = \Pi$, $F_\mathfrak{m} = F$, and the transition relation $(q_\mathfrak{m}, \sigma, q'_\mathfrak{m}) \in \mathfrak{d}_\mathfrak{m}$ is defined as

- for all $q_\mathfrak{m} = (q_0, \Delta(q_0)) \in Q_{\mathfrak{m}0}$,
  - $(q_0, \Delta(q_0)) \overset{\sigma(q_0,q'')}{\longrightarrow} (q_0, q'', \Delta(q''))$, where $q'' \in \Delta(q_0)$;
- for all $q_\mathfrak{m} = (q, q', \Delta(q')) \in Q_\mathfrak{m} \setminus (Q_{\mathfrak{m}0} \cup F_\mathfrak{m})$,
  - $(q, q', \Delta(q')) \overset{\sigma(q',q'')}{\longrightarrow} (q', q'', \Delta(q''))$, such that $q, q', q'' \in Q$, $q'' \in \Delta(q')$ and $q'' \notin F$; and
  - $(q, q', \Delta(q')) \overset{\sigma(q',q'')}{\longrightarrow} q''$, such that $q, q', q'' \in Q$, $q'' \in \Delta(q')$ and $q'' \in F$.

The control policy that is a candidate for solving Problem 5.3.8 is given as

$$\upsilon(x, q_\mathfrak{m}) = \mathsf{u}_{\mathfrak{S}_{(q'_\mathfrak{m})}}(x), \quad \forall (q_\mathfrak{m}, L(x), q'_\mathfrak{m}) \in \mathfrak{d}_\mathfrak{m}. \tag{5.6.2}$$

In the next subsection, we discuss the computation of bound on the probability of satisfying the specification under such a policy, which then can be used for checking if this policy is indeed a solution for Problem 5.3.8.

Remark 5.6.1. *The control policy in* (5.6.2) *is a Markov policy on the augmented space $X \times Q_\mathfrak{m}$. Such a policy is equivalent to a history dependent policy on the state set X of the system as discussed in Section 5.2 (see [TMKA13] for a proof).*

Example 5.6.2. *(Example 5.3.9 continued) The DFA $\mathcal{A}_\mathfrak{m} = (Q_\mathfrak{m}, Q_{\mathfrak{m}0}, \Pi_\mathfrak{m}, \mathfrak{d}_\mathfrak{m}, F_\mathfrak{m})$ modeling the switching mechanism between policies for the system in Example 5.3.9 is shown in Figure 5.3.* □

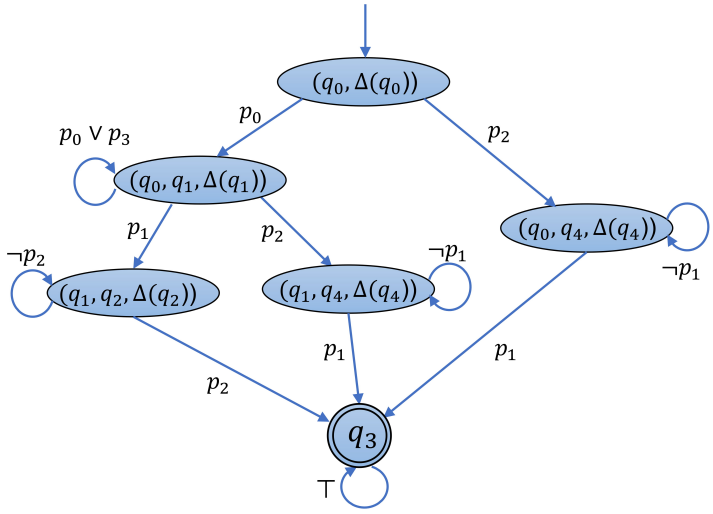

Figure 5.3: DFA $\mathcal{A}_{\mathrm{m}}$ representing switching mechanism for controllers for Example 5.3.9.

5.6.2 Computation of Probabilities

Next theorem provides an upper bound on the probability that the state evolution of the system satisfies the specification $\neg\varphi$.

Theorem 5.6.3. *For a given LTL$_F$ specification φ, let $\mathcal{A}_{\neg\varphi}$ be a DFA corresponding to its negation. For $p \in \Pi$, let $\mathcal{R}_N^p$ be the set defined in* (5.5.2)*, and $\mathcal{P}^p$ be the set of runs of length 3 augmented with a horizon defined in* (5.5.3)*. The probability that the state evolution of Σ_s^{dt} starting from any initial state $x_0 \in L^{-1}(p)$ under the control policy in* (5.6.2) *satisfies $\neg\varphi$ within time horizon $[0, N) \subseteq \mathbb{N}_0$ is upper bounded by*

$$\mathbb{P}_v^{x_0}\{L(\boldsymbol{x}_N) \models \neg\varphi\} \leq \sum_{\mathbf{q}\in\mathcal{R}_N^p} \prod \left\{(\varrho_\nu + c_\nu T) \,|\, \nu = (q, q', q'', T) \in \mathcal{P}^p(\mathbf{q})\right\}, \tag{5.6.3}$$

where $\varrho_\nu + c_\nu T$ is computed via Theorem 5.4.6 which is the upper bound on the probability of the trajectories of Σ_s^{dt} starting from $X_0 := L^{-1}(\sigma(q, q'))$ and reaching $X_1 := L^{-1}(\sigma(q', q''))$ within time horizon $[0, T) \subseteq \mathbb{N}_0$.

Proof. For $p \in \Pi$, consider an accepting run $\mathbf{q} \in \mathcal{R}_N^p$ and set $\mathcal{P}^p(\mathbf{q})$ as defined in (5.5.3). We apply Theorem 5.4.6 to any $\nu = (q, q', q'', T) \in \mathcal{P}^p(\mathbf{q})$. The probability that the state evolution of Σ_s^{dt} starts from any initial state $x_0 \in L^{-1}(\sigma(q, q'))$ and reaches $L^{-1}(\sigma(q', q''))$ under control input $u_\nu(x)$ within time horizon $[0, T] \subseteq \mathbb{N}_0$ is upper bounded by $\varrho_\nu + c_\nu T$. Now the upper bound on the probability of the trace of the state evolution (*i.e.*, $L(\mathbf{x}_N)$) reaching accepting state following trace corresponding to $\mathbf{q}$ is given by the product of the probability bounds corresponding to all elements $\nu = (q, q', q'', T) \in \mathcal{P}^p(\mathbf{q})$ and is given by

$$\mathbb{P}\{\sigma(\mathbf{q}) \models \neg\varphi\} \leq \prod \left\{(\varrho_\nu + c_\nu T) \,|\, \nu = (q, q', q'', T) \in \mathcal{P}^p(\mathbf{q})\right\}. \tag{5.6.4}$$

Note that, the way we computed time horizon T, we always get the upper bound for the probabilities for all possible combinations of self-loops for accepting state runs of length

less than or equal to $N+1$. The upper bound on the probability that the state evolution of the system Σ_s^{dt} starting from any initial state $x_0 \in L^{-1}(p)$ violating φ can be computed by summing the probability bounds for all possible accepting runs as computed in (5.6.4) and is given by

$$\mathbb{P}_v^{x_0}\{L(\mathbf{x}_N) \models \neg\varphi\} \leq \sum_{\mathbf{q}\in\mathcal{R}_N^p} \prod \left\{(\varrho_\nu + c_\nu T) \,|\, \nu = (q, q', q'', T) \in \mathcal{P}^p(\mathbf{q})\right\}.$$

□

Theorem 5.6.3 enables us to decompose the computation into a collection of sequential reachability, compute bounds on the reachability probabilities using Theorem 5.4.6, and then combine the bounds in a sum-product expression.

Remark 5.6.4. *In case we are unable to find control barrier functions for some of the elements $\nu \in \mathcal{P}^p(\mathbf{q})$ in (5.6.3), we replace the related term $(\varrho_\nu + c_\nu T)$ by the pessimistic bound 1. In order to get a non-trivial bound in (5.6.3), at least one control barrier function must be found for each $\mathbf{q} \in \mathcal{R}_N^p$.*

Corollary 5.6.5. *Given the result of Theorem 5.6.3, the probability that the trajectories of Σ_s^{dt} of length N starting from any $x_0 \in L^{-1}(p)$ satisfies LTL_F specification φ is lower-bounded by*

$$\mathbb{P}_v^{x_0}\{L(\boldsymbol{x}_N) \models \varphi\} \geq 1 - \mathbb{P}_v^{x_0}\{L(\boldsymbol{x}_N) \models \neg\varphi\}.$$

5.7 Computation of Control Barrier function

Proving the existence of a control barrier function and finding one are in general hard problems. But if we restrict the class of systems and labeling functions, we can construct computationally efficient techniques to search for control barrier functions and corresponding control policies of specific forms. In this subsection, we provide two possible approaches for computing control barrier functions and corresponding control policies for dt-SCS Σ_s^{dt} with respectively continuous and discrete input sets.

5.7.1 Continuous Input Sets

We propose a technique using sum-of-squares (SOS) optimization [Par03], relying on the fact that a polynomial is non-negative if it can be written as a sum of squares of different polynomials. In order to utilize an SOS optimization, we raise the following assumption.

Assumption 5.7.1. *System Σ_s^{dt} has a continuous state set $X \subseteq \mathbb{R}^n$ and a continuous input set $U \subseteq \mathbb{R}^m$. Its vector field $f_s : X \times U \times V_w \to X$ is a polynomial function of state x and input u for any $w \in V_w$. Partition sets $X_i = L^{-1}(p_i)$, $i \in [0; M]$, are bounded semi-algebraic sets, i.e., they can be represented by polynomial equalities and inequalities.*

Under Assumption 5.7.1, we can formulate conditions in Theorem 5.4.6 as an SOS optimization to search for a polynomial control barrier function $\mathcal{B}(\cdot)$, a polynomial control policy $u(\cdot)$ and an upper bound $(\varrho + cT_d)$. The following lemma provides a set of sufficient conditions for the existence of such control barrier function required in Theorem 5.4.6, which can be solved as an SOS optimization.

Lemma 5.7.2. *Suppose Assumption 5.7.1 holds and sets X_0, X_1, X can be defined by vectors of polynomial inequalities $X_0 = \{x \in \mathbb{R}^n \mid \mathsf{g}_0(x) \geq 0\}$, $X_1 = \{x \in \mathbb{R}^n \mid \mathsf{g}_1(x) \geq 0\}$, and $X = \{x \in \mathbb{R}^n \mid \mathsf{g}(x) \geq 0\}$, where the inequalities are defined element-wise. Suppose there exists a sum-of-square polynomial $\mathcal{B}(x)$, constants $\varrho \in [0, 1]$ and $c \geq 0$, polynomials $\Upsilon_{u_i}(x)$ corresponding to the i^{th} input in $u = (u_1, u_2, \ldots, u_m) \in U \subseteq \mathbb{R}^m$, and vectors of sum-of-squares polynomials $\Upsilon_0(x)$, $\Upsilon_1(x)$, and $\Upsilon(x)$ of appropriate size such that following expressions are sum-of-squares polynomials*

$$-\mathcal{B}(x) - \Upsilon_0^T(x)\mathsf{g}_0(x) + \varrho \tag{5.7.1}$$

$$\mathcal{B}(x) - \Upsilon_1^T(x)\mathsf{g}_1(x) - 1 \tag{5.7.2}$$

$$-\mathbb{E}[\mathcal{B}(f_s(x,u,w))|x,u] + \mathcal{B}(x) - \sum_{i=1}^{m}(u_i - \Upsilon_{u_i}(x)) - \Upsilon^T(x)\mathsf{g}(x) + c. \tag{5.7.3}$$

Then, $\mathcal{B}(x)$ satisfies conditions in Theorem 5.4.6 and $u_i = \Upsilon_{u_i}(x)$ gives the corresponding stationary control policy.

Proof. Since the entries $\mathcal{B}(x)$ and $\Upsilon_0(x)$ in $-\mathcal{B}(x) - \Upsilon_0^T(x)\mathsf{g}_0(x) + \varrho$ are sum-of-squares, we have $0 \leq \mathcal{B}(x) + \Upsilon_0^T(x)\mathsf{g}_0(x) \leq \varrho$. Since the term $\Upsilon_0^T(x)\mathsf{g}_0(x)$ is non-negative over X_0, (5.7.1) implies condition (5.4.4) in Theorem 5.4.6. Similarly, we can show that (5.7.2) implies condition (5.4.5) in Theorem 5.4.6. Now consider (5.7.3). If we choose control input $u_i = \Upsilon_{u_i}(x)$ and since the term $\Upsilon^T(x)\mathsf{g}(x)$ is non-negative over set X, we have $\mathbb{E}[\mathcal{B}(f_s(x,u,w))|x,u] \leq \mathcal{B}(x) + c$ which implies that the function $\mathcal{B}(x)$ is a control barrier function. This concludes the proof. □

Remark 5.7.3. *Assumption 5.7.1 is essential for applying the results of Lemma 5.7.2 to* any *LTL_F specification. For a given specification, we can relax this assumption and allow some of the partition sets X_i to be unbounded. For this, we require that the labels corresponding to unbounded partition sets should only appear either on self-loops or on accepting runs of length less than 3. For instance, Example 1 has an unbounded partition set X_3 and its corresponding label p_3 satisfies this requirement (see Figure 5.1), thus the results are still applicable.*

Based on Lemma 5.7.2, for any $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, a polynomial control barrier function $B_\nu(x)$ and controller $\mathsf{u}_\nu(x)$ as in (5.6.1) can be computed using SOSTOOLS [PPP02] in conjunction with a semidefinite programming solver such as SeDuMi [Stu99]. The computed barrier function will satisfy conditions in Theorem 5.4.6 while minimizing constants ϱ_ν and c_ν. Having values of ϱ_ν and c_ν for all $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, one can simply utilize results of Theorem 5.6.3 and Corollary 5.6.5 to compute a lower bound on the probability of satisfying the given specification to check the solution to Problem 5.3.8.

Table 5.1: Controllers $u_\nu(x)$, constants ϱ_ν, and c_ν for all $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, where $c_\nu = 0$.

$\mathfrak{S}_{(q,q',\Delta(q'))}$	$u_\nu(x) = a_0x_1^2 + a_1x_1x_2 + a_2x_1 + a_3x^2 + a_4x_2 + a_5$ $[a_0, a_1, a_2, a_3, a_4, a_5]$	ϱ_ν
$\{(q_0,q_1,q_2,3),(q_0,q_1,q_4,3)\}$	[1.745e-3,3.664e-6,1.884e-4,1.938e-3,3.886e-4,0.161]	4.883e-4
$\{q_1,q_2,q_3,3\}$	[1.321e-3,3.252e-5,2.544e-4,1.828e-3,4.212e-3,0.228]	0.002
$\{q_1,q_4,q_3,3\}$	[1.754e-3,−6.636e-6,1.636e-4,1.934e-3,−2.170e-3,0.163]	9.766e-4
$\{q_0,q_4,q_3,4\}$	[1.754e-3,−6.636e-6,1.636e-4,1.934e-3,−2.170e-3,0.163]	9.766e-4

Remark 5.7.4. *To minimize the values of ϱ_ν and c_ν for each $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, one can simply utilize the bisection procedure by iteratively fixing ϱ_ν and minimizing over c_ν and then fixing the obtained c_ν and minimizing over ϱ_ν. In this way, we give priority to minimizing c_ν to obtain a tight upper bound $(\varrho_\nu + c_\nu T_d)$ which is less sensitive to the finite time horizon T_d.*

Remark 5.7.5. *The procedure discussed above may result in a more conservative probability bounds due to the computation of common control barrier function in some cases. To obtain less conservative bounds one can simply substitute the constructed control policy in the dynamics of the system and recompute control barrier functions minimizing constants ϱ_ν and c_ν for each $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$ using Lemma 5.7.2. Then utilize these values to compute $\underline{\vartheta}$ in Problem 5.3.8 using Theorem 5.6.3 and Corollary 5.6.5.*

Example 5.7.6. *(Example 5.3.9 continued) To compute control policy $u_\nu(x)$ and values of ϱ_ν and c_ν for each $\nu \in \mathcal{P}(\mathcal{A}_{\neg\varphi})$, we use SOS optimization according to Lemma 5.7.2 and minimize values of ϱ and c using bisection method. The optimization problem is solved using SOSTOOLS and SeDuMi. We choose control barrier functions $\mathcal{B}$, SOS polynomials $\Upsilon_0, \Upsilon_1, \Upsilon$, and polynomial controller Υ_u of orders 4, 2, 2, 2 and 2, respectively. The obtained controllers $u_\nu(x)$ and values of ϱ_ν and c_ν are listed in Table 5.1. Now using Theorem 5.6.3, one gets*

$\mathbb{P}_\upsilon^{x_0}\{L(\boldsymbol{x}_N) \models \neg\varphi\} \leq 4.883e\text{-}4\times 2e\text{-}3 + 4.883e\text{-}4\times 9.766e\text{-}4 = 1.453e\text{-}6$, *for all* $x_0 \in L^{-1}(p_0)$;
$\mathbb{P}_\upsilon^{x_0}\{L(\boldsymbol{x}_N) \models \neg\varphi\} \leq 9.766e\text{-}4$, *for all* $x_0 \in L^{-1}(p_2)$; *and*
$\mathbb{P}_\upsilon^{x_0}\{L(\boldsymbol{x}_N) \models \neg\varphi\} = 1$, *for all* $x_0 \in L^{-1}(p_1) \cup L^{-1}(p_3)$.

The control policy is given by $\upsilon(x, q_\mathfrak{m}) = \mathfrak{u}_{\mathfrak{S}_{(q'_\mathfrak{m})}}(x)$, where $(q_\mathfrak{m}, L(x), q'_\mathfrak{m}) \in \mathfrak{d}_M$ is a transition in DFA $\mathcal{A}_\mathfrak{m}$ as shown in Figure 5.3. □

5.7.2 Finite Input Sets

We use a counter-example guided inductive synthesis (CEGIS) framework to find control barrier functions for the system Σ_s^{dt} with a finite input set U. The approach uses satisfiability (feasibility) solvers for finding control barrier function of a given parametric form that handles quantified formulas by alternating between series of quantifier-free formulas using existing satisfiability modulo theories (SMT) solvers (*viz.*, Z3 [dMB08],

dReal [GKC13], and OptiMathSAT [ST15]). In order to use CEGIS framework, we raise the following assumption.

Assumption 5.7.7. *System Σ_s^{dt} has a compact state set $X \subset \mathbb{R}^n$ and a finite input set $U = \{u_1, u_2, \ldots, u_l\}$, where $u_i \in \mathbb{R}^m$, $i \in [1; l]$. Partition sets $X_i = L^{-1}(p_i)$, $i \in [0; M]$, are bounded semi-algebraic sets.*

Under Assumption 5.7.7, we can formulate conditions of Theorem 5.4.6 as a satisfiability problem which can search for control barrier functions using CEGIS approach. The following Lemma gives a feasibility condition that is equivalent to conditions of Theorem 5.4.6.

Lemma 5.7.8. *Suppose Assumption 5.7.7 holds and X_0, X_1, X are bounded semi algebraic sets. Suppose there exists a function $\mathcal{B}(x)$, constants $\varrho \in [0, 1]$, and $c \geq 0$, such that following expression is true*

$$\bigwedge_{x \in X} \mathcal{B}(x) \geq 0 \bigwedge_{x \in X_0} \mathcal{B}(x) \leq \varrho \bigwedge_{x \in X_1} \mathcal{B}(x) \geq 1 \bigwedge_{x \in X} \Big(\bigvee_{u \in U} (\mathbb{E}[\mathcal{B}(f_s(x, u, w)) \mid x, u] \leq \mathcal{B}(x) + c) \Big). \tag{5.7.4}$$

Then, $\mathcal{B}(x)$ satisfies conditions of Theorem 5.4.6 and any u in $\{u_i \in U \mid \mathbb{E}[\mathcal{B}(f_s(x, u_i)) \mid x, u_i] \leq \mathcal{B}(x) + c\}$ gives a corresponding control input.

Now, we briefly explain the idea of CEGIS framework for computation of such a function $\mathcal{B}(x)$.

1. Define a parameterized control barrier function of the form $\mathcal{B}(p, x) = \sum_{i=1}^{\mathsf{r}} p_i b_i(x)$ with some user-defined (possibly nonlinear) basis functions $b_i(x)$ and unknown coefficients $p_i \in \mathbb{R}$, $i \in [1; \mathsf{r}]$.

2. Select a finite set of samples $\overline{X} \subset X$, a constant $\varrho \in [0, 1]$, and $c \geq 0$.

3. Compute a candidate control barrier function $\mathcal{B}(p, x)$ (*i*.e., coefficients p_i) such that the following expression is true.

$$\psi(p, x) := \bigwedge_{x \in \overline{X}} \mathcal{B}(p, x) \geq 0 \bigwedge_{x \in \overline{X} \cap X_0} \mathcal{B}(p, x) \leq \varrho \bigwedge_{x \in \overline{X} \cap X_1} \mathcal{B}(p, x) \geq 1$$
$$\bigwedge_{x \in \overline{X}} \Big(\bigvee_{u \in U} (\mathbb{E}[\mathcal{B}(p, f_s(x, u, w)) \mid x, u] \leq \mathcal{B}(p, x) + c) \Big).$$

   The above expression results in linear arithmetic formula that involves boolean combinations of linear inequality constraints in p_i, which can be efficiently solved with the help of SMT solvers Z3 [dMB08] or OptiMathSAT [ST15].

4. Search for a counter example $x_c \in X$ such that the candidate solution $\mathcal{B}(p, x)$ obtained in the previous step satisfies $\neg\psi(p, x)$. Note that for a given p, satisfaction of $\neg\psi(p, x)$ is equivalent to the feasibility of a nonlinear constraint over x. If

$\neg\psi(p,x)$ has no feasible solution, the obtained candidate solution is a true control barrier function for all $x \in X$ which terminates the algorithm. Otherwise, if $\neg\psi(p,x)$ is feasible for some $x = x_c \in X$, then we add that counter-example x_c to the finite set, $\overline{X} := \overline{X} \cup \{x_c\}$, and reiterate Steps 3–4.

There are two possible ways to search for counter-examples:

(a) *Using SMT solvers*: To check satisfiability of $\neg\psi(p,x)$, one can use an SMT solver that can handle nonlinear constraints. For example, dReal [GKC13] is a general purpose nonlinear delta-satisfiability solver suitable for solving quantifier-free nonlinear constraints involving polynomials, trigonometric, and rational functions over compact sets X. We refer the interested readers to [RS17] for a more detailed discussion.

(b) *Using nonlinear optimization toolboxes*: To find counter-examples, one can alternatively solve a nonlinear optimization problem and check satisfaction of the following condition

$$\text{If } \Big(\min_{x\in X} \mathcal{B}(p,x) < 0, \text{ OR } \min_{x\in X_0} -\mathcal{B}(p,x) + \varrho < 0, \text{ OR } \min_{x\in X_1} \mathcal{B}(p,x) - 1 < 0,$$
$$\text{OR } \min_{x\in X} \max_{u\in U} -\mathbb{E}[\mathcal{B}(p, f_s(x,u,w)) \mid x,u] + \mathcal{B}(p,x) + c < 0 \Big)$$

then x is a counter-example.

To solve nonlinear optimization problems, one can use existing numerical optimization techniques such as sequential quadratic programming. Note that, the methods may run into local optima, however, one can utilize multi-start techniques [Mar03] to obtain global optima. For the final rigorous verification step, one can use tools like RSolver[1] which extends a basic interval branch-and-bound method with interval constraint propagation. A detailed discussion on the verification algorithm used in RSolver can be found in [Rat06, Rat17].

This CEGIS algorithm is then iterated to minimize the values of ϱ and c in (5.7.4) as discussed in Remark 5.7.4. Note that, the CEGIS procedure either (i) terminates after some finite iterations with a control barrier function satisfying (5.7.4), (ii) terminates with a counter example proving that no solution exists, or (iii) runs forever. In order to guarantee termination of the algorithm, one can set an upper bound on the number of unsuccessful iterations.

5.7.3 Computational Complexity

Characterizing the computational complexity of the proposed approaches is a very difficult task in general. However, in this subsection, we provide some analysis on the computational complexity.

[1] http://rsolver.sourceforge.net

From the construction of directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, explained in Section 5.5, the number of triplets and hence the number of control barrier functions needed to be computed are bounded by $|\mathcal{V}|^3 = |Q|^3$, where $|\mathcal{V}|$ is the number of vertices in $\mathcal{G}$. However, this is the worst-case bound. In practice, the number of control barrier functions is much less. In particular, it is given by the number of all unique successive pairs of atomic propositions corresponding to the elements $\nu \in P(\mathcal{A}_{\neg\varphi})$. Further, it is known that $|Q|$ is at most $|\neg\varphi|2^{|\neg\varphi|}$, where $|\neg\varphi|$ is the length of formula $\neg\varphi$ in terms of number of operations [BKL08], but in practice, it is much smaller than this bound [KB06].

In the case of sum-of-squares optimization, the computational complexity of finding polynomials $B, \Upsilon_0, \Upsilon_1, \Upsilon_{u_i}$, and Υ in Lemma 5.7.2 depends on both the degree of polynomials appearing in (5.7.1)-(5.7.2) and the number of state variables. It is shown that for fixed degrees, the required computations grow polynomially with respect to the dimension [WTL15]. Hence, we expect that this technique is more scalable in comparison with the discretization-based approaches, especially for large-dimensional systems. For the CEGIS approach, due to its iterative nature and lack of guarantee on termination, it is difficult to provide any analysis on the computational complexity.

5.8 Case Studies

In this section, we consider two case studies to demonstrate the effectiveness of our results.

5.8.1 Temperature Control of a Room

For the purpose of illustration of the proposed results, we use the discretized dynamics of a room temperature control and is given by stochastic difference equation

$$\mathbf{x}(k+1) = \mathbf{x}(k) + \tau_s(\alpha_e(T_e - \mathbf{x}(k)) + \alpha_H(T_h - \mathbf{x}(k))\upsilon(k)) + 0.1\omega(k), \tag{5.8.1}$$

where $\mathbf{x}(k)$ denotes the temperature of the room, $\upsilon(k)$ represents ratio of the heater valve being open, $\omega(k)$ is a standard normal random variable that models environmental uncertainties, $\tau_s = 5$ minutes is the sampling time, $T_h = 55°C$ is the heater temperature, $T_e = 15°C$ is the ambient temperature, and $\alpha_e = 8 \times 10^{-3}$ and $\alpha_H = 3.6 \times 10^{-3}$ are heat exchange coefficients. All the coefficients are taken from Section 3.5.

The state set of the system is $X \subseteq \mathbb{R}$. We consider regions of interest $X_0 = [21, 22]$, $X_1 = [0, 20]$, $X_2 = [23, 45]$, and $X_3 = X \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_i) = p_i$ for all $x_i \in X_i$, $i \in \{0, 1, 2, 3\}$. The objective is to compute a control policy with a potentially tight lower bound on the probability that the state evolution of length $N = 50$ satisfies the LTL_F formula $\varphi = p_0 \wedge \Box\neg(p_1 \vee p_2)$. The DFA $\mathcal{A}_{\neg\varphi}$ corresponding to $\neg\varphi$ is shown in Figure 5.4. One can readily see that, we have sets $\mathcal{P}^{p_0} = \{(q_0, q_1, q_2, 49)\}$ and $\mathcal{P}^{p_1} = \mathcal{P}^{p_2} = \mathcal{P}^{p_3} = \emptyset$. Next, we discuss the computational results for two cases of finite and continuous input sets.

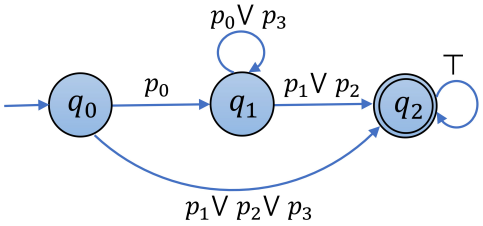

Figure 5.4: DFA $\mathcal{A}_{\neg\varphi}$ that accept all traces of $\neg\varphi$, where $\varphi = p_0 \wedge \Box\neg(p_1 \vee p_2)$.

Finite input set.

We consider that the control input $\upsilon(k)$ takes value in the set $U = \{0, 0.5, 1\}$ (the heater valve is either closed, half open, or full open) and the temperature lies in the bounded set $X = [0, 45]$. We compute a control barrier function of order 4 using the CEGIS approach discussed in Subsection 5.7.2 as the following:

$$\mathcal{B}(x) = 0.2167x^4 - 18.6242x^3 + 6.0032e2x^2 - 8.5998e3x + 4.6196e4.$$

The corresponding controller is

$$\mathsf{u}(x) = \min\{u_i \in U \mid \mathbb{E}[\mathcal{B}(f_s(x, u_i)) \mid x, u_i] \leq \mathcal{B}(x) + c\}. \tag{5.8.2}$$

One can readily see that the DFA of switching mechanism $\mathcal{A}_{\mathfrak{m}}$ contains only three states $Q_{\mathfrak{m}} = \{(q_0, \Delta(q_0)), (q_0, q_1, \Delta(q_1)), q_2\}$, thus we have control policy $\upsilon(x, q_{\mathfrak{m}}) \equiv u(x)$. The lower bound $\mathbb{P}^{x_0}_{\upsilon}\{L(\mathbf{x}_N) \models \varphi\} \geq 0.9766$ for all $x_0 \in L^{-1}(p_0)$ is obtained using SMT solver Z3 and employing sequential quadratic programming for computing counterexamples as described in Subsection 5.7.2. Values of ϱ and c are obtained as 0.008313 and 0.0003125, respectively. The implementation performed using Z3 SMT solver along with sequential quadratic program in Python on an iMac (3.5 GHz Intel Core i7 processor) and it took around 4 minutes to find a control barrier function and the associated lower bound. Figure 5.5 depicts the control barrier function and the corresponding conditions in Theorem 5.4.6: condition (5.4.4) is shown in a snippet in the top figure, condition (5.4.5) is shown in the top figure, and condition (5.4.2) for the control barrier function with control input $u(x)$ is shown in the bottom figure. Figure 5.6 presents the control policy in (5.8.2) and Figure 5.7 shows a few realizations of the temperature under this policy.

Continuous input set.

Let us assume the system has the state space $X = \mathbb{R}$ and the continuous input set $U = [0, 1]$ (the heater valve can be positioned continuously from fully closed to fully open). As described in Subsection 5.7.1, using Lemma 5.7.2 we compute a control barrier function of order 4 as follows

$$\mathcal{B}(x) = 0.1911x_1^4 - 16.4779x_1^3 + 532.6393x_1^2 - 7651.3308x_1 + 41212.3666,$$

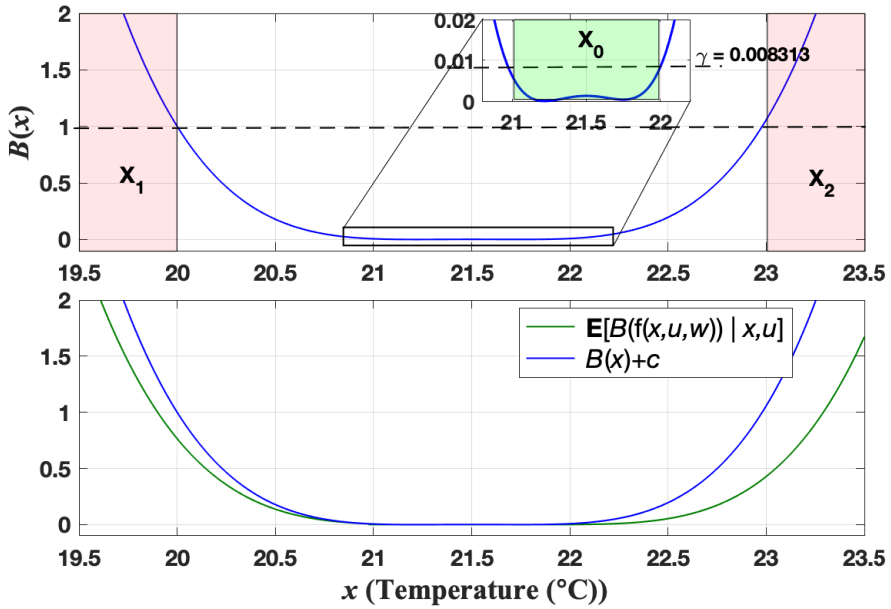

Figure 5.5: Room temperature control: control barrier function and the associated conditions from Theorem 5.4.6. Condition (5.4.4) is shown in the snippet in the top figure, condition (5.4.5) is shown in the top figure, and condition (5.4.2) for the control barrier function under policy u is shown in the bottom figure.

and the corresponding control policy of order 4 as

$$\mathsf{u}(x) = -1.018e\text{-}6x^4 + 7.563e\text{-}5x^3 - 0.001872x^2 + 0.02022x + 0.3944. \tag{5.8.3}$$

The values $\varrho = 0.015625$, $c = 0.00125$, and the lower bound $\mathbb{P}_v^{x_0}\{L(\mathbf{x}_N) \models \varphi\} \geq 0.9281$ is obtained using SOSTOOLS and SeDuMi for all $x_0 \in L^{-1}(p_0)$, as discussed in Subsection 5.7.1. The bound in this case is more conservative that the previous case with a finite input set. This is mainly due to the optimization algorithm that assumes fixed-degree polynomials $\mathcal{B}(\cdot)$, $\Upsilon_0(\cdot)$, $\Upsilon_1(\cdot)$, $\Upsilon(\cdot)$, and $\Upsilon_u(\cdot)$. The computed lower bound can be improved by increasing the degrees but will result in a larger computational cost. The control policy and a few realizations of the temperature under this policy are shown in figures 5.8 and 5.9, respectively.

Discretization-based approaches provide a policy that is generally time-dependent. So it is not possible to directly compare our approach with them but using these techniques, we can validate the lower bound provided by our approach a posteriori. For this purpose, we combine our synthesized policy with the system to obtain an autonomous system and then use the toolbox FAUST^2 [SGA15] that computes an interval for the probability based on finite abstractions of the system. The toolbox takes around 4 minutes to verify the system using 314 abstract states. The probability satisfies

$$\mathbb{P}_v^{x_0}\{L(\mathbf{x}_N) \models \varphi\} \in [1 - 5.458 \times 10^{-4}, 1 - 3.612 \times 10^{-4}] \quad \text{for all } x_0 \in L^{-1}(p_0),$$

which confirms the lower bound provided by our approach.

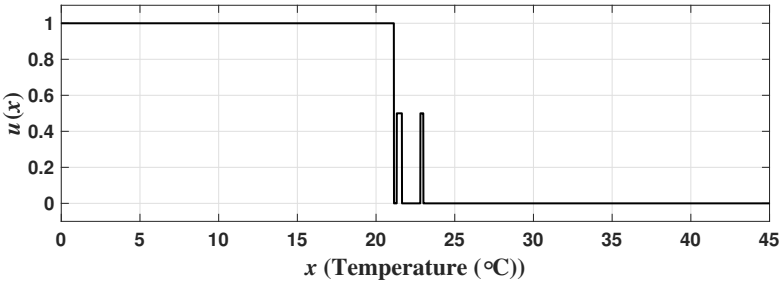

Figure 5.6: Room temperature control: controller $\mathsf{u} : X \to \{0, 0.5, 1\}$ as given in (5.8.2).

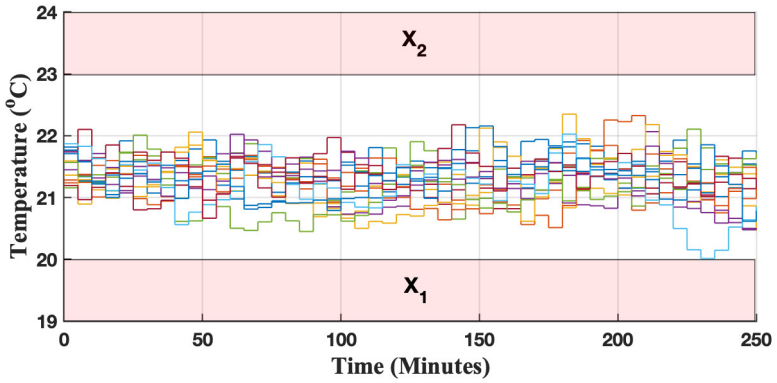

Figure 5.7: Room temperature control: temperature evolution under control policy in (5.8.2).

5.8.2 Lane Keeping of a Vehicle

For the second case study, we consider a kinematic single-track model of a vehicle, specifically, BMW 320i, adopted from [AKM17] by discretizing the model and adding noises to capture the effect of uneven road. The corresponding nonlinear stochastic difference equation is

$$\begin{aligned}
\mathbf{x}_1(k+1) &= \mathbf{x}_1(k) + \tau_s v \cos(\mathbf{x}_4(k)) + 0.1\omega_1(k) \\
\mathbf{x}_2(k+1) &= \mathbf{x}_2(k) + \tau_s v \sin(\mathbf{x}_4(k)) + 0.01\omega_2(k) \\
\mathbf{x}_3(k+1) &= \mathbf{x}_3(k) + \tau_s \upsilon(k) \\
\mathbf{x}_4(k+1) &= \mathbf{x}_4(k) + \frac{\tau_s v}{l_{wb}} \tan(\mathbf{x}_3(k)) + 0.0005\omega_3(k),
\end{aligned}$$

where states $\mathbf{x}_1(k), \mathbf{x}_2(k), \mathbf{x}_3(k)$, and $\mathbf{x}_4(k)$ represent x-position, y-position, the steering angle, and the heading angle, respectively. The schematic showing states in the single-track model is shown in Figure 5.10. The control input representing steering velocity is denoted by $\upsilon(k)$. The terms $\omega_1(k)$, $\omega_2(k)$, and $\omega_3(k)$ are noises in position and heading generated due to uneven road modelled using standard normal distribution at kth instance. The parameters $\tau_s = 0.01s$, $l_{wb} = 2.578m$, and $v = 10m/s$ represent the sampling time, the wheelbase, and velocity, respectively. We consider the state set

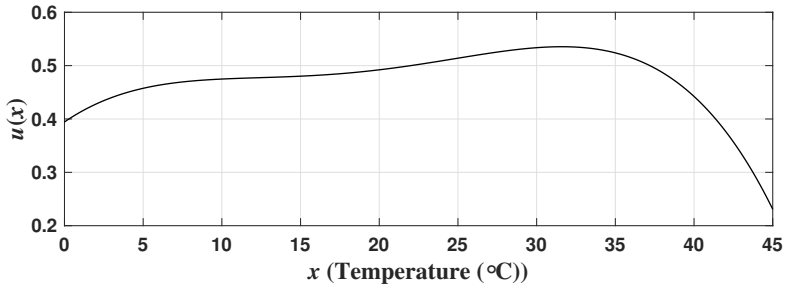

Figure 5.8: Room temperature control: controller $\mathsf{u} : X \to [0, 1]$ as given in (5.8.3).

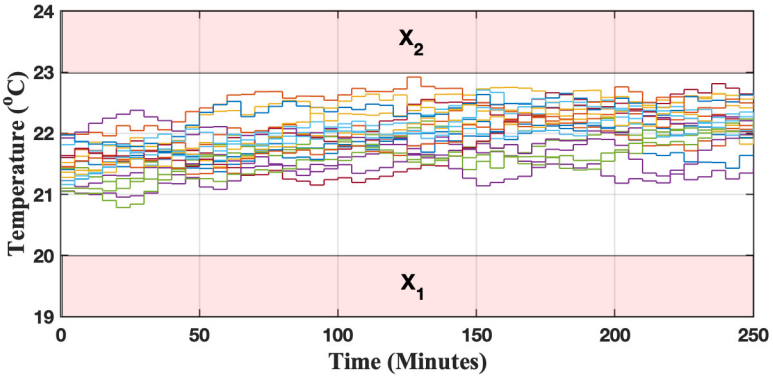

Figure 5.9: Room temperature control: temperature evolution under control policy in (5.8.3).

$X = [0, 50] \times [-6, 6] \times [-0.05, 0.05] \times [-0.1, 0.1]$, finite input set $U = \{-0.5, 0, 0.5\}$, regions of interest $X_0 = [0, 5] \times [-0.1, 0.1] \times [-0.005, 0.005] \times [-0.05, 0.05]$, $X_1 = [0, 50] \times [-6, -2] \times [-0.05, 0.05] \times [-0.1, 0.1]$, $X_1 = [0, 50] \times [2, 6] \times [-0.05, 0.05] \times [-0.1, 0.1]$, and $X_3 = X \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_i) = p_i$ for all $x_i \in X_i$, $i \in \{0, 1, 2, 3\}$. Our goal is to design a control policy to keep the vehicle in the middle lane for the time horizon of 4 seconds (*i*.e., $N = 400$). The specification can be written as an LTL_F formula $\varphi = p_0 \wedge \Box\neg(p_1 \vee p_2)$. Using CEGIS approach discussed in Subsection 5.7.2, we compute a control barrier function as the following:

$$\begin{aligned}\mathcal{B}(x) =& 2.1794e\text{-}6x_1^2 + 6.2500e\text{-}2x_2^2 - 15.3131x_3^2 + 1.0363x_4^2 + 1.3088e\text{-}4x_1 \\ & - 4.4330e\text{-}5x_2 + 0.3592x_3 - 0.2488x_4 + 5.9126e\text{-}2,\end{aligned}$$

and the corresponding control policy as

$$\mathsf{u}(x) \in \{u_i \in U \mid \mathbb{E}[\mathcal{B}(f_s(x, u_i)) \mid x, u_i] \leq \mathcal{B}(x) + c\}, \tag{5.8.4}$$

which guarantees $\mathbb{P}_\rho^{x_0}\{L(\mathbf{x}_N) \models \varphi\} \geq 0.8688$ with values $\varrho = 0.03125$ and $c = 0.00025$. Figure 5.11 shows a few realizations of the system under the control policy (5.8.4). The implementation performed using the Z3 SMT solver along with the sequential quadratic

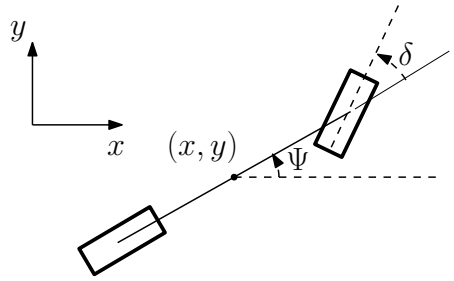

Figure 5.10: Single-track model.

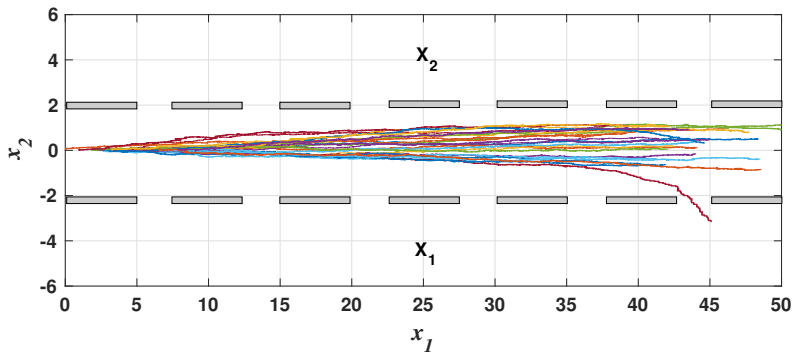

Figure 5.11: Several closed-loop realization using controller in (5.8.4).

program in Python on an iMac (3.5 GHz Intel Core i7 processor) and it took around 30 hours to find a control barrier certificate and the associated lower bound. Note that, since the procedure described in Subsection 5.7.2 is highly parallelizable, the execution time can be reduced significantly. Note that due to the large dimension of the state set, FAUST^2 is not able to give a lower bound on the probability of satisfaction. However, for the sake comparison, we employ the Monte-Carlo approach to obtain the empirical probability interval as $\mathbb{P}_{\rho}^{x_0}\{L(\mathbf{x}_N) \models \varphi\} \in [0.9202, 0.9630]$ with the confidence $1 - 10^{-10}$ using 10^5 realizations with the controller in (5.8.4), which confirms the lower bound obtained using our approach.

6 Compositional Controller Synthesis for Interconnected Control Systems

This chapter provides a compositional framework for synthesizing hybrid controllers for interconnected discrete-time control systems enforcing specifications expressed by co-Büchi automata. To synthesize such hybrid controllers, we utilize a notion of control barrier functions. We show that such control barrier functions can be constructed compositionally by assuming some small-gain type conditions and composing so-called local control barrier functions computed for subsystems.

6.1 Introduction

The existing techniques for the synthesis of controllers enforcing complex specifications (i.e, discrete abstraction based techniques and control barrier function based techniques) become very intractable for large-scale interconnected systems. In abstraction based techniques, the computational complexity for computing abstractions and controllers grows exponentially with the dimension of state and input sets of concrete systems. In barrier function based techniques, the computational complexity of searching for control barrier functions grows in polynomial time [JSZ18, WTL15] with respect to the dimension of the system. To alleviate this issue of dimensionality, one can leverage compositional approaches. Some of the results in that direction are listed below.

6.1.1 Related Literature

Compositional abstraction-based approaches

To address the aforementioned scalability issue, several results were proposed by utilizing the compositional abstraction-based synthesis where the synthesis is performed by computing the abstractions and (possibly) controllers for smaller subsystems; see the results in [MGW17, PPD16, SZ19b, SZ19c, SGZ18, SJZG21, SJZG18, and references therein] for more details.

Control barrier functions

Alternatively, a discretization-free approach, based on control barrier functions, has shown the potential to solve the formal synthesis problems as well. Assuming a prior

knowledge of control barrier functions, several techniques have been recently introduced to ensure the safety of dynamical systems (see [AXGT17, ACE+19, and the references therein]), or the satisfaction of a set of signal temporal logic tasks for multi-agent systems [LD19c, LD19a]. The results in [JSZ20a] provide techniques to search for control barrier functions to synthesize controllers for stochastic control systems enforcing a class of temporal logic specifications over finite time horizons. Though promising, the computational complexity of searching for control barrier functions grows in polynomial time [JSZ18, WTL15] with respect to the dimension of the system and, hence, the existing approaches [AXGT17, ACE+19, JSZ20a] will also become computationally intractable while dealing with large-scale interconnected systems.

6.1.2 Contributions

Motivated by the above results and their limitations, this chapter proposes a controller synthesis approach for large-scale systems against complex logic specifications via the compositional construction of control barrier functions. To the best of our knowledge, this work is the first to utilize the compositional construction of control barrier functions for synthesizing hybrid controllers for interconnected discrete-time control systems against specifications expressed by co-Büchi automata. In order to achieve this, we first decompose the given specification to simpler reachability tasks based on automata representing the complements of original co-Büchi automata. Then, we provide a systematic approach to solve those simpler tasks by computing corresponding control barrier functions. Those control barrier functions are obtained by composing so-called local control barrier functions while utilizing some small-gain type conditions. In the final step, we combine those control barrier functions and controllers obtained for simpler tasks to obtain hybrid controllers ensuring the desired complex specifications over large-scale interconnected systems. In addition, we provide two systematic approaches to search for local control barrier functions under suitable assumptions on the dynamics of the subsystems. The first approach is using the sum-of-squares optimization [Par03] and the second one is utilizing a counter-example guided inductive synthesis approach [RS17].

Finally, we demonstrate the effectiveness of the proposed results on two large-scale case studies with 10^4 state dimensions. First, we apply our results to the temperature regulation in a circular building by synthesizing controllers for a network containing N rooms for any $N \geq 3$ ensuring the satisfaction of a specification given by a deterministic co-Büchi automaton. Additionally, we also apply the proposed techniques to a nonlinear example of a fully connected network of Kuramoto oscillators and synthesize hybrid controllers ensuring the satisfaction of a given specification.

6.2 Interconnected Control Systems

Special notations used in this chapter

Given $N \in \mathbb{N}$, vectors $x_i \in \mathbb{R}^{n_i}$, $n_i \in \mathbb{N}$, and $i \in [1; N]$, we use $x = [x_1; \ldots; x_N]$ to denote the vector in $\mathbb{R}^n$ with $n = \sum_i n_i$ consisting of the concatenation of vectors x_i.

First, we define discrete-time control subsystems which will be later interconnected to form a large-scale discrete-time control system.

Definition 6.2.1. *A control subsystem Σ_i is a tuple*

$$\Sigma_i = (X_i, U_i, W_i, f_i, Y_i, h_i), \quad i \in [1; N], \tag{6.2.1}$$

where X_i, U_i, W_i, and Y_i are the state set, the external input set, the internal input set, and the output set, respectively.

The function $f_i : X_i \times U_i \times W_i \to X_i$ is the transition function and $h_i : X_i \to Y_i$ is the output function. The discrete-time control subsystem Σ_i is described by difference equations of the form

$$\Sigma_i : \begin{cases} \mathbf{x}_i(k+1) = f_i(\mathbf{x}_i(k), \upsilon_i(k), \omega_i(k)), \\ \mathbf{y}_i(k) = h_i(\mathbf{x}_i(k)), \end{cases} \tag{6.2.2}$$

where $\mathbf{x}_i : \mathbb{N}_0 \to X_i$, $\mathbf{y}_i : \mathbb{N}_0 \to Y_i$, $\upsilon_i : \mathbb{N}_0 \to U_i$, and $\omega_i : \mathbb{N}_0 \to W_i$ are the state run, output run, external input run, and internal input run, respectively.

Now, we provide a formal definition of interconnected discrete-time control systems.

Definition 6.2.2. *Consider $N \in \mathbb{N}$ control subsystems $\Sigma_i = (X_i, U_i, W_i, f_i, Y_i, h_i)$ with their inputs and outputs partitioned as*

$$w_i = [w_{i1}; \ldots; w_{i(i-1)}; w_{i(i+1)}; \ldots; w_{iN}], W_i = \prod_{j=1, j\neq i}^{N} W_{ij},$$

$$y_i = [y_{i1}; \ldots; y_{iN}], Y_i = \prod_{j=1}^{N} Y_{ij},$$

with $w_{ij} \in W_{ij}$, $y_{ij} = h_{ij}(x_i)$ and output function

$$h_i(x_i) = [h_{i1}(x_i); \ldots; h_{iN}(x_i)] \quad \text{with} \quad h_{ii}(x_i) = x_i.$$

The interconnected control system $\Sigma_{\mathcal{I}} = \mathcal{I}(\Sigma_1, \ldots, \Sigma_N)$ is a tuple

$$\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}}), \tag{6.2.3}$$

described by the difference equation

$$\mathbf{x}(k+1) = f_{\mathcal{I}}(\mathbf{x}(k), \upsilon(k)), \tag{6.2.4}$$

where $X = \prod_{i=1}^{N} X_i$, $U = \prod_{i=1}^{N} U_i$, and function

$$f_{\mathcal{I}}(x, u) = [f_1(x_1, u_1, w_1); \ldots; f_N(x_N, u_N, w_N)],$$

where $x = [x_1; \ldots; x_N] \in X$, $u = [u_1; \ldots; u_N] \in U$, and the interconnection variables are constrained by $w_{ij} = y_{ji}$, $Y_{ji} \subseteq W_{ij}$, $\forall i, j \in [1; N], i \neq j$. Moreover, let $\mathbf{x}_{x,\upsilon}$ denote a state run of $\Sigma_{\mathcal{I}}$ starting from initial state $x \in X$ under input run $\upsilon : \mathbb{N} \to U$. An example of the interconnection of three control subsystems Σ_1, Σ_2, and Σ_3 is illustrated in Figure 6.1.

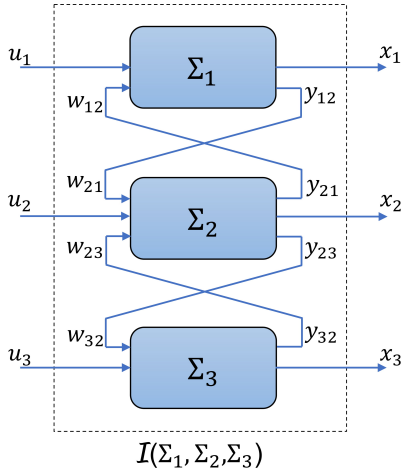

Figure 6.1: Interconnection of three control subsystems Σ_1, Σ_2, and Σ_3 with h_{13} and h_{31} being zero maps.

In the above definition, we assumed that one has access to the full state information of subsystems (i.e. $h_{ii}(x_i) = x_i$) for the sake of controller synthesis. However, for the sake of internal interconnections, we work with the outputs of states (i.e. $h_{ij}, i, j \in [1; N], i \neq j$) (cf. Figure 6.1).

We are interested in synthesizing control policies υ for system $\Sigma_{\mathcal{I}}$ enforcing given complex specifications. Here, we consider *history-dependent policies* given by $\upsilon = (\upsilon_0, \upsilon_1, \ldots, \upsilon_k, \ldots)$ with functions $\upsilon_k : \mathcal{H}_k \to U$, where $\mathcal{H}_k$ is the set of all k-histories $\mathbf{h}_k$ defined as $\mathbf{h}_k := (\mathbf{x}(0), \upsilon(0), \mathbf{x}(1), \upsilon(1), \ldots, \mathbf{x}(k-1), \upsilon(k-1), \mathbf{x}(k))$. A subclass of those policies are called *stationary* and are defined as $\mathsf{u} = (\mathsf{u}, \mathsf{u}, \ldots, \mathsf{u}, \ldots)$ with a function $\mathsf{u} : X \to U$. In stationary policies, the mapping at time k depends only on the current state $\mathbf{x}(k)$ and does not change over time.

6.3 Preliminaries

6.3.1 Class of Specifications

Here, we consider the class of specifications expressed by deterministic co-Büchi automata (DCA) [Löd01] as defined next.

Definition 6.3.1. *A deterministic co-Büchi automaton (DCA) is a tuple* $\mathcal{A} = (Q, Q_0, \Sigma_{\mathcal{A}}, \mathfrak{d}, F)$*, where* Q *is a finite set of states,* $Q_0 \subseteq Q$ *is a set of initial states,* $\Sigma_{\mathcal{A}}$ *is a finite set of alphabet,* $\mathfrak{d} : Q \times \Sigma_{\mathcal{A}} \to Q$ *is a transition function, and* $F \subseteq Q$ *is a set of final states.*

We use notation $q \xrightarrow{\sigma} q'$ to denote transition $(q, \sigma, q') \in \mathfrak{d}$. We also denote the set of all successor states of a state $q \in Q$ by $\Delta(q)$. Consider an infinite state run $\mathbf{q} = (q_0, q_1, \ldots) \in Q^\omega$ such that $q_0 \in Q_0$, $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $i \geq 0$ and let $\mathsf{Inf}(\mathbf{q})$ be the set of states that occurs infinitely many times in $\mathbf{q}$. An infinite word (a.k.a trace)

$\sigma = (\sigma_0, \sigma_1, \ldots) \in \Sigma_{\mathcal{A}}^{\omega}$ is accepted by DCA $\mathcal{A}$ if there exists an infinite state run $\mathbf{q}$ such that $\mathsf{Inf}(\mathbf{q}) \cap F = \emptyset$. The set of words accepted by $\mathcal{A}$ is called the accepting language of $\mathcal{A}$ and is denoted by $\mathcal{L}(\mathcal{A})$.

A deterministic Büchi automaton (DBA) is defined syntactically exactly as a deterministic co-Büchi automaton except that its accepting runs are those for which $\mathsf{Inf}(\mathbf{q}) \cap F \neq \emptyset$. Note that the complement of a deterministic co-Büchi automaton is a deterministic Büchi automaton [Löd01].

In this chapter, we consider those specifications given by the accepting languages of DCA $\mathcal{A}$ defined over the set of atomic propositions Π, i.e., the alphabet[1] $\Sigma_{\mathcal{A}} = \Pi$. We should highlight that the temporal logic specifications represented using *obligation* properties [MP12] (including boolean combinations of safety and guarantee properties) are all recognized by deterministic weak automata [DEK07] which are included in DCA. For other temporal logic formulae, one can readily check the existence of DCA using the tool `SPOT` [DLLF+16].

6.3.2 Satisfaction of Specifications by Interconnected Control Systems

In this subsection, we define how the specification given by the accepting language of DCA $\mathcal{A}$ is satisfied by the system $\Sigma_{\mathcal{I}}$ as in Definition 6.2.2.

Definition 6.3.2. *Consider an interconnected control system $\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}})$ as in Definition 6.2.2 and a specification expressed by DCA $\mathcal{A} = (Q, Q_0, \Pi, \mathfrak{d}, F)$. In order to reason about the given specification for the system $\Sigma_{\mathcal{I}}$, we use a measurable labeling function $L : X \to \Pi$. In addition, consider an infinite state run $\mathbf{x} = (\mathbf{x}(0), \mathbf{x}(1), \ldots) \in X^{\omega}$, and labeling function $L : X \to \Pi$. Then, the corresponding trace over Π is given by $L(\mathbf{x}) := (\sigma_0, \sigma_1, \ldots) \in \Pi^{\omega}$, where $\sigma_k = L(\mathbf{x}(k))$ for all $k \in \{0, 1, \ldots\}$.*

Note that we abuse the notation by using map $L(\cdot)$ over X^{ω}, i.e., $L(\mathbf{x}(0), \mathbf{x}(1), \ldots) \equiv (L(\mathbf{x}(0)), L(\mathbf{x}(1)), \ldots)$. Their distinction is clear from the context. Next we define the satisfaction of specifications by the control systems $\Sigma_{\mathcal{I}}$.

Definition 6.3.3. *Consider an interconnected control system $\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}})$ as in Definition 6.2.2, a specification given by the accepting language of DCA $\mathcal{A} = (Q, Q_0, \Pi, \mathfrak{d}, F)$, and a labeling function $L : X \to \Pi$. We say that the state run of $\Sigma_{\mathcal{I}}$ starting from initial state $x \in X$ under control policy υ satisfies the specification given by $\mathcal{A}$, denoted by $L(\mathbf{x}_{x,\upsilon}) \models \mathcal{A}$, if $L(\mathbf{x}_{x,\upsilon}) \in \mathcal{L}(\mathcal{A})$.*

6.3.3 Problem Definition

The main synthesis problem in this chapter is formally defined next.

Problem 6.3.4. *Given an interconnected control system $\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}})$ as in Definition 6.2.2, a specification given by the accepting language of DCA $\mathcal{A} = (Q, Q_0, \Pi, \mathfrak{d}, F)$ over*

[1] For properties expressed by DCA $\mathcal{A}$ over atomic propositions Π, $\mathcal{A}$ is usually constructed over the alphabet $\Sigma_{\mathcal{A}} = 2^{\Pi}$. Without loss of generality, we work with the set Π directly as the alphabet rather than its power set.

a set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$*, and a labeling function* $L : X \to \Pi$*, compute a control policy* υ *(if existing) such that* $L(\mathbf{x}_{x,\upsilon}) \models \mathcal{A}$ *for all* $x \in L^{-1}(p_i)$ *and some* $i \in [0; M]$*.*

Finding a solution to Problem 6.3.4 (if existing) is difficult in general. In this chapter, we provide a method that is sound in solving the problem. To construct a control policy υ, our approach utilizes a notion of control barrier functions as defined in the next section. Later, we provide a compositional approach on constructing such control barrier functions to make it tractable for large-scale systems.

6.4 Control Barrier Function

In this section, we define a notion of control barrier functions.

Definition 6.4.1. *A function* $\mathcal{B} : X \to \mathbb{R}_0^+$ *is a control barrier function for an interconnected control system* $\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}})$ *as in Definition 6.2.2 if for any* $x \in X$ *there exists an input* $u \in U$ *such that*

$$\mathcal{B}(f_{\mathcal{I}}(x, u)) \leq \kappa(\mathcal{B}(x)), \tag{6.4.1}$$

for some $\kappa \in \mathcal{K}_\infty$ *with* $\kappa \leq \mathcal{I}_d$*.*

Note that the above definition associates a stationary policy $\mathsf{u} : X \to U$ according to the existential quantifier on the input for any state $x \in X$. The importance of the existence of a control barrier function for system $\Sigma_{\mathcal{I}}$ is shown in the following proposition.

Proposition 6.4.2. *Consider an interconnected control system* $\Sigma_{\mathcal{I}} = (X, U, f_{\mathcal{I}})$*, and sets* $X_a, X_b \subseteq X$*. Assume that there exits a control barrier function* $\mathcal{B} : X \to \mathbb{R}_0^+$ *as defined in Definition 6.4.1 with a stationary policy* $\mathsf{u} : X \to U$ *and constants* $\epsilon_1, \epsilon_2 \in \mathbb{R}^+$ *with* $\epsilon_2 \geq \epsilon_1$ *such that*

$$\mathcal{B}(x) \leq \epsilon_1, \qquad \forall x \in X_a, \tag{6.4.2}$$

$$\mathcal{B}(x) > \epsilon_2, \qquad \forall x \in X_b. \tag{6.4.3}$$

Then, for the state run $\mathbf{x}_{x,\mathsf{u}}$ *of* $\Sigma_{\mathcal{I}}$ *starting from any initial state* $x \in X_a$ *and under corresponding policy* $\mathsf{u}(\cdot)$*, one has* $\mathbf{x}_{x,\mathsf{u}}(k) \cap X_b = \emptyset$*,* $\forall k \in \mathbb{N}_0$*.*

Proof. We prove by contradiction. Consider a state run $\mathbf{x}_{x,\mathsf{u}}$ of $\Sigma_{\mathcal{I}}$ that starts at some $x \in X_a$. Suppose $\mathbf{x}_{x,\mathsf{u}}$ reaches a state inside X_b. Following (6.4.2) and (6.4.3), one has $\mathcal{B}(\mathbf{x}(0)) \leq \epsilon_1$ and $\mathcal{B}(\mathbf{x}(k)) > \epsilon_2$ for some $k \in \mathbb{N}_0$. Since $\mathcal{B}(\cdot)$ is a control barrier function and by using inequality (6.4.1), one can conclude that $\epsilon_2 < \mathcal{B}(\mathbf{x}(k)) \leq \mathcal{B}(\mathbf{x}(0)) \leq \epsilon_1$. This contradicts $\epsilon_2 \geq \epsilon_1$ which completes the proof. □

The interpretation of Proposition 6.4.2 is illustrated in Figure 6.2. In the next section, we discuss how to translate Problem 6.3.4 for a given specification into the computation of a collection of control barrier functions each satisfying conditions as in Proposition 6.4.2.

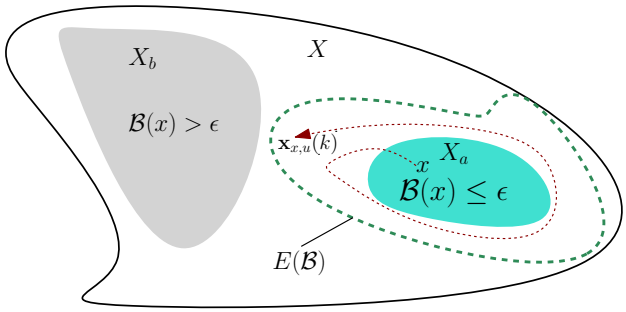

Figure 6.2: Illustration of a set X containing sets X_a and X_b: the dashed line illustrates the ϵ-level set of $\mathcal{B}$, defined as $E(\mathcal{B})=\{x\in X|\mathcal{B}(x)=\epsilon\}$, and the dotted curve is the run of system $\Sigma_{\mathcal{I}}$.

6.5 Formal Synthesis using Control Barrier Functions

In order to synthesize control policies using control barrier functions enforcing specifications expressed by DCA $\mathcal{A}$, we first provide the decomposition of specifications into sequential reachabilities.

6.5.1 Sequential Reachability Decomposition

Consider a DCA $\mathcal{A} = (Q, Q_0, \Pi, \mathfrak{d}, F)$ expressing the properties of interest for the system $\Sigma_{\mathcal{I}}$. Consider the DBA $\mathcal{A}^c = (Q, Q_0, \Pi, \mathfrak{d}, F)$ whose language is the complement of the language of DCA $\mathcal{A}$. As one can readily see, the DBA $\mathcal{A}^c$ has the same structure as the DCA $\mathcal{A}$, but with the Büchi accepting condition. The infinite sequence $\mathbf{q} = (q_0, q_1, \ldots) \in Q^\omega$ is called an accepting state run if $q_0 \in Q_0$ and there exists infinitely many $j \geq 0$ such that $q_j \in F$, and there exists an infinite word $\sigma = (\sigma_0, \sigma_1, \ldots) \in \Pi^\omega$ such that $q_k \xrightarrow{\sigma_k} q_{k+1}$ for all $k \in \mathbb{N}_0$. For a given accepting state run $\mathbf{q}$, we denote the corresponding infinite words by $\sigma(\mathbf{q}) \subseteq \Pi^\omega$. We also use a similar notation to denote finite words corresponding to finite state runs (i.e., $\sigma(\overline{\mathbf{q}}) \in \Pi^n$ for $\overline{\mathbf{q}} \in Q^{n+1}, n \in \mathbb{N}_0$). It is known [BKL08, Lemma 4.39] that there exists a word $\sigma \in \Pi^\omega$ accepted by $\mathcal{A}^c$ if and only if there exists a state run of $\mathcal{A}^c$ of the form $\mathbf{q} = (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, (q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s)^\omega) \in Q^\omega$, where $\mathsf{m}_r, \mathsf{m}_s \in \mathbb{N}$, $q_0^r \in Q_0$ and $q_0^s \in F$. Let $\overline{\mathbf{q}}$ be a finite state run fragment of an accepting run $\mathbf{q}$ constructed by considering infinite sequence $(q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s)$ only once and is given by $\overline{\mathbf{q}} = (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s, q_0^s) \in Q^*$. Let $\overline{\mathcal{R}}$ be the set of all such finite state run fragments excluding self-loops,

$$\begin{aligned}\overline{\mathcal{R}} := \{\overline{\mathbf{q}} =& (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s, q_0^s) \mid q_0^r \in Q_0, q_0^s \in F, \\ & q_i^r \neq q_{i+1}^r, \forall i < \mathsf{m}_r, \text{ and } q_j^s \neq q_{j+1}^s, \forall j < \mathsf{m}_s\}. \end{aligned} \tag{6.5.1}$$

For each $p \in \Pi$, we define a set $\overline{\mathcal{R}}^p$ as

$$\overline{\mathcal{R}}^p := \{\overline{\mathbf{q}} = (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s, q_0^s) \in \mathcal{R} | \sigma(q_0^r, q_1^r) = p\}. \tag{6.5.2}$$

Decomposition into sequential reachability is performed as follows. For any $\overline{\mathbf{q}} = (q_0, q_1, \ldots, q_{\mathsf{m}_r+\mathsf{m}_s+3}) \in \overline{\mathcal{R}}^p$, we define $\overline{\mathcal{P}}^p(\overline{\mathbf{q}})$ as a set of all state runs of length 3,

$$\overline{\mathcal{P}}^p(\overline{\mathbf{q}}) := \{(q_i, q_{i+1}, q_{i+2},) \mid 0 \leq i \leq \mathsf{m}_r + \mathsf{m}_s + 1\}. \tag{6.5.3}$$

We define $\overline{\mathcal{P}}(\mathcal{A}^c) = \bigcup_{p\in\Pi} \bigcup_{\overline{\mathbf{q}}\in\overline{\mathcal{R}}^p} \overline{\mathcal{P}}^p(\overline{\mathbf{q}})$.

Having $\overline{\mathcal{P}}^p(\overline{\mathbf{q}})$ defined in (6.5.3) as the set of state runs of length 3, now we provide a systematic approach to compute a policy such that the state runs of $\Sigma_{\mathcal{I}}$ satisfy the specification expressed by DCA $\mathcal{A}$. Given DBA $\mathcal{A}^c$, our approach relies on performing computation of control barrier functions for each element of $\overline{\mathcal{P}}(\mathcal{A}^c)$, which at the end provides control policies ensuring that we never have accepting runs in the complement of the given specification (i.e., DCA $\mathcal{A}$). To provide the result on the construction of control policies to solve Problem 6.3.4, we provide the following lemma which is a direct consequence of results in Proposition 6.4.2 and, hence, provided without a proof.

Lemma 6.5.1. *For $p \in \Pi$ and $\overline{\boldsymbol{q}} \in \overline{\mathcal{R}}^p$, consider $(q, q', q'') \in \overline{\mathcal{P}}^p(\overline{\boldsymbol{q}})$. If there exists a control barrier function with stationary policy $\mathsf{u}(\cdot)$ satisfying conditions (6.4.2) and (6.4.3) in Proposition 6.4.2 with $X_a = L^{-1}(\sigma(q, q'))$ and $X_b = L^{-1}(\sigma(q', q''))$, then the state run $\mathbf{x}_{x,\mathsf{u}}$ of $\Sigma_{\mathcal{I}}$ starting from any initial state $x \in X_a$ under policy $\mathsf{u}(\cdot)$ satisfies $\mathbf{x}_{x,\mathsf{u}}(k) \cap L^{-1}(\sigma(q', q'')) = \emptyset\ \forall k \in \mathbb{N}_0$.*

Lemma 6.5.1 uses control barrier functions along with appropriate choices of stationary control policies $\mathsf{u}(\cdot)$ for elements in $\overline{\mathcal{P}}(\mathcal{A}^c)$ as mentioned in Proposition 6.4.2. However, computation of control barrier functions and the controllers for each element of $\overline{\mathcal{P}}(\mathcal{A}^c)$ can cause ambiguity similar to the one discussed in Section 5.6. That can be similarly resolved by defining the partition set of elements sharing a common control barrier function and a corresponding control policy as

$$\overline{\mathfrak{S}}_{(q,q',\Delta(q'))} := \{(q, q', q'') \in \overline{\mathcal{P}}(\mathcal{A}^c) \mid q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\}.$$

The control barrier function and the control policy corresponding to the partition set $\overline{\mathfrak{S}}_{(q,q',\Delta(q'))}$ are denoted by $\mathcal{B}_{\overline{\mathfrak{S}}_{(q,q',\Delta(q'))}}(x)$ and $\mathsf{u}_{\overline{\mathfrak{S}}_{(q,q',\Delta(q'))}}(x)$, respectively. Thus, for all $\nu \in \overline{\mathcal{P}}(\mathcal{A}^c)$, we have

$$\mathcal{B}_\nu(x) = \mathcal{B}_{\overline{\mathfrak{S}}_{(q,q',\Delta(q'))}}(x) \text{ and } \mathsf{u}_\nu(x) = \mathsf{u}_{\overline{\mathfrak{S}}_{(q,q',\Delta(q'))}}(x), \text{if } \nu \in \overline{\mathfrak{S}}_{(q,q',\Delta(q'))}. \tag{6.5.4}$$

6.5.2 Hybrid Control Policy

Similar to Subection 5.6.1, we define control policy with the help of the switching mechanism given by an automata $\mathcal{A}_{\mathfrak{m}} = (Q_{\mathfrak{m}}, Q_{\mathfrak{m}0}, \Pi_{\mathfrak{m}}, \mathfrak{d}_{\mathfrak{m}})$, where $Q_{\mathfrak{m}} := Q_{\mathfrak{m}0} \cup \{(q, q', \Delta(q')) \mid q, q' \in Q\}$ is the set of states, $Q_{\mathfrak{m}0} := \{(q_0, \Delta(q_0)) \mid q_0 \in Q_0\}$ is the set of initial states, $\Pi_{\mathfrak{m}} = \Pi$, and the transition relation $(q_{\mathfrak{m}}, \sigma, q'_{\mathfrak{m}}) \in \mathfrak{d}_{\mathfrak{m}}$ is defined as

- for all $q_{\mathfrak{m}} = (q_0, \Delta(q_0)) \in Q_{\mathfrak{m}0}$, $(q_0, \Delta(q_0)) \overset{\sigma(q_0,q'')}{\longrightarrow} (q_0, q'', \Delta(q''))$, where $q_0 \overset{\sigma(q_0,q'')}{\longrightarrow} q''$;
- for all $q_{\mathfrak{m}} = (q, q', \Delta(q')) \in Q_{\mathfrak{m}} \setminus Q_{\mathfrak{m}0}$, $(q, q', \Delta(q')) \overset{\sigma(q',q'')}{\longrightarrow} (q', q'', \Delta(q''))$, such that $q, q', q'' \in Q$, $q' \overset{\sigma(q',q'')}{\longrightarrow} q''$.

The control policy that is a candidate for solving Problem 6.3.4 is given by

$$\upsilon(x, q_{\mathfrak{m}}) = \mathsf{u}_{\overline{\mathfrak{S}}_{(q'_{\mathfrak{m}})}}(x), \quad \forall (q_{\mathfrak{m}}, L(x), q'_{\mathfrak{m}}) \in \mathfrak{d}_{\mathfrak{m}}. \tag{6.5.5}$$

In the next theorem, we show that the policy given in (6.5.5) is indeed a solution for Problem 6.3.4.

Theorem 6.5.2. *Given $p \in \Pi$, assume that there exists $(q, q', q'') \in \overline{\mathcal{P}}^p(\overline{\boldsymbol{q}})$, for all $\overline{\boldsymbol{q}} \in \overline{\mathcal{R}}^p$ for which we have a control barrier function and a controller as given in* (6.5.4)*. Then the state run $\mathbf{x}_{x,\upsilon}$ of $\Sigma_{\mathcal{I}}$ starting from any initial state $x \in L^{-1}(p)$ under policy υ given in* (6.5.5) *satisfies the accepting language of DCA $\mathcal{A}$, i.e., $L(\mathbf{x}_{x,\upsilon}(k)) \models \mathcal{A}$ for all $k \in \mathbb{N}_0$.*

Proof. Consider $p \in \Pi$ and an accepting state run $\mathbf{q} = (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, (q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s)^\omega) \in Q^\omega$ in $\mathcal{A}^c$ with $\sigma(q_0^r, q_1^r) = p$. Let the corresponding finite state run be $\overline{\mathbf{q}} \in \overline{\mathcal{R}}^p$ as defined in Subsection 6.5.1. If for a triplet $(q, q', q'') \in \overline{\mathcal{P}}^p(\overline{\mathbf{q}})$ one can find a control barrier function with a stationary control policy $\mathsf{u}(\cdot)$, from Lemma 6.5.1 one can conclude $\sigma(\mathbf{q}) \notin \mathcal{L}(\mathcal{A}^c)$. Now, if there exist control barrier functions and corresponding controllers as defined in (6.5.4) for a triplet $(q, q', q'') \in \overline{\mathcal{P}}^p(\overline{\mathbf{q}})$ for any $\overline{\mathbf{q}} \in \overline{\mathcal{R}}^p$, one has $\sigma(\mathbf{q}) \notin \mathcal{L}(\mathcal{A}^c)$ for any accepting state run $\mathbf{q} = (q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r, (q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s)^\omega) \in Q^\omega$ satisfying $\sigma(q_0^r, q_1^r) = p$. By utilizing the definition of labeling function L, this implies that the state run $\mathbf{x}_{x,\upsilon}$ of $\Sigma_{\mathcal{I}}$ starting from any initial state $x \in L^{-1}(p)$ under policy υ given in (6.5.5) satisfies $L(\mathbf{x}_{x,\upsilon}(k)) \notin \mathcal{L}(\mathcal{A}^c)$ for all $k \in \mathbb{N}_0$. Hence, we have $L(\mathbf{x}_{x,\upsilon}(k)) \in \mathcal{L}(\mathcal{A})$ for all $k \in \mathbb{N}_0$ and for any initial state $x \in L^{-1}(p)$. □

Remark 6.5.3. *Theorem 6.5.2 says that in order to satisfy the given specification by the system $\Sigma_{\mathcal{I}}$ starting from any initial state $x \in L^{-1}(p)$, one needs to find a control barrier function as in* (6.5.4) *satisfying Lemma 6.5.1 for at least one $(q, q', q'') \in \overline{\mathcal{P}}^p(\overline{\boldsymbol{q}})$ for each $\overline{\boldsymbol{q}} \in \overline{\mathcal{R}}^p$. For the rest, one can choose control inputs arbitrarily.*

Remark 6.5.4. *For any $(q, q', q'') \in \overline{\mathfrak{S}}_{(q,q',\Delta(q'))}$, if $L^{-1}(\sigma(q, q')) \cap L^{-1}(\sigma(q', q'')) \neq \emptyset$, there exists no control barrier function satisfying conditions in Proposition 6.4.2. This follows directly due to the conflict in conditions* (6.4.2) *and* (6.4.3)*.*

6.6 Compositional Construction of Control Barrier Functions

In this section, we provide a method for compositional construction of control barrier functions for interconnected systems $\Sigma_{\mathcal{I}}$ in Definition 6.2.2. Suppose we are given control subsystems $\Sigma_i = (X_i, U_i, W_i, f_i, Y_i, h_i)$, $i \in [1, N]$, and assume sets X_a and X_b introduced in Proposition 6.4.2 can be decomposed as $X_a = \prod_{i=1}^N X_{ai}$ and $X_b = \prod_{i=1}^N X_{bi}$. Note that sets X_a and X_b are associated with some atomic propositions in Π through a labeling function $L : X \to \Pi$. This implies that all the sets associated with atomic propositions in Π have the decomposed structure as X_a and X_b. The result provided in this section is mainly used to obtain control barrier functions compositionally to satisfy the reachability tasks as given in Lemma 6.5.1. Here, we assume that each control subsystem Σ_i admits a local control barrier function as defined next.

Definition 6.6.1. *Let $\Sigma_i = (X_i, U_i, W_i, f_i, Y_i, h_i)$ be a control subsystem, where $i \in [1; N]$. A function $\mathcal{B}_i : X_i \to \mathbb{R}_0^+$ is called a local control barrier function for Σ_i if it satisfies the following conditions:*

$$\mathcal{B}_i(x_i) \geq \alpha_i(\|h_i(x_i)\|_\infty), \qquad \forall x_i \in X_i, \tag{6.6.1}$$
$$\mathcal{B}_i(x_i) \leq \bar{\epsilon}_i, \qquad \forall x_i \in X_{ai}, \tag{6.6.2}$$
$$\mathcal{B}_i(x_i) > \underline{\epsilon}_i, \qquad \forall x_i \in X_{bi}, \tag{6.6.3}$$

and $\forall x_i \in X_i \; \exists \, u_i \in U_i, \forall w_i \in W_i$ such that

$$\mathcal{B}_i(f_i(x_i, w_i, u_i)) \leq \max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\|w_i\|_\infty)\}, \tag{6.6.4}$$

for some $\alpha_i, \kappa_i, \gamma_{wi} \in \mathcal{K}_\infty$ with $\kappa_i \leq \mathcal{I}_d$, and some $\underline{\epsilon}_i, \bar{\epsilon}_i \in \mathbb{R}_0^+$.

Local control barrier functions of subsystems are mainly for constructing control barrier functions for the interconnected systems and they are not used directly for verifying any reachability task.

Remark 6.6.2. *Note that condition $\epsilon_1 \leq \epsilon_2$ in Definition 6.4.1 requires implicitly that $X_a \cap X_b = \emptyset$. However, in Definition 6.6.1 we do not require any condition between $\underline{\epsilon}_i$ and $\bar{\epsilon}_i$ because one may have $X_{ai} \cap X_{bi} \neq \emptyset$ even though $X_a \cap X_b = \emptyset$.*

Remark 6.6.3. *Note that condition* (6.6.4) *in Definition 6.6.1 implies that control input u_i only depends on the state x_i and is independent of internal input w_i. This allows us to design (if possible) decentralized control policies which do not require state information of other subsystems. However, if we change the sequence of quantifiers in* (6.6.4) *to $\forall x_i \in X_i \; \forall w_i \in W_i \; \exists \, u_i \in U_i$, then one obtains distributed control policies which require state informations of neighboring subsystems through internal inputs w_i.*

For functions κ_i, α_i, and γ_{wi} associated with $\mathcal{B}_i$, $\forall i \in [1; N]$, appeared in Definition 6.6.1, we define

$$\gamma_{ij} := \begin{cases} \kappa_i & \text{if} \quad i = j, \\ \gamma_{wi} \circ \alpha_j^{-1} & \text{if} \quad i \neq j, \end{cases} \quad \forall i, j \in [1; N]. \tag{6.6.5}$$

In order to establish the main compositionality results of the chapter, we raise the following small-gain type assumption.

Assumption 6.6.4. *Assume that functions γ_{ij} defined in* (6.6.5) *satisfy*

$$\gamma_{i_1 i_2} \circ \gamma_{i_2 i_3} \circ \cdots \circ \gamma_{i_{r-1} i_r} \circ \gamma_{i_r i_1} < \mathcal{I}_d, \tag{6.6.6}$$

$\forall (i_1, \ldots, i_r) \in [1; N]^r$, *where $r \in [1; N]$.*

Note that by using Theorem 5.2 in [DRW10], the small-gain condition (6.6.6) *implies that there exist $\varphi_i \in \mathcal{K}_\infty$, $\forall i \in [1; N]$, satisfying*

$$\max_{j \in [1; N]} \{\varphi_i^{-1} \circ \gamma_{ij} \circ \varphi_j\} < \mathcal{I}_d. \tag{6.6.7}$$

The next theorem provides a compositionality approach to compute a control barrier function for interconnected system $\Sigma_{\mathcal{I}}$ in Definition 6.2.2 via local control barrier functions of subsystems Σ_i.

Theorem 6.6.5. *Consider the interconnected control system $\Sigma_{\mathcal{I}} = \mathcal{I}(\Sigma_1, \ldots, \Sigma_N)$ induced by $N \in \mathbb{N}_{\geq 1}$ control subsystems Σ_i. Assume that each Σ_i admits a local control barrier function $\mathcal{B}_i$ as defined in Definition 6.6.1. Let Assumption 6.6.4 hold and $\max_{i\in[1;N]}\{\varphi_i^{-1}(\overline{\epsilon}_i)\} \leq \max_{i\in[1;N]}\{\varphi_i^{-1}(\underline{\epsilon}_i)\}$. Then, function $\mathcal{B} : X \to \mathbb{R}_0^+$ defined as*

$$\mathcal{B}(x) := \max_{i\in[1;N]}\{\varphi_i^{-1} \circ \mathcal{B}_i(x_i)\},$$

is a control barrier function for the interconnected control system $\Sigma_{\mathcal{I}}$ satisfying conditions (6.4.2) *and* (6.4.3) *in Proposition 6.4.2 with $X_a = \prod_{i=1}^N X_{ai}$ and $X_b = \prod_{i=1}^N X_{bi}$.*

Proof. First, let $\kappa = \max_{i,j\in[1,N]}\{\varphi_i^{-1} \circ \gamma_{ij} \circ \varphi_j\}$. It follows from (6.6.7) that $\kappa < \mathcal{I}_d$.

Now $\forall x = [x_1; \ldots; x_N] \in \prod_{i=1}^N X_i = X$ $\exists u = [u_1; \ldots; u_N] \in \prod_{i=1}^N U_i = U$ such that one gets the following chain of inequalities

$$\begin{aligned}
\mathcal{B}(f_{\mathcal{I}}(x,u)) &= \max_i\{\varphi_i^{-1} \circ \mathcal{B}_i(f_i(x_i,u_i,w_i))\} \leq \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\|w_i\|_\infty)\}\big)\Big\}\\
&= \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\max_{j,j\neq i}\{\|w_{ij}\|_\infty\})\}\big)\Big\}\\
&= \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\max_{j,j\neq i}\{\|y_{ji}\|_\infty\})\}\big)\Big\}\\
&= \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\max_{j,j\neq i}\{\|h_{ji}(x_j)\|_\infty\})\}\big)\Big\}\\
&\leq \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\max_{j,j\neq i}\{\|h_j(x_j)\|_\infty\})\}\big)\Big\}\\
&\leq \max_i \Big\{\varphi_i^{-1}\big(\max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\max_{j,j\neq i}\{\alpha_j^{-1} \circ \mathcal{B}_j(x_j)\}\big)\Big\}\\
&\leq \max_{i,j} \Big\{\varphi_i^{-1} \circ \gamma_{ij} \circ \mathcal{B}_j(x_j)\Big\} = \max_{i,j} \Big\{\varphi_i^{-1} \circ \gamma_{ij} \circ \varphi_j \circ \varphi_j^{-1} \circ \mathcal{B}_j(x_j)\Big\}\\
&\leq \max_{i,j,l} \Big\{\varphi_i^{-1} \circ \gamma_{ij} \circ \varphi_j \circ \varphi_l^{-1} \circ \mathcal{B}_l(x_l)\Big\} = \max_{i,j} \Big\{\varphi_i^{-1} \circ \gamma_{ij} \circ \varphi_j \circ \mathcal{B}(x)\Big\} = \kappa(\mathcal{B}(x)),
\end{aligned}$$

satisfying condition (6.4.1).

Now, we show that conditions (6.4.2) and (6.4.3) hold. From conditions (6.6.2) and (6.6.3), $\forall x = [x_1; \ldots; x_N] \in \prod_{i=1}^N X_{ai} = X_a$, one has

$$\mathcal{B}(x) = \max_{i\in[1;N]}\{\varphi_i^{-1} \circ \mathcal{B}_i(x_i)\} \leq \max_{i\in[1;N]}\{\varphi_i^{-1}(\overline{\epsilon}_i)\},$$

and $\forall x = [x_1; \ldots; x_N] \in \prod_{i=1}^N X_{bi} = X_b$

$$\mathcal{B}(x) = \max_{i\in[1;N]}\{\varphi_i^{-1} \circ \mathcal{B}_i(x_i)\} > \max_{i\in[1;N]}\{\varphi_i^{-1}(\underline{\epsilon}_i)\},$$

satisfying conditions (6.4.2) and (6.4.3) with

$$\epsilon_1 = \max_{i\in[1;N]}\{\varphi_i^{-1}(\overline{\epsilon}_i)\}, \epsilon_2 = \max_{i\in[1;N]}\{\varphi_i^{-1}(\underline{\epsilon}_i)\}.$$

This concludes the proof. □

Now, we provide a discussion about the feasibility of inequality

$$\max_{i\in[1;N]}\{\varphi_i^{-1}(\overline{\epsilon}_i)\} \leq \max_{i\in[1;N]}\{\varphi_i^{-1}(\underline{\epsilon}_i)\}, \tag{6.6.8}$$

required in Theorem 6.6.5. In general, inequality (6.6.8) is not very restrictive. Indeed, functions φ_i in (6.6.7) play the role of rescaling the control barrier functions of the individual subsystems while normalizing the effect of internal gains of other subsystems (see [DRW10] for a similar discussion in the context of Lyapunov stability). Due to this scaling, one can expect that such an inequality holds in many applications.

In the case that $X_{ai} \cap X_{bi} = \emptyset$, $\forall i \in [1;N]$, inequality (6.6.8) always holds with $\max_{i\in[1;N]}\{\overline{\epsilon}_i\} \leq \min_{i\in[1;N]}\{\underline{\epsilon}_i\}$. Note that we can always impose such a condition over $\underline{\epsilon}_i$ and $\overline{\epsilon}_i$ whenever $X_{ai} \cap X_{bi} = \emptyset, \forall i \in [1;N]$. In the case where $\varphi_i = \varphi_j, \forall i,j \in [1;N]$, inequality (6.6.8) simply reduces to $\max_{i\in[1;N]}\{\overline{\epsilon}_i\} \leq \max_{i\in[1;N]}\{\underline{\epsilon}_i\}$.

Remark 6.6.6. *In the context of stability analysis of interconnected nonlinear control systems, condition* (6.6.6) *is commonly used to show different stability proprieties (e.g., uniform asymptotic stability or input-to-state stability) for the entire network by investigating stability criteria for subsystems. Moreover, condition* (6.6.6) *is also been shown to be tight and cannot be weakened in the context of stability verification of interconnected systems. We refer interested readers to [DRW07] for more details on the tightness analysis of small-gain condition* (6.6.6).

Remark 6.6.7. *Here, we provide a general guideline on the computation of $\mathcal{K}_\infty$ functions $\varphi_i, i \in [1;N]$ as follows: (i) In the case of having $N \geq 1$ subsystems, functions $\varphi_i, i \in [1;N]$, can be constructed numerically using the algorithm proposed in [Eav72] and the technique provided in [DRW10, Proposition 8.8], see [Ruf07, Chapter 4]; (ii) Simple construction techniques are provided in [JMW96] and [DRW10, Section 9] for the case of two and three subsystems, respectively; (iii) the $\mathcal{K}_\infty$ functions $\varphi_i, i \in [1;N]$, can be always chosen as identity functions provided that $\gamma_{ij} < \mathcal{I}_d$, $\forall\ i,j \in [1;N]$, for functions γ_{ij} appeared in* (6.6.5).

6.6.1 Computation of Local Control Barrier Functions

Proving the existence of a control barrier function and finding one are in general hard problems. However, under some assumptions over systems dynamics, control inputs, and labeling functions, one can search for a local control barrier functions and corresponding control policies of specific forms. In this subsection, we provide two potential solutions: one using sum-of-squares (SOS) program and the other one using counterexample guided inductive synthesis (CEGIS).

Sum-of-squares program

In order to formulate conditions in Definition 6.6.1 as an SOS optimization to search for a polynomial local control barrier function $\mathcal{B}_i(\cdot)$ and a polynomial stationary control policy $\mathsf{u}_i(\cdot)$, we raise the following assumption.

Assumption 6.6.8. *Subsystem Σ_i has a continuous state set $X_i \subseteq \mathbb{R}^{n_i}$, a continuous external input set $U_i \subseteq \mathbb{R}^{m_i}$, and a continuous internal input set $W_i \subseteq \mathbb{R}^{p_i}$. Its transition function $f_i : X_i \times U_i \times W_i \to X_i$ is polynomial in variables x_i, u_i, and w_i.*

The following lemma provides a set of sufficient conditions for the existence of local control barrier functions required in Theorem 6.6.5, which can be solved as an SOS optimization.

Lemma 6.6.9. *Suppose Assumption 6.6.8 holds and sets X_{ai}, X_{bi}, X_i can be defined as $X_{ai} = \{x_i \in \mathbb{R}^{n_i} \mid \mathsf{g}_{ai}(x_i) \geq 0\}$, $X_{bi} = \{x_i \in \mathbb{R}^{n_i} \mid \mathsf{g}_{bi}(x_i) \geq 0\}$, $X_i = \{x_i \in \mathbb{R}^{n_i} \mid \mathsf{g}_i(x_i) \geq 0\}$, and $W_i = \{w_i \in \mathbb{R}^{p_i} \mid \mathsf{g}_{wi}(x_i) \geq 0\}$, where the inequalities are defined element-wise and $\mathsf{g}_{ai}, \mathsf{g}_{bi}, \mathsf{g}_i, \mathsf{g}_{wi}$ are vectors of polynomial functions. Suppose there exists a sum-of-squares polynomial $\mathcal{B}_i(x_i)$, polynomials $\lambda_{u_{ji}}(x_i)$ corresponding to the j^{th} input in $u_i = (u_{1i}, u_{2i}, \ldots, u_{m_i i}) \in U_i \subseteq \mathbb{R}^{m_i}$, and vectors of sum-of-squares polynomials $\lambda_{ai}(x_i)$, $\lambda_{bi}(x_i)$, $\lambda_i(x_i)$, $\overline{\lambda}_i(x_i)$, $\lambda_{wi}(w_i)$ of appropriate size, and $\hat{\alpha}_i, \hat{\kappa}_i, \hat{\gamma}_{wi} \in \mathcal{K}_\infty$ with $\hat{\kappa}_i \leq \mathcal{I}_d$ such that following expressions are sum-of-squares polynomials:*

$$\mathcal{B}_i(x_i) - \hat{\alpha}_i(\|h_i(x_i)\|_\infty) - \lambda_i^T(x_i)\mathsf{g}_i(x_i), \tag{6.6.9}$$

$$-\mathcal{B}_i(x_i) + \overline{\epsilon}_i - \lambda_{ai}^T(x_i)\mathsf{g}_{ai}(x_i), \tag{6.6.10}$$

$$\mathcal{B}_i(x_i) - \underline{\epsilon}_i - \lambda_{bi}^T(x_i)\mathsf{g}_{bi}(x_i), \tag{6.6.11}$$

$$-\mathcal{B}_i(f_i(x_i, w_i, u_i)) + \hat{\kappa}_i(\mathcal{B}_i(x_i)) + \hat{\gamma}_{wi}(\|w_i\|_\infty) - \sum_{j=1}^{m_i}(u_{ji} - \lambda_{u_{ji}}(x_i))$$
$$-\overline{\lambda}_i^T(x_i)\mathsf{g}_i(x_i) - \lambda_{wi}^T(w_i)\mathsf{g}_{wi}(w_i), \tag{6.6.12}$$

where $\underline{\epsilon}_i, \overline{\epsilon}_i$ are the constants introduced in Definition 6.6.1. Then $\mathcal{B}_i(x_i)$ satisfies conditions (6.6.1)-(6.6.4) *in Definition 6.6.1 and $u_i = [\lambda_{u_{1i}}(x_i); \ldots, \lambda_{u_{m_i i}}(x_i)]$, $i \in [1, N]$, is the corresponding controller.*

Proof. Following a similar argument as the one in the proof of Lemma 5.7.2, conditions (6.6.9)-(6.6.12) imply

$$\mathcal{B}_i(x_i) \geq \hat{\alpha}_i(\|h_i(x_i)\|_\infty), \qquad \forall x_i \in X_i, \tag{6.6.13}$$

$$\mathcal{B}_i(x_i) \leq \overline{\epsilon}_i, \qquad \forall x_i \in X_{ai}, \tag{6.6.14}$$

$$\mathcal{B}_i(x_i) > \underline{\epsilon}_i, \qquad \forall x_i \in X_{bi}, \tag{6.6.15}$$

and $\forall x_i \in X_i\ \exists\ u_i \in U_i, \forall w_i \in W_i$ such that

$$\mathcal{B}_i(f_i(x_i, w_i, u_i)) \leq \hat{\kappa}_i(\mathcal{B}_i(x_i)) + \hat{\gamma}_{wi}(\|w_i\|_\infty). \tag{6.6.16}$$

By using Theorem 1 in [SGZ18], condition (6.6.16) can be written as

$$\mathcal{B}_i(f_i(x_i, w_i, u_i)) \leq \max\{\kappa_i(\mathcal{B}_i(x_i)), \gamma_{wi}(\|w_i\|_\infty)\},$$

where $\kappa_i = \mathcal{I}_d - (\mathcal{I}_d - \psi_i) \circ (\mathcal{I}_d - \hat{\kappa}_i)$, $\gamma_{wi} = (\mathcal{I}_d - \hat{\kappa}_i)^{-1} \circ \psi_i^{-1} \circ \hat{\gamma}_{wi}$, with $\psi_i \in \mathcal{K}_\infty$ chosen arbitrarily such that $\psi_i < \mathcal{I}_d$. Let $\alpha_i = \hat{\alpha}_i$ and this concludes the proof. □

Remark 6.6.10. *Note that function $\hat{\kappa}_i(\cdot)$ in (5.7.4) can cause nonlinearity on the unknown parameters of $\mathcal{B}_i$. A possible way to avoid this is to consider a linear function $\hat{\kappa}_i(r) = c_i r, \forall r \geq 0$, with some constant $0 < c_i < 1$. Then one can use bisection method to minimize the value of c_i.*

One can utilize existing tools such as `SOSTOOL` [PPP02] in conjunction with a semidefinite programming solver such as `SeDuMi` [Stu99] to compute a sum-of-squares polynomial $\mathcal{B}_i(x_i)$ satisfying (6.6.9)-(6.6.12).

Counter-example guided synthesis approach

This approach uses feasibility solvers for finding local control barrier functions of a given form using Satisfiability Modulo Theories (SMT) solvers such as `Z3` [dMB08], `OptiMathSAT` [ST15], `MathSAT` [CGSS13], or `dReal` [GKC13]. In order to use the CEGIS framework, we raise the following assumption.

Assumption 6.6.11. *Each control subsystem Σ_i, $i \in [1; N]$, has compact state set X_i, compact internal input set W_i, and a finite input set U_i.*

Under Assumption 6.6.11, conditions (6.6.1)-(6.6.4) can be rephrased as a satisfiability problem which can be searched for local control barrier function using the CEGIS approach. The feasibility condition that is required to be satisfied for the existence of a local control barrier function $\mathcal{B}_i$ is given in the following lemma.

Lemma 6.6.12. *Consider control subsystem $\Sigma_i = (X_i, U_i, W_i, f_i, Y_i, h_i)$ satisfying Assumption 6.6.11. Suppose there exists a function $\mathcal{B}_i(x_i)$ and $\mathcal{K}_\infty$ functions $\hat{\alpha}_i, \hat{\kappa}_i$ and $\hat{\gamma}_{wi}$ such that the following expression is true*

$$\begin{aligned}
&\bigwedge_{x_i \in X_i} \mathcal{B}_i(x_i) \geq \hat{\alpha}_i(\|h_i(x_i)\|_\infty) \bigwedge_{x_i \in X_{ai}} \mathcal{B}_i(x_i) \leq \bar{\epsilon}_i \bigwedge_{x_i \in X_{bi}} \mathcal{B}_i(x_i) > \underline{\epsilon}_i \\
&\bigwedge_{x_i \in X_i} \Big(\bigvee_{u_i \in U_i} \Big(\bigwedge_{w_i \in W_i} \big(\mathcal{B}_i(f_i(x_i, w_i, u_i)) \leq \hat{\kappa}_i(\mathcal{B}_i(x_i)) + \hat{\gamma}_{wi}(\|w_i\|_\infty)\big)\Big)\Big),
\end{aligned} \tag{6.6.17}$$

where $\underline{\epsilon}_i, \bar{\epsilon}_i$ are the constants introduced in Definition 6.6.1. Then $\mathcal{B}_i(x_i)$ satisfies conditions (6.6.1)-(6.6.4) in Definition 6.6.1.

Note that condition (6.6.17) implies conditions (6.6.13)-(6.6.16) which imply (6.6.1)-(6.6.4). One can utilize the CEGIS approach to search for control barrier functions solving the feasibility problem in (6.6.17). For the detailed discussion on CEGIS approach, we kindly refer interested readers to Section 5.7.2 (or [JSZ20a, Subsection 5.3.2]).

6.7 Case Studies

6.7.1 Room Temperature Control

The evolution of the temperature $\mathbf{T}$ of N rooms are described by the interconnected discrete-time model:

$$\Sigma_{\mathcal{I}} : \mathbf{T}(k+1) = A\mathbf{T}(k) + \alpha_e T_E + \alpha_h T_h \upsilon(k),$$

where $A \in \mathbb{R}^{N\times N}$ is a matrix with elements $\{A\}_{ii} = (1-2\alpha-\alpha_e-\alpha_h\upsilon_i(k))$, $\{A\}_{i(i+1)} = \{A\}_{(i+1)i} = \{A\}_{1N} = \{A\}_{N1} = \alpha$, $\forall i \in [1; N-1]$, and all other elements are identically zero, $\mathbf{T}(k) = [\mathbf{T}_1(k); \ldots; \mathbf{T}_N(k)]$, $\upsilon(k) = [\upsilon_1(k); \ldots; \upsilon_N(k)]$, $T_E = [T_{e1}; \ldots; T_{eN}]$, where $\upsilon_i(k) \in [0, 1]$ for all $i \in [1; N]$ represents ratio of the heater valve being open. The other parameters are as follow: $\forall i \in [1; N]$, $T_{ei} = 15\,^\circ C$ is the external temperature and $T_h = 55\,^\circ C$ is the heater temperature. Parameters $\alpha = 5 \times 10^{-2}$, $\alpha_e = 8 \times 10^{-3}$, and $\alpha_h = 3.6 \times 10^{-3}$ are heat exchange coefficients. All the parameters are adopted from Section 3.5. .

The state set of the system is $T \subseteq \mathbb{R}^N$. We consider regions of interest $X_0 = [20.5, 22.5]^N$, $X_1 = [0, 20]^N$, $X_2 = [23, 45]^N$, and $X_3 = T \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_j) = p_j$ for all $x_j \in X_j$, $j \in \{0, 1, 2, 3\}$. The objective is to compute a control policy ensuring satisfaction of the specification given by the accepting language of the DCA $\mathcal{A}$ in Figure 6.3. In English, language of $\mathcal{A}$ entails that if we start in X_0 it will always stay away from X_1 or X_2. Note that, the corresponding DBA $\mathcal{A}^c$ accepting complement of $\mathcal{L}(\mathcal{A})$ has exactly the same structure as in Figure 6.3, but with the Büchi accepting condition. One can readily see that, we have sets $\overline{\mathcal{P}}^{p_0} = \{(q_0, q_1, q_2), (q_1, q_2, q_2)\}$ and $\overline{\mathcal{P}}^{p_1} = \overline{\mathcal{P}}^{p_2} = \overline{\mathcal{P}}^{p_3} = \{(q_0, q_2, q_2)\}$. Following Remark 6.5.4, we only need to compute a control barrier function corresponding to triplet (q_0, q_1, q_2). In order to apply our com-

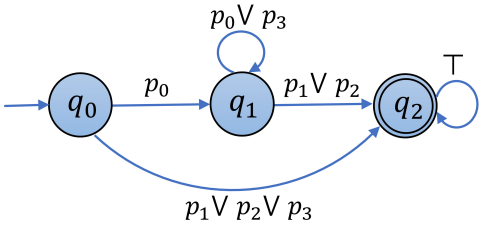

Figure 6.3: DCA $\mathcal{A}$ representing specification.

positionality result, we need to decompose the system $\Sigma_{\mathcal{I}}$ into subsystems Σ_i, $i \in [1; N]$. Accordingly, by introducing Σ_i described by

$$\Sigma_i : \begin{cases} \mathbf{T}_i(k+1) = a\mathbf{T}_i(k) + d\omega_i(k) + \alpha_e T_{ei} + \alpha_h T_h \upsilon_i(k), \\ \mathbf{y}_i(k) = \mathbf{T}_i(k), \end{cases}$$

one can readily verify that $\Sigma_{\mathcal{I}} = \mathcal{I}(\Sigma_1, \ldots, \Sigma_N)$, where $a = 1 - 2\alpha - \alpha_e - \alpha_h\upsilon_i(k)$, $d = [\alpha; \alpha]^T$, and $\omega_i(k) = [\mathbf{y}_{i-1}(k); \mathbf{y}_{i+1}(k)]$ (with $\mathbf{y}_0 = \mathbf{y}_N$ and $\mathbf{y}_{N+1} = \mathbf{y}_1$).

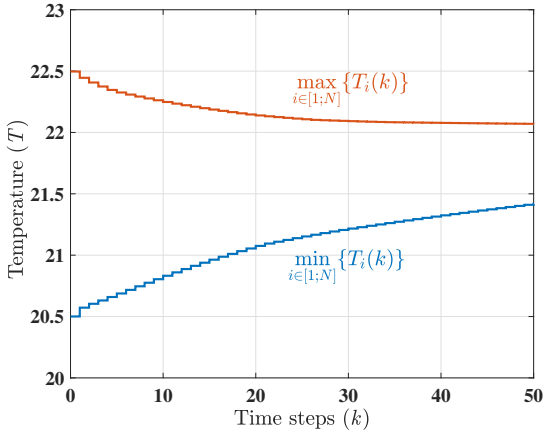

Figure 6.4: Bounds inside which trajectories are evolving.

To compute local control barrier functions, we solve sum-of-squares program using `SOSTOOLS` and `SeDuMi` as described in Subsection 5.7.1. By using Lemma 5.7.2, for all $i \in [1; N]$, we compute local control barrier functions of order 2 as $\mathcal{B}_i(x_i) = 0.07456x_i^2 - 3.18x_i + 73.79$ and the corresponding stationary control policy of order 1 as $\mathsf{u}_i(x_i) = -0.002398x_i + 0.5357$ with $X_{ai} = [20.5, 22.5]$, $X_{bi} = [0, 20] \cup [23, 45]$, constants $\overline{\epsilon}_i = \underline{\epsilon}_i = 40$, and functions $\hat{\alpha}_i(r) = 1.5r$, $\hat{\kappa}_i(r) = 0.65r$, and $\hat{\gamma}_{wi}(r) = 0.5r$ $\forall r \in \mathbb{R}_0^+$. One can readily verify that the small-gain assumption in (6.6.6) holds with $\gamma_{ij}(r) = 0.95r$, $\forall r \in \mathbb{R}_0^+$. Then by utilizing results in Theorem 6.6.5, we get overall control barrier function $\mathcal{B}(x) := \max_{i\in[1;N]}\{\varphi_i^{-1} \circ \mathcal{B}_i(x_i)\}$ with $\varphi_i = \mathcal{I}_d$ and corresponding control policy is given by $\mathsf{u}(x) = [\mathsf{u}_1(x_1); \ldots; \mathsf{u}_N(x_N)]$. One can readily see that only one stationary control policy is enough for enforcing the specification, thus we do not need switching mechanism. Figure 6.4 shows the maximum and minimum of state trajectories at each time-step of the closed-loop system $\Sigma_{\mathcal{I}}$ with 10000 rooms starting from an initial state in X_0.

6.7.2 Controlled Kuramoto Oscillators

For the second case study, we consider the Kuramoto oscillator which has large applications in neural networks [EK91], pacemakers in heart [WD13], automated vehicle coordination [KLMJ07], and power grids [DB10]. In particular, we apply our approach to a variant of the controlled Kuramoto model from [SA15]. The dynamic for an interconnection of N-oscillators is given by:

$$\Sigma_{\mathcal{I}} : \theta(k+1) = \theta(k) + \tau\Omega + \frac{\tau K}{N}\phi(\theta(k)) + \upsilon(k),$$

where $\theta(k) = [\theta_1(k); \ldots; \theta_N(k)] \in \Theta \subseteq [0, 2\pi]^N$ is the phase of the oscillators, $\Omega = [\Omega_1; \ldots; \Omega_N] = \mathbf{1}_N$ is the natural frequency of the oscillators, $\phi(\theta(k)) = [\sum_{j\in[1;N]} \sin(\theta_j(k) -$

$\theta_1(k)); \ldots; \sum_{j\in[1;N]} \sin(\theta_j(k) - \theta_N(k))]$, $K = 1$ is the coupling strength, $\tau = 0.2$, and control input $\upsilon(k) = [\upsilon_1(k); \ldots; \upsilon_N(k)]$, where $\upsilon_i(k) \in U_i = \{-0.6, -0.5, \ldots, 0.5, 0.6\}$, $i \in [1; N]$. We consider regions of interest $X_0 = [0, \frac{\pi}{3}]^N$, $X_1 = [\frac{5\pi}{12}, \frac{7\pi}{12}]^N$, $X_2 = [\frac{2\pi}{3}, \pi]^N$, $X_3 = [\pi, \frac{4\pi}{3}]^N$, $X_4 = [\frac{17\pi}{12}, \frac{19\pi}{12}]^N$ and $X_5 = [\frac{5\pi}{3}, 2\pi]^N$, $X_6 = X \setminus (X_0 \cup X_1 \cup X_2 \cup X_3 \cup X_4 \cup X_5)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6\}$ with labeling function $L(x_i) = p_i$ for all $x_i \in X_i$, $i \in \{0, 1, 2, 3, 4, 5, 6\}$. The objective is to compute a control policy ensuring satisfaction of the specification given by the accepting language of the DCA $\mathcal{A}$ in Figure 6.5. This corresponds to the LTL specification

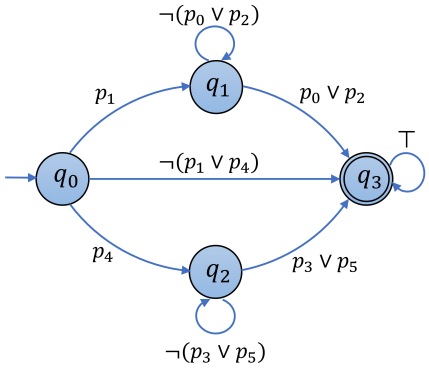

Figure 6.5: DCA $\mathcal{A}$ representing the specification.

$(p_1 \wedge \Box\neg(p_0 \vee p_2)) \vee (p_4 \wedge \Box\neg(p_3 \vee p_5))$. In English, language of $\mathcal{A}$ entails that if we start in X_1, it will always stay away from X_0 or X_2 or if we start in X_4, it will always stay away from X_3 or X_5. Note that, the DBA $\mathcal{A}^c$ accepting complement of $\mathcal{L}(\mathcal{A})$ has exactly the same structure as in Figure 6.5, but with the Büchi accepting condition. As described in Section 6.5.1, we have sets $\overline{\mathcal{P}}^{p_1} = \{(q_0, q_1, q_3), (q_1, q_3, q_3)\}$, $\overline{\mathcal{P}}^{p_4} = \{(q_0, q_2, q_3), (q_2, q_3, q_3)\}$, and $\overline{\mathcal{P}}^{p_0} = \overline{\mathcal{P}}^{p_2} = \overline{\mathcal{P}}^{p_3} = \overline{\mathcal{P}}^{p_5} = \overline{\mathcal{P}}^{p_6} = \{(q_0, q_3, q_3)\}$. Following Remark 6.5.4, there exists no control barrier function corresponding to (q_0, q_3, q_3), (q_1, q_3, q_3), and (q_2, q_3, q_3). This implies that we need to compute only two control barrier functions. Now by introducing subsystems Σ_i, $i \in [1; N]$, described by

$$\Sigma_i : \begin{cases} \theta_i(k+1) = \theta_i(k) + \tau\Omega_i + \frac{K\tau}{N} \sum_{j=1}^{N} \sin(\omega_{ij}(k) - \theta_i(k)) + \upsilon_i(k), \\ \vartheta_i(k) = \theta_i(k), \end{cases}$$

one can readily verify that $\Sigma_{\mathcal{I}} = \mathcal{I}(\Sigma_1, \ldots, \Sigma_N)$, where $\omega_{ij} = \theta_j$.

To compute these control barrier functions, we apply our compositionality technique and utilize CEGIS approach, as discussed in Subsection 6.6.1. For the demonstration of the results, we fix N=10000. The order 2 polynomial local control barrier function corresponding to triplet (q_0, q_1, q_3) is obtained for all $i \in [1; N]$, as $\mathcal{B}_i(x_i) = 10.9427x_i^2 - 34.3775x_i + 29$ with $X_{ai} = [\frac{5\pi}{12}, \frac{7\pi}{12}]$, $X_{bi} = [0, \frac{\pi}{3}] \cup [\frac{2\pi}{3}, \pi]$, constants $\overline{\epsilon}_i = \underline{\epsilon}_i = 5$, functions $\hat{\alpha}_i(r) = 0.5r^2$, $\hat{\kappa}_i(r) = 1.6 \times 10^{-6} r$, and $\hat{\gamma}_{w_i}(r) = 0.4368r^2$ $\forall r \in \mathbb{R}_0^+$ satisfying conditions in Lemma 6.6.12. Then, by utilizing results in Theorem 6.6.5, we get the overall control barrier function as $\mathcal{B}(x) := \max_{i\in[1;N]}\{\varphi_i^{-1}\mathcal{B}_i(x_i)\}$ with $\varphi_i = \mathcal{I}_d$

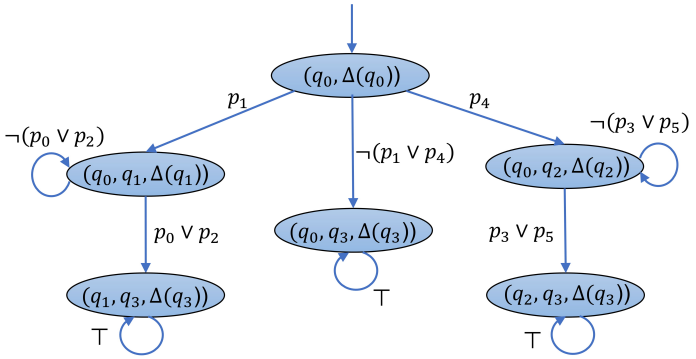

Figure 6.6: Switching mechanism for controllers.

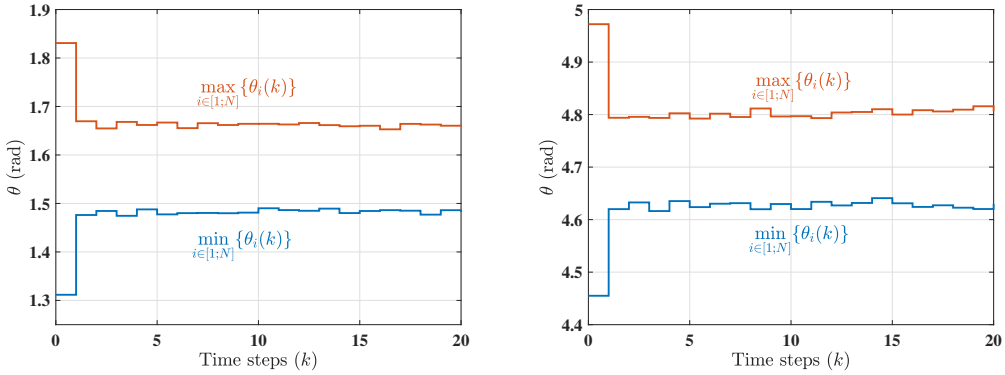

Figure 6.7: Bounds inside which trajectories of the Kuramoto model with 10000 oscillators evolve with an initial state starting in region X_1 (left) and an initial state starting in region X_4 (right).

and the corresponding *determinized* controller for each subsystem is given by $\mathsf{u}_i(x_i) = \min\{u_i \in U_i \mid \mathcal{B}_i(f_i(x_i, w_i^*, u_i)) \leq \hat{\kappa}_i(\mathcal{B}_i(x_i)) + \hat{\gamma}_{wi}(\|w_i^*\|_\infty)\}$ for an arbitrarily chosen $w_i^* \in W_i = [0, 2\pi]^{N-1}$. Similarly, the local control barrier function corresponding to triplet (q_0, q_2, q_3) is obtained for all $i \in [1; N]$, as $\mathcal{B}_i(x_i) = 7.2951x_i^2 - 68.7549x_i + 175$ with $X_{ai} = [\frac{17\pi}{12}, \frac{19\pi}{12}]$, $X_{bi} = [\pi, \frac{4\pi}{3}] \cup [\frac{5\pi}{3}, 2\pi]$, constants $\bar{\epsilon}_i = \underline{\epsilon}_i = 15$, functions $\alpha_i(r) = 0.5r^2$, $\hat{\kappa}_i(r) = 1.6 \times 10^{-6}r$, and $\hat{\gamma}_{w_i}(r) = 0.2912r^2$ for all $r \in \mathbb{R}_0^+$ satisfying conditions in Lemma 6.6.12. The corresponding determinized controller here is also given as $\mathsf{u}_i(x_i) = \min\{u_i \in U_i \mid \mathcal{B}_i(f_i(x_i, w_i^*, u_i)) \leq \hat{\kappa}_i(\mathcal{B}_i(x_i)) + \hat{\gamma}_{wi}(\|w_i^*\|_\infty)\}$ for an arbitrarily chosen $w_i^* \in W_i$. Note that in both scenarios the small-gain condition in (6.6.6) holds with $\gamma_{ij}(s) = 0.5824r$ and $\gamma_{ij}(s) = 0.8736r$, $\forall r \in \mathbb{R}_0^+$, respectively. The switching mechanism for controllers to obtain hybrid control policy $\upsilon(x, q_{\mathfrak{m}})$ as defined in (6.5.5) is shown in Figure 6.6. Figure 6.7(a) and Figure 6.7(b) show the maximum and minimum bounds inside which all the state trajectories of the closed-loop system $\Sigma_{\mathcal{I}}$ starting from an initial state in X_1 and X_4 evolves, respectively. From Figure 6.7, one can readily check the satisfaction of the given specification.

7 Controller Synthesis for Unknown Control Systems

The results provided in previous chapters assume the availability of a precise mathematical model of the system which is usually not possible in many real-world applications due to increasing complexity. In such cases, it is challenging to synthesize controller enforcing complex specification with some formal guarantee. In this chapter, we study the synthesis of hybrid controllers for unknown, continuous-time nonlinear control affine systems enforcing specifications expressed by co-Büchi automata as discussed in Subsection 6.3.1. We use a data-driven approach utilizing Gaussian processes, to learn unknown dynamics together with a probability bound on the accuracy of the learned model. Then, we provide a systematic approach to solve controller synthesis problems using control barrier functions providing the lower bound on the probability of satisfaction of the given specification.

7.1 Introduction

The conventional techniques for synthesizing controllers enforcing complex specifications require a precise mathematical model of the system including results proposed in previous chapters of this thesis. However, there are many control applications where the precise model description can not be derived analytically. In such cases, due to advances in sensor and processing technologies, one can take advantage of data-driven approaches from machine learning to identify unmodeled dynamics with high precision and complement the mathematical analysis from control theory. In order to solve such controller synthesis problems, in this chapter, we utilize the Gaussian process (GP) which is a data-driven approach providing a non-parametric probabilistic modeling framework.

7.1.1 Related Literature

Controllers using Gaussian processes

In recent years, there have been many results utilizing Gaussian processes to design controllers for unknown dynamical systems [Koc16]. The results include utilizing GPs for providing model predictive control scheme [KMSRG04, KBTK18], adaptive control [CKHV14], tracking control of Euler–Lagrange systems [BKH19], backstepping control

[CH19], control Lyapunov approaches [UPH18a], and feedback linearization schemes [UBKH17] for partially or fully unknown dynamics. However, all these existing works mainly focus on conventional stability or tracking objectives and not on complex logic specifications.

Barrier functions for unknown dynamical systems

On the other hand, approaches based on control barrier functions have shown great potential in solving controller synthesis problems ensuring safety (see [AXGT17, ACE+19, and references therein]) and more complex logic specifications [JSZ20a, YTB19]. Unfortunately, there are very few results available in the literature utilizing notions of control barrier functions for unknown dynamical systems. Assuming a prior knowledge of control barrier functions, the results in [WTE18] and [COMB19] provide the safe online learning of the Gaussian process model and safe learning of the reinforcement learning policy, respectively. In this chapter, we are interested in utilizing both Gaussian processes and control barrier functions for solving the controller synthesis problem against complex logic specifications for unknown nonlinear control systems.

7.1.2 Contributions

To the best of our knowledge, the results presented in this chapter are the first to combine notions of control barrier functions and Gaussian process models to synthesize controllers enforcing complex logic specifications over unknown nonlinear control affine systems. Here, we assume that complex specifications are given as infinite strings over deterministic co-Büchi automata as discussed in Subsection 6.3.1. In order to solve this controller synthesis problem, first, we learn the Gaussian process model from the noisy measurements along with the probabilistic guarantee on the model accuracy. Second, we decompose the given specification to simpler reachability tasks based on automata representing the complements of original co-Büchi automata. Then, we provide a systematic approach to solve those simpler tasks by computing corresponding control barrier functions and associated controllers while considering learned GP along with its corresponding confidence. In the final step, we combine these controllers to obtain a hybrid one together with a lower bound on the probability of satisfying the given specifications. The effectiveness of the proposed results is demonstrated using a jet-engine example.

7.2 Control Affine Systems

We consider nonlinear, continuous-time control affine systems Σ_a defined as

$$\dot{\xi} = \tilde{f}(\xi) + \tilde{g}(\xi)\upsilon, \tag{7.2.1}$$

where $\xi(t) \in X \subset \mathbb{R}^n$ is the state vector evolving in a compact set X and $\upsilon(t) \in U \subseteq \mathbb{R}^m$ is the control input both at time $t \in \mathbb{R}_0^+$. We use $\xi_{x_0\upsilon}(t)$ to denote the value of the trajectory of the system Σ_a starting form initial state x_0 under the input signal υ at time $t \in \mathbb{R}_0^+$. We use $\tilde{f}_j$ to represent the jth component of the vector function $\tilde{f}$, where

$j \in [1; n]$. We assume that function $\tilde{f} : X \to \mathbb{R}^n$ is unknown and function $\tilde{g} : X \to \mathbb{R}^{n\times m}$ is known. We also assume that the unknown function $\tilde{f}$ has low complexity, as measured under the reproducing kernel Hilbert space (RKHS) norm [PR16] as described later in Assumption 7.2.1. The reproducing kernel Hilbert space (RKHS) is a Hilbert space of square integrable functions that includes functions of the form $l(x) = \sum_i \alpha_i \mathsf{k}(x, x_i)$, where $\alpha_i \in \mathbb{R}$, $x, x_i \in X \subset \mathbb{R}^n$, and $\mathsf{k} : X \times X \to \mathbb{R}_0^+$ is a symmetric positive definite function referred to as kernel. The corresponding induced RKHS norm is denoted by $\|l\|_{\mathsf{k}}$. For a detailed discussion on RKHS and RKHS norm, we kindly refer interested readers to [PR16].

Assumption 7.2.1. *The function $\tilde{f}$ in (7.2.1) has bounded RKHS norm with respect to known kernel k, that is $\|\tilde{f}_j\|_{\mathsf{k}} \leq \infty$ for all $j \in [1; n]$.*

Note that, for most of the kernels [WR06] used in practice, an RKHS is dense in the space of continuous functions restricted to a compact domain X. Thus, they can uniformly approximate any continuous function on a compact set X [SKF08]. In addition, we raise the following assumption on the availability of a training set which is essential for any data-driven approach.

Assumption 7.2.2. *We have access to measurements $x \in X$ and $y = \tilde{f}(x) + w$, where $w \sim \mathcal{N}(0_n, \rho_f^2 I_n)$ is an additive noise.*

Note that from a practical point of view, the measurements $\tilde{f}(x)$ (i.e., the derivative) can be obtained approximately using state measurements x by running the system (7.2.1) from different initial conditions and with input signal $\upsilon \equiv 0$. To accommodate the approximation uncertainties, we consider the measurement noise w.

Given a system Σ_a in (7.2.1) satisfying the above assumptions, we are interested in synthesizing a controller enforcing complex specifications given using DCA as describe in Subsection 6.3.1 with some probabilistic guarantee. In general, one needs history dependent policies to enforce such complex specifications. In this chapter, we provide construction of a *hybrid controller*, defined on the augmented space of continuous state set of Σ_a and discrete state set of an automaton that defines the switching mechanism over the set of stationary control policies. Such a hybrid controller is an equivalent representation for a history dependent policy (see Remark 5.6.1).

7.3 Preliminaries

7.3.1 Satisfaction of Specification by Systems

A given system Σ_a in (7.2.1) is connected to the specification given by the accepting language of a DCA $\mathcal{A}$ defined over the set of atomic propositions Π, with the help of a labeling function $L : X \to \Pi$ as described in the next definition which is similar to [WTL15, Definition 2].

Definition 7.3.1. *For a system Σ_a in (7.2.1) and a labeling function $L : X \to \Pi$, an infinite sequence $\sigma(\xi_{x_0\upsilon}) = (\sigma_0, \sigma_1, \ldots) \in \Pi^\omega$ is an infinite trace of the trajectory $\xi_{x_0\upsilon}$ of*

Σ_a if there exists an associated timing sequence $t_0, t_1, \ldots$ such that $t_0 = 0$, $t_s \to \infty$ as $s \to \infty$, and for all $j \in \mathbb{N}$, $t_j \in \mathbb{R}_0^+$, and the following conditions hold

- *$t_j < t_{j+1}$;*
- *$\xi_{x_0 \upsilon}(t_j) \in L^{-1}(\sigma_j)$;*
- *If $\sigma_j \neq \sigma_{j+1}$, then for some $t'_j \in [t_j, t_{j+1}]$, $\xi_{x_0 \upsilon}(t) \in L^{-1}(\sigma_j)$ for all $t \in (t_j, t'_j)$; $\xi_{x_0 \upsilon}(t) \in L^{-1}(\sigma_{j+1})$ for all $t \in (t'_j, t_{j+1})$; and either $\xi_{x_0 \upsilon}(t'_j) \in L^{-1}(\sigma_j)$ or $\xi_{x_0 \upsilon}(t'_j) \in L^{-1}(\sigma_{j+1})$.*

Next, we define the satisfaction of specification given by the language of DCA $\mathcal{A}$.

Definition 7.3.2. *Consider a system Σ_a in* (7.2.1)*, a specification given by the accepting language of a DCA $\mathcal{A}$, and $\sigma(\xi_{x_0 \upsilon})$ (i.e., an infinite trace of trajectory $\xi_{x_0 \upsilon}$) as in Definition 7.3.1. We say that the trajectory of Σ_a starting from initial state $x_0 \in X$ under input signal υ satisfies specification given by $\mathcal{A}$, denoted by $\sigma(\xi_{x_0 \upsilon}) \models \mathcal{A}$, if $\sigma(\xi_{x_0 \upsilon}) \in \mathcal{L}(\mathcal{A})$.*

7.3.2 Problem Definition

The main controller synthesis problem in this chapter is formally defined next.

Problem 7.3.3. *Consider a system Σ_a in* (7.2.1) *with unknown function $\tilde{f}$ and satisfying Assumptions 7.2.1 and 7.2.2, a specification given by the accepting language of a DCA $\mathcal{A} = (Q, Q_0, \Pi, \delta, F)$ over a set of atomic propositions $\Pi = \{p_0, p_1, \ldots, p_M\}$, and a labeling function $L : X \to \Pi$. We aim at computing a hybrid control policy υ (if existing) such that $\sigma(\xi_{x_0 \upsilon}) \models \mathcal{A}$ for all $x_0 \in L^{-1}(p_i)$ and some $i \in [0; M]$ with a (possibly tight) lower bound on the satisfaction probability.*

Finding a solution to Problem 7.3.3 (if existing) is difficult in general. In this chapter, we provide a method that is sound in solving this problem. To construct a hybrid control policy υ, we first model the unknown dynamics using Gaussian processes and then utilize a notion of control barrier functions as discussed in the next two sections.

7.4 Gaussian Process Model

Gaussian processes (GPs) are a non-parametric regression method, where the goal is to find an approximation of a nonlinear map $\tilde{f} : X \to \mathbb{R}^n$. Since $\tilde{f}$ is n-dimensional, each component $\tilde{f}_j$ is approximated with a Gaussian process $\tilde{f}_j^a(x) \sim \mathcal{GP}(\mathsf{m}_j(x), \mathsf{k}_j(x, x'))$, $j \in [1; n]$, where $\mathsf{m}_j : X \to \mathbb{R}$ is a mean function and $\mathsf{k}_j : X \times X \to \mathbb{R}$ is a covariance function (a.k.a., kernel) which measures similarity between any two states $x, x' \in X$. In general, any real-valued function can be used for m_j (it is common practice to set $\mathsf{m}_j(x) = 0$, $\forall x \in X$ and $\forall j \in [1; n]$) and the choice of kernel function is problem dependent, the most commonly used kernels include the linear, squared-exponential,

and Matèrn kernels [WR06]. The approximation of $\tilde{f}$ with n independent GPs is

$$\tilde{f}^a(x) = \begin{cases} \tilde{f}_1^a(x) \sim \mathcal{GP}(0, \mathsf{k}_1(x, x')), \\ \vdots \\ \tilde{f}_n^a(x) \sim \mathcal{GP}(0, \mathsf{k}_n(x, x')). \end{cases} \tag{7.4.1}$$

Given a set of N measurements $\{y^{(1)}, \dots, y^{(N)}\}$ and $\{x^{(1)}, \dots, x^{(N)}\}$, where $y^{(i)} = f(x^{(i)}) + w^{(i)}$ as in Assumption 7.2.2, the posterior distribution corresponding to $\tilde{f}_j(x)$, for $j \in [1; n]$ at an arbitrary state $x \in X$ is computed as a normal distribution $\mathcal{N}(\mu_j(x), \rho_j(x))$ with mean and covariance

$$\mu_j(x) = \overline{\mathsf{k}}_j^T (\mathsf{K}_j + \rho_f^2 I_N)^{-1} y_j, \tag{7.4.2}$$

$$\rho_j^2(x) = \mathsf{k}_j(x, x) - \overline{\mathsf{k}}_j^T (\mathsf{K}_j + \rho_f^2 I_N)^{-1} \overline{\mathsf{k}}_j, \tag{7.4.3}$$

where $\overline{\mathsf{k}}_j = [\mathsf{k}_j(x^{(1)}, x) \cdots \mathsf{k}_j(x^{(N)}, x)]^T \in \mathbb{R}^N$, $y_j = [y_j^{(1)} \cdots y_j^{(N)}]^T \in \mathbb{R}^N$, and

$$\mathsf{K}_j = \begin{bmatrix} \mathsf{k}_j(x^{(1)}, x^{(1)}) & \cdots & \mathsf{k}_j(x^{(1)}, x^{(N)}) \\ \vdots & \ddots & \vdots \\ \mathsf{k}_j(x^{(N)}, x^{(1)}) & \cdots & \mathsf{k}_j(x^{(N)}, x^{(N)}) \end{bmatrix} \in \mathbb{R}^{N \times N}. \tag{7.4.4}$$

Now consider the bound $\overline{\rho}_j^2 = \max_{x \in X} \rho_j^2(x)$. The existence of such bound follows from continuity of kernels. The approximation of overall $\tilde{f}$ can be obtained by concatenating μ_j and ρ_j in (7.4.2) and (7.4.3) as follows

$$\mu(x) := [\mu_1(x), \dots, \mu_n(x)]^T, \tag{7.4.5}$$

$$\rho^2(x) := [\rho_1^2(x), \dots, \rho_n^2(x)]^T. \tag{7.4.6}$$

Considering Assumption 7.2.1, one can upper bound the difference between true value of $\tilde{f}_j(x)$ and inferred mean $\mu_j(x)$ with high probability as given in the next lemma.

Lemma 7.4.1. *Consider a system Σ_a in* (7.2.1) *with Assumptions 7.2.1 and 7.2.2, and the learned Gaussian process model with mean μ_j and standard deviation ρ_j as given in* (7.4.2) *and* (7.4.3)*, respectively. Then, the model error is bounded by*

$$\mathbb{P}\Big\{ \bigcap_{j=1}^{n} \mu_j(x) - \beta_j \rho_j(x) \le \tilde{f}_j(x) \le \mu_j(x) + \beta_j \rho_j(x), \forall x \in X \Big\} \ge (1 - \varepsilon)^n, \tag{7.4.7}$$

with $\varepsilon \in (0, 1)$ and $\beta_j := \sqrt{2\|\tilde{f}_j\|_{\mathsf{k}_j}^2 + 300\gamma_j \log^3(\frac{N+1}{\varepsilon})}$, where N is the number of finite data samples and γ_j is the information gain (see Remark 7.4.2).

Proof. The proof is similar to that of [UPH18b, Lemma 2] and follows from [SKKS12, Theorem 6] by extending the inequality (given for the scalar case) $\mathbb{P}\Big\{\mu_j(x) - \beta_j \rho_j(x) \le \tilde{f}_j(x) \le \mu_j(x) + \beta_j \rho_j(x), \forall x \in X\Big\} \ge (1 - \varepsilon)$ to n-dimensional state-space. □

By using the bound $\overline{\rho}_j$, one can rewrite (7.4.7) as

$$\mathbb{P}\Big\{\tilde{f}(x) \in \{\mu(x) + d \mid d \in \mathcal{D}\}, \forall x \in X\Big\} \geq (1-\varepsilon)^n, \tag{7.4.8}$$

where $\mathcal{D} := \{[d_1, \ldots, d_n]^T \mid d_j \in [-\beta_j\overline{\rho}_j, \beta_j\overline{\rho}_j], j \in [1; n]\}$.

Remark 7.4.2. *The information gain γ_j in Lemma 7.4.1 quantifies the maximum mutual information between a finite set of data-samples and actual function $\tilde{f}_j$. Exact evaluation of γ_j is an NP-hard problem in general, however, it can be greedily approximated and has a sublinear dependency on the number of data-samples N for many commonly used kernels. For the detailed discussion regarding the upper bound on γ_j, we kindly refer the interested readers to [SKKS12].*

7.5 Control Barrier Functions

In this section, we provide sufficient conditions using so-called control barrier functions under which, we can provide the result providing guarantees on reachability specifications. First, we provide the result assuming the availability of full knowledge of the system. Then, we use it to provide the result for unknown systems.

Lemma 7.5.1. *Consider a system Σ_a in* (7.2.1) *and sets X_0, $X_1 \subseteq X$. Suppose there exists a differentiable function $\mathcal{B} : X \to \mathbb{R}$ satisfying following conditions*

$$\forall x \in X_0 \qquad \mathcal{B}(x) \leq 0, \tag{7.5.1}$$

$$\forall x \in X_1 \qquad \mathcal{B}(x) > 0, \tag{7.5.2}$$

$$\forall x \in X\ \exists u \in U \qquad \frac{\partial \mathcal{B}}{\partial x}(x)(\tilde{f}(x) + \tilde{g}(x)u) \leq 0. \tag{7.5.3}$$

Then, for a trajectory $\xi_{x_0\upsilon}$ of system (7.2.1) *starting from any $x_0 \in X_0$ under a stationary control policy υ associated to $\mathcal{B}$ (cf.* (7.5.3)*), one gets $\xi_{x_0\upsilon}(t) \notin X_1\ \forall t \in \mathbb{R}_0^+$.*

Proof. We prove by contradiction. Consider a trajectory $\xi_{x_0\upsilon}$ of Σ_a starting at $x_0 \in X_0$. Suppose $\xi_{x_0\upsilon}$ reaches a state inside X_1. Following (7.5.1) and (7.5.2), one has $\mathcal{B}(\xi_{x_0\upsilon}(0)) \leq 0$ and $\mathcal{B}(\xi_{x_0\upsilon}(t)) > 0$ for some $t \in \mathbb{R}_0^+$. By using inequality (7.5.3), one can conclude that $\mathcal{B}(\xi_{x_0\upsilon}(t)) \leq \mathcal{B}(\xi_{x_0\upsilon}(0)) \leq 0$ for all $t \in \mathbb{R}_0^+$. This contradicts our assumption and concludes the proof. □

The function $\mathcal{B}$ in Lemma 7.5.1 satisfying (7.5.1)-(7.5.3) is usually referred to as a control barrier function.

Remark 7.5.2. *Condition* (7.5.3) *implicitly associates a stationary controller $\mathsf{u} : X \to U$ according to the existential quantifier on u for any $x \in X$. The control policy υ driving the system is readily given by $\upsilon(t) = \mathsf{u}(\xi(t))$, where ξ is the trajectory of the system.*

Now, in order to extend the above result to unknown systems described in Section 7.2, we first learn the Gaussian process model as discussed in Section 7.4. Particularly, the next result provides a probabilistic guarantee on the reachability specification given in Lemma 7.5.1.

Theorem 7.5.3. *Consider a system Σ_a in (7.2.1), the learned Gaussian process model with mean $\mu(\cdot)$ and covariance $\rho^2(\cdot)$ as given in (7.4.5) and (7.4.6), and the result in Lemma 7.4.1. Let X_0, $X_1 \subseteq X$. Suppose there exists a differentiable function $\mathcal{B} : X \to \mathbb{R}$*

$$\forall x \in X_0 \qquad \mathcal{B}(x) \leq 0, \tag{7.5.4}$$

$$\forall x \in X_1 \qquad \mathcal{B}(x) > 0, \tag{7.5.5}$$

$$\forall x \in X\ \exists u \in U\ \forall d \in \mathcal{D} \qquad \frac{\partial \mathcal{B}}{\partial x}(x)(\mu(x) + d + \tilde{g}(x)u) \leq 0, \tag{7.5.6}$$

where $\mathcal{D}$ is the set defined in (2.3.11). Then, for a trajectory $\xi_{x_0\upsilon}$ of (7.2.1) starting from any $x_0 \in X_0$ under a stationary control policy υ associated to $\mathcal{B}$ (cf. condition (7.5.6)), there exists $t \in \mathbb{R}_0^+$ such that $\xi_{x_0\upsilon}(t) \in X_1$ with probability at most $1-(1-\varepsilon)^n$.

Proof. The first two inequalities are equivalent to (7.5.1) and (7.5.2). Now consider condition (7.5.6). Following the result of Lemma 7.4.1, we have $\mathbb{P}\Big\{\tilde{f}(x) \in \{\mu(x) + d \mid d \in \mathcal{D}\}, \forall x \in X\Big\} \geq (1-\varepsilon)^n$ which implies $\mathbb{P}\{\forall x \in X\ \exists u \in U,\ \frac{\partial \mathcal{B}}{\partial x}(x)(\tilde{f}(x) + \tilde{g}(x)u) \leq 0\} \geq (1-\varepsilon)^n$. Let us consider following events: A_1 representing existence of $\mathcal{B}$ satisfying (7.5.4)-(7.5.6), A_2 representing existence of $\mathcal{B}$ satisfying (7.5.1)-(7.5.3), and A_3 representing that $\xi_{x_0\upsilon}(t) \notin X_1\ \forall x_0 \in X_0\ \forall t \in \mathbb{R}_0^+$. Now, $\mathbb{P}\{A_1 \implies A_3\} = \mathbb{P}\{(A_1 \implies A_2) \text{ and } (A_2 \implies A_3)\} = \mathbb{P}\{A_1 \implies A_2\}\mathbb{P}\{A_2 \implies A_3\} \geq (1-\varepsilon)^n$. The last equality follows from the fact that the events $A_1 \implies A_2$ and $A_2 \implies A_3$ are mutually independent. The last inequality is obtained by using $\mathbb{P}\{A_2 \implies A_3\} = 1$ which follows from Lemma 7.5.1 and $\mathbb{P}\{A_1 \implies A_2\} \geq (1-\varepsilon)^n$ showed in the first part of the proof. Thus, the existence of $\mathcal{B}$ satisfying (7.5.4)-(7.5.6) implies that for any $x_0 \in X_0$, one has $\xi_{x_0\upsilon}(t) \notin X_1$ for all $t \in \mathbb{R}_0^+$ with probability at least $(1-\varepsilon)^n$ which implies that there exists $t \in \mathbb{R}_0^+$ such that $\xi_{x_0\upsilon}(t) \in X_1$ with probability at most $1-(1-\varepsilon)^n$. This concludes the proof. □

Note that condition (7.5.6) in Theorem 7.5.3 associates a stationary control policy υ according to the existential quantifier on u which provides input trajectory υ given state trajectory ξ and for any arbitrary choice of $d \in \mathcal{D}$. In the next section, we discuss how to translate synthesis Problem 7.3.3 for any specification given using DCA $\mathcal{A}$ into a computation of a collection of control barrier functions each satisfying conditions in Theorem 7.5.3.

7.6 Controller Synthesis using Control Barrier Functions

In order to synthesize control policies using control barrier functions enforcing specifications expressed by DCAs $\mathcal{A}$, we consider the decomposition of specifications into sequential reachabilities as described in Subsection 6.5.1.

7.6.1 Hybrid Control Policy

Consider the set of state runs of length 3, $\overline{\mathcal{P}}^p(\overline{\mathsf{q}})$, as defined in (6.5.3). Now we provide a systematic approach to compute a control policy such that the trajectories of Σ_a

satisfy the specification expressed by DCA $\mathcal{A}$ with a high probability. Given DBA $\mathcal{A}^c$, our approach relies on performing the computation of control barrier functions for each element of $\overline{\mathcal{P}}(\mathcal{A}^c)$, which at the end provides control policies ensuring that we do not have accepting runs in the complement of the given specification (i.e., DCA $\mathcal{A}$) with a high probability. To provide the result on the construction of control policies solving Problem 6.3.4, we provide the following lemma which is a direct consequence of results in Theorem 6.4.2 and, hence, provided without a proof.

Lemma 7.6.1. *For a triplet $(q, q', q'') \in \overline{\mathcal{P}}(\mathcal{A}^c)$, if there exists a control barrier function with a corresponding stationary control policy υ satisfying conditions* (7.5.4)-(7.5.6) *in Theorem 6.4.2 with $X_0 = L^{-1}(\sigma(q, q'))$ and $X_1 = L^{-1}(\sigma(q', q''))$, then the trajectory $\xi_{x_0\upsilon}$ of Σ_a starting from any initial state $x_0 \in X_0$ under policy υ reaches X_1 with a probability at most $1 - (1 - \varepsilon)^n$.*

Lemma 7.6.1 uses control barrier functions along with appropriate choices of stationary control policies υ to provide reachability probabilities for elements in $\overline{\mathcal{P}}(\mathcal{A}^c)$ as mentioned in Theorem 6.4.2. Following discussion in Subsection 6.5.1, in order to avoid ambiguity while utilizing controller we partition $\overline{\mathcal{P}}(\mathcal{A}^c)$ and put the elements sharing a common control barrier function and a corresponding controller in the same partition set. These sets can be formally defined as: $\phi_{(q,q',\Delta(q'))} := \{(q, q', q'') \in \overline{\mathcal{P}}(\mathcal{A}^c) \mid q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\}$. The control barrier function and the stationary controller corresponding to the partition set $\phi_{(q,q',\Delta(q'))}$ are denoted by $\mathcal{B}_{\phi_{(q,q',\Delta(q'))}}(x)$ and $\mathsf{u}_{\phi_{(q,q',\Delta(q'))}}(x)$, respectively. Thus, for all $\eta \in \overline{\mathcal{P}}(\mathcal{A}^c)$, we have

$$\mathcal{B}_\eta(x) = \mathcal{B}_{\phi_{(q,q',\Delta(q'))}}(x) \text{ and } \mathsf{u}_\eta(x) = \mathsf{u}_{\phi_{(q,q',\Delta(q'))}}(x), \quad \text{if } \eta \in \phi_{(q,q',\Delta(q'))}. \tag{7.6.1}$$

As describe in Subsection 6.5.2, the hybrid controller defined over augmented state-space $X \times Q_{\mathfrak{m}}$, where $Q_{\mathfrak{m}}$ is the set of states of switching automaton (see Subsection 6.5.2), solving Problem 7.3.3 is given by

$$\tilde{\mathsf{u}}(x, q_{\mathfrak{m}}) = \mathsf{u}_{\phi_{(q'_{\mathfrak{m}})}}(x), \quad \forall (q_{\mathfrak{m}}, L(x), q'_{\mathfrak{m}}) \in \delta_{\mathfrak{m}}. \tag{7.6.2}$$

The corresponding hybrid control policy υ is given by $\upsilon(t) = \tilde{\mathsf{u}}(\xi(t), q_{\mathfrak{m}})$.

7.6.2 Computation of Satisfaction Probability

The next theorem provides an upper bound on the probability that the state trajectory of the system Σ_a satisfies the accepting language of DBA $\mathcal{A}^c$.

Theorem 7.6.2. *For a given DCA $\mathcal{A}$ representing specification, let $\mathcal{A}^c$ be a DBA accepting complement of the language of $\mathcal{A}$. For $p \in \Pi$, let $\overline{\mathcal{R}}^p$ and $\overline{\mathcal{P}}^p$ be the sets defined in* (5.5.1) *and* (5.5.3)*, respectively. The probability that the state trajectory $\xi_{x_0\upsilon}$ of Σ_a starting from any initial state $x_0 \in L^{-1}(p)$ under the hybrid control policy υ associated with the hybrid controller* (5.6.2) *accepts the language of DBA $\mathcal{A}^c$ is upper bounded by*

$$\mathbb{P}\{\sigma(\xi_{x_0\upsilon}) \models \mathcal{A}^c\} \leq \sum_{\overline{\mathsf{q}} \in \overline{\mathcal{R}}^p} (1 - (1 - \varepsilon)^n)^{|\tilde{\mathcal{P}}^p(\overline{\mathsf{q}})|}, \tag{7.6.3}$$

where $\tilde{\mathcal{P}}^p(\overline{\mathsf{q}})$ represents the set of triplets in $\overline{\mathcal{P}}^p(\overline{\mathsf{q}})$ for which there exist control barrier functions as given in Theorem 6.4.2 and $1-(1-\varepsilon)^n$ is the upper bound on the reachability probability associated with the triplet $(q,q',q'') \in \tilde{\mathcal{P}}^p(\overline{\mathsf{q}})$ as defined in Lemma 7.6.1.

Proof. Consider $p \in \Pi$ and an accepting state run $\mathsf{q}=(q_0^r, q_1^r, \ldots, q_{\mathsf{m}_r}^r(q_0^s, q_1^s, \ldots, q_{\mathsf{m}_s}^s)^\omega) \in Q^\omega$ in $\mathcal{A}^c$ with $\sigma(q_0^r, q_1^r) = p$. Let the corresponding finite state run be $\overline{\mathsf{q}} \in \overline{\mathcal{R}}^p$ and consider the set $\overline{\mathcal{P}}^p(\overline{\mathsf{q}})$ as defined in Subsection 6.5.1. We apply Lemma 7.6.1 to any triplet $\eta = (q,q',q'') \in \overline{\mathcal{P}}^p(\overline{\mathsf{q}})$. The probability that the trajectory of the system Σ_a starts from any initial state in $x_0 \in L^{-1}(\sigma(q,q'))$ and reaches $L^{-1}(\sigma(q',q''))$ using controller $\mathsf{u}_\eta(x)$ is upper bounded by $1-(1-\varepsilon)^n$. Consider a set $\tilde{\mathcal{P}}^p(\overline{\mathsf{q}})$ containing triplets in $\overline{\mathcal{P}}^p(\overline{\mathsf{q}})$ for which there exist control barrier functions. Now the upper bound on the probability of the trace of the trajectories (i.e., $\sigma(\xi_{x_0\upsilon})$) having accepting run corresponding to q is in $\mathcal{L}(\mathcal{A}^c)$ is given by the product of the probability bounds corresponding to all elements $(q,q',q'') \in \tilde{\mathcal{P}}^p(\overline{\mathsf{q}})$ and is given by $\mathbb{P}\{\sigma(\mathsf{q}) \in \mathcal{L}(\mathcal{A}^c)\} \leq (1-(1-\varepsilon)^n)^{|\tilde{\mathcal{P}}^p(\overline{\mathsf{q}})|}$.

The upper bound on the probability that the state trajectory of Σ_a starting from any initial state $x_0 \in L^{-1}(p)$ under a hybrid control policy υ satisfying specification $\mathcal{A}^c$ is computed by summing the probability bounds for all possible accepting runs in the set $\overline{\mathcal{R}}^p$ which is given by $\mathbb{P}\{\sigma(\xi_{x_0\upsilon}) \models \mathcal{A}^c\} \leq \sum_{\overline{\mathsf{q}} \in \overline{\mathcal{R}}^p}(1-(1-\varepsilon)^n)^{|\tilde{\mathcal{P}}^p(\overline{\mathsf{q}})|}$. □

Remark 7.6.3. *If one does not find a control barrier function for a particular triplet $(q,q',q'') \in \overline{\mathcal{P}}(\mathcal{A}^c)$, one can choose an arbitrary control input and choose pessimistic probability bound* 1.

Remark 7.6.4. *For any $(q,q',q'') \in \phi_{(q,q',\Delta(q'))}$, if $L^{-1}(\sigma(q,q')) \cap L^{-1}(\sigma(q',q'')) \neq \emptyset$, there exists no control barrier function satisfying conditions in Proposition 6.4.2. This follows directly due to the conflict in conditions* (7.5.4) *and* (7.5.5)*. For example consider the triplet $(q_4,q_3,q_3) \in \overline{\mathcal{P}}^{p_2}(q_0,q_4,q_3)$ in Example 1. There, we have $L^{-1}(p_1) \cap L^{-1}(\top) = L^{-1}(p_1) \neq \emptyset$, so there is no need to search for a control barrier function in this case since there is none.*

Corollary 7.6.5. *Given the result of Theorem 7.6.2, the probability that the trajectories of Σ_a starting from any $x_0 \in L^{-1}(p)$ under the hybrid control policy υ satisfy the specification given by DCA $\mathcal{A}$ is lower bounded by $\mathbb{P}\{\sigma(\xi_{x_0\upsilon}) \models \mathcal{A}\} \geq 1 - \mathbb{P}\{\sigma(\xi_{x_0\upsilon}) \models \mathcal{A}^c\}$.*

7.6.3 Computation of Control Barrier Functions

One can utilize the CEGIS approach (as discussed in Section 5.7.2)to search for control barrier functions solving feasibility condition given in the next Lemma.

Lemma 7.6.6. *Consider a control affine system Σ_a with a finite input set $U = \{u_1, u_2, \ldots, u_l\}$, where $u_i \in \mathbb{R}^m$, $i \in [1;l]$. Suppose there exists a function $\mathcal{B}(x)$ such that the following expression is true*

$$\bigwedge_{x \in X_0} B(x) \leq 0 \bigwedge_{x \in X_1} \mathcal{B}(x) > 0 \bigwedge_{x \in X} \Big(\bigvee_{u \in U} \Big(\bigwedge_{d \in \mathcal{D}} \big(\frac{\partial \mathcal{B}}{\partial x}(\mu(x) + d + \tilde{g}(x)u) \leq 0\big)\Big)\Big). \tag{7.6.4}$$

Then, $\mathcal{B}(x)$ satisfies conditions (7.5.4)-(7.5.6) in Theorem 6.4.2 and any $\mathsf{u} : X \to U$ with $\mathsf{u}(x) := \{u \in U \mid \frac{\partial \mathcal{B}}{\partial x}(\mu(x) + d + \tilde{g}(x)u) \leq 0\}$ for any arbitrary choice of $d \in \mathcal{D}$ is a corresponding stationary controller.

7.7 Case Study

We consider the nonlinear Moore-Greitzer jet engine model in no-stall mode [ZPMT11] as a case study. Consider the unknown nonlinear dynamics given as

$$\tilde{f}(x) = \begin{bmatrix} \tilde{f}_1(x) \\ \tilde{f}_2(x) \end{bmatrix} = \begin{bmatrix} -x_2 - \frac{3}{2}x_1^2 - \frac{1}{2}x_1^3 \\ x_1 \end{bmatrix} \quad \text{and} \quad \tilde{g}(x) = \begin{bmatrix} 0 \\ -1 \end{bmatrix},$$

where $x = [x_1, x_2]^T$, $x_1 = \Phi - 1$, $x_2 = \Psi - \psi - 2$, Φ is the mass flow, Ψ is the pressure rise, and ψ is a constant. The control input $u \in U = \{-2, -1.5, -1, -0.5, 0, 0.5, 1, 1.5, 2\}$. We consider a compact state-space $X = [-1, 3] \times [-4, 4]$ and regions of interest $X_0 = [0, 1] \times [-1, 1]$, $X_1 = [-1, 0] \times [-4, -2.5]$, $X_2 = [-1, 3] \times [2, 4]$ and $X_3 = X \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_j) = p_j$ for all $x_j \in X_j$, $j \in \{0, 1, 2, 3\}$. Note that functions f_1 and f_2 are continuous thus they satisfy Assumption 7.2.1 [SKKS12]. The objective here is to compute a control policy ensuring satisfaction of a specification given by the accepting language of the DCA $\mathcal{A}$ in Figure 7.1 with high confidence. In English, language of $\mathcal{A}$ entails that if the system starts in X_0 it will always stay away from X_1 or X_2. Note that, the corresponding DBA $\mathcal{A}^c$ accepting complement of $\mathcal{L}(\mathcal{A})$ has exactly the same structure as in Figure 7.1, but with the Büchi accepting condition. One can readily see that, we have sets $\overline{\mathcal{P}}^{p_0} = \{(q_0, q_1, q_2), (q_1, q_2, q_2)\}$ and $\overline{\mathcal{P}}^{p_1} = \overline{\mathcal{P}}^{p_2} = \overline{\mathcal{P}}^{p_3} = \{(q_0, q_2, q_2)\}$. Following Remark 6.5.4, we only need to compute a control barrier function corresponding to the triplet (q_0, q_1, q_2). In order to synthesize a control policy enforcing the aforementioned specification, we first learn the unknown model using Gaussian processes. For learning Gaussian process model, we collected 25 data samples of x and $y = \tilde{f}(x) + w$, where $w \sim \mathcal{N}(0, \rho_f^2 I_n)$, $\rho_f = 0.01$, by simulating the system with several initial states chosen randomly using uniform distribution. We used squared-exponential kernel [WR06] defined as $k_j(x, x') = \rho_{k_j}^2 \exp\left(\sum_{i=1}^{2} \frac{(x_i - x_i')^2}{-2l_{ji}^2}\right), j \in \{1, 2\}$, where $\rho_{k_1}^2 = 224.4168$ and $\rho_{k_2}^2 = 24.5311$ are signal variances and $l_{11} = 6.6030$, $l_{12} = 327.5503$, $l_{21} = 42.1995$,

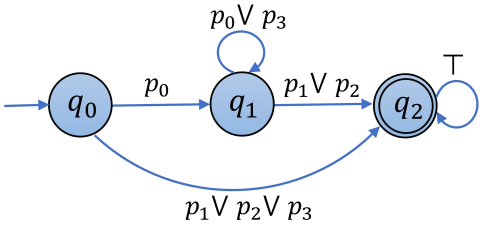

Figure 7.1: DCA $\mathcal{A}$ representing specification.

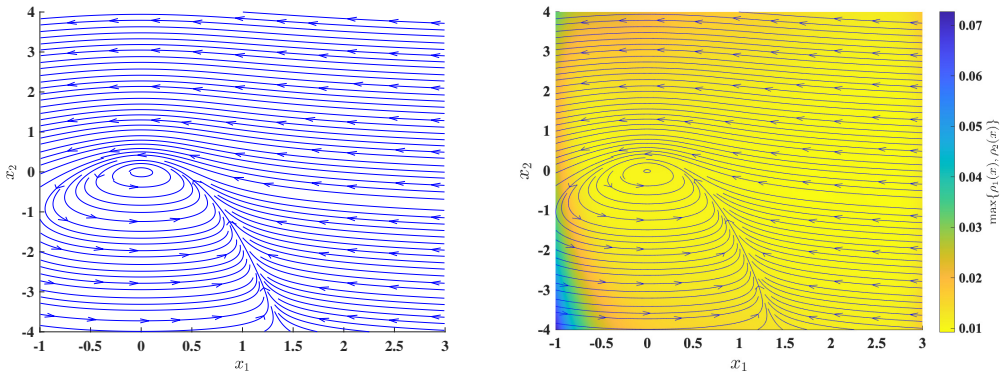

Figure 7.2: Illustration of the vector field learned using GPs: (left) vector field $\tilde{f}(x)$ of the original system, (right) learned vector field $\mu(x)$ with the maximum standard deviation shown using colormap.

and $l_{22} = 6.4648 \times 10^6$ are lengthscales. These parameters are obtained through likelihood maximization using a quasi-Newton method. The inferred mean and variance are represented as in (7.4.5) and (7.4.6) with $\max\{\overline{\rho}_1, \overline{\rho}_2\} = 0.0727$. Figure 7.2 illustrates the actual and the learned vector fields. Computing $\|\tilde{f}_j\|_{k_j}$ and γ_j, $j \in \{1, 2\}$, is a hard problem in general. Thus, we employ Monte-Carlo approach to obtain the probability bound on the accuracy of the learned model provided in Lemma 7.4.1. For a choice of $\beta_1 = \beta_2 = 1$, we obtained a probability interval for the probability in (2.3.11) as $\mathbb{P}\Big\{\tilde{f}(x) \in \{\mu(x) + d \mid d \in \mathcal{D}\}, \forall x \in X\Big\} \in [0.9937, 0.9975]$, where $\mathcal{D} = [-0.0727, 0.0727]^2$ with confidence $1 - 10^{-10}$ using 10^6 realizations. Thus, one can choose the lower bound $(1 - \epsilon)^2$ as 0.9937. Note that for a fix error bound, if we increase the number of data samples N used for learning GPs, we get the tighter lower bound on the probability in (2.3.11). Figure 7.3 shows the effect of the increase in number of data samples N on the lower bound on the probability (top) and the maximum standard deviation $\overline{\rho}_{\max}$ (bottom). Next, we compute a control barrier function of order 2 using CEGIS approach discussed in Subsection 7.6.3 as the following: $\mathcal{B}(x) = -4292.8910 + 1129.2414x_1 + 1010.3266x_2 + 1274.3322x_1^2 + 1564.8195x_2^2 - 1368.6064x_1x_2$. The corresponding controller is given by

$$\mathsf{u}(x) = \min\{u \in U \mid \frac{\partial \mathcal{B}}{\partial x}(\mu(x) + d + \tilde{g}(x)u) \leq 0\}, \tag{7.7.1}$$

for an arbitrarily chosen $d \in [-0.0727, 0.0727]^2$. Using Corollary 7.6.5, one can have a lower bound on the probability that the trajectories of the system starting from any initial state $x_0 \in X_0$ under the control policy υ in (7.7.1) satisfy the specification given by the DCA $\mathcal{A}$ in Figure 7.1 as $\mathbb{P}\{\sigma(\xi_{x_0\upsilon}) \models \mathcal{A}\} \geq 0.9937$. One can readily see that only one stationary control policy is enough for enforcing specification, thus, one does not need a switching mechanism. The closed-loop trajectories using controller (7.7.1) starting from several initial conditions in X_0 are shown in Figure 7.4.

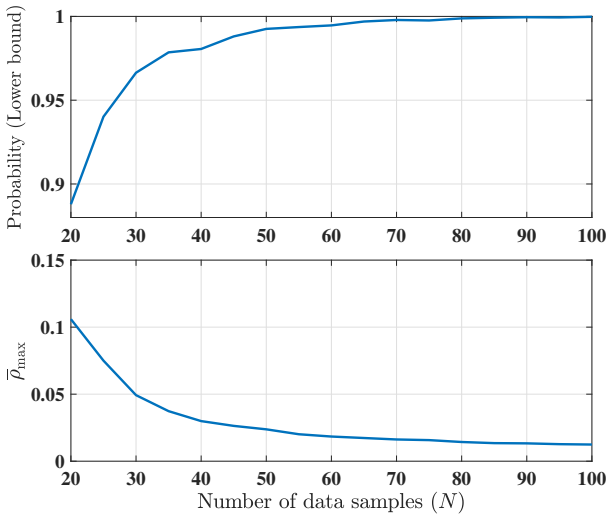

Figure 7.3: The change in lower bound on the probability (top) and the maximum standard deviation $\overline{\rho}_{\max}$ (bottom) with increase in the number of data samples N.

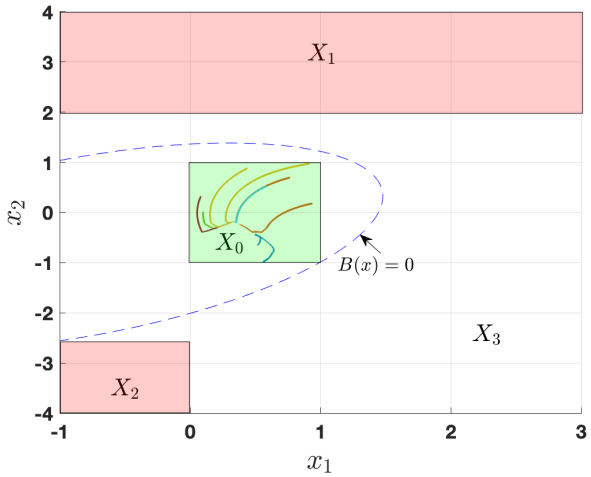

Figure 7.4: The solid colored lines are closed-loop trajectories starting from several initial states in X_0. The dashed curve shows the zero level set of $\mathcal{B}(x)$, defined as $\{x \in X \mid \mathcal{B}(x) = 0\}$ and the regions of interest are shown using colored rectangles.

8 Conclusions and Future Directions

8.1 Summary

In this thesis, we discussed the synthesis of controllers for complex dynamical systems enforcing complex logic specifications expressed as linear temporal logic formulae or (in)finite traces on automata. In particular, to alleviate the issue of the curse of dimensionality that arises in conventional approaches, we developed controller synthesis techniques without state-space discretization. In addition, we considered various complexities such as noisy dynamics, dependency on state history (or delays in the dynamics), lack of knowledge of the precise mathematical model, and interconnection between subsystems. In the remainder, we summarize the content of the thesis.

Chapter 2 introduced a coordinate invariant notion of incremental stability for stochastic control systems with jumps and provided a feedback controller design approach based on backstepping scheme providing controllers together with the corresponding incremental Lyapunov functions enforcing a class of stochastic control systems, namely, stochastic Hamiltonian systems with jumps, incrementally stable. It also introduced a notion of incremental input-to-state stability for retarded jump-diffusion systems and provided sufficient conditions for it in terms of the existence of a notion of incremental Lyapunov functions.

Considering the incremental stability results of Chapter 2, Chapter 3 provided a construction of finite abstractions which are approximately bisimilar to the corresponding non-stochastic version of retarded jump-diffusion systems. Then, under some mild assumptions over incremental Lyapunov functions, we obtained a lower bound on the probability such that the distance between output trajectories of the obtained finite abstraction and those of the original retarded jump-diffusion system remains close over a finite time horizon. One can leverage the proposed probability closeness to synthesize a control policy using constructed finite abstractions and refine it back to the original system while providing a guarantee on the probability of satisfaction over the original system.

Chapter 4 presented QUEST, a new open-source tool for automated controller synthesis of incrementally input-to-state stable nonlinear control systems leveraging state-space discretization-free finite abstraction based approaches.

Chapter 5 proposed alternative discretization-free approaches based on control barrier functions for stochastic control systems. In particular, the approach computes a hybrid control policy together with a lower bound on the probability of satisfying a specification encoded as LTL over finite traces. It utilizes computation of control barrier functions

and uses sum-of-squares (SOS) optimization or counter-example guided inductive synthesis (CEGIS) to obtain such policies.

In Chapter 6, we proposed a scheme for designing decentralized (or distributed) hybrid control policies for interconnected discrete-time control systems enforcing specifications expressed using a language of deterministic co-Büchi automata. We utilized a small-gain type condition to provide a construction of control barrier functions by composing so-called local control barrier functions computed for subsystems. In other words, the compositional approach proposed in Chapter 6 helps to alleviate the problem of scalability resulting from searching control barrier functions via sum-of-squares (SOS) optimization and counter-example guided inductive synthesis approaches proposed in Chapter 5.

Motivating from the fact that the precise mathematical model is difficult to obtain for many real-world complex systems, Chapter 7 proposed a scheme for designing hybrid control policies for such unknown nonlinear control affine systems enforcing specifications expressed by deterministic co-Büchi automata with some formal probabilistic guarantee. In particular, we utilized control barrier functions computed for approximate models learned through Gaussian processes with high confidence. In the end, we provided data-dependent lower bounds on the probabilities of satisfying the considered specifications for original unknown systems.

8.2 Recommendations for Future Research

In this section, we discuss some interesting topics that ought to consider as future research lines.

Incremental stability

The result on designing controllers rendering stochastic control systems incrementally stable provided in Chapter 2 is currently limited to a class of systems, namely, Hamiltonian stochastic control systems with jumps (which are mostly limited to model mechanical or electrical systems). Considering the usefulness of incremental stability property (see Section 2.1), it is a promising direction to investigate the design of controllers enforcing this property for a class of stochastic control systems represented by more general stochastic differential equations. The results in Chapter 3 are based on the assumption that a class of infinite-dimensional systems, namely, retarded jump-diffusion systems are incrementally stable, however, not all the real world systems are incrementally stable. This necessitates the need for the synthesis of controllers that enforce the incremental stability property for retarded jump-diffusion systems. One could also consider a controller synthesis and also the characterization of incremental stability for more general infinite-dimensional systems such as the systems represented using partial differential equations.

State-space discretization-free abstractions

The results proposed in Chapter 3 provide a probabilistic guarantee on the closeness of trajectories of a class of infinite-dimensional stochastic systems, namely, retarded jump-diffusion systems and their finite abstractions over a finite-time horizon. More investigation is needed to extend those results for providing a guarantee over an infinite-time horizon.

Improving QUEST

In our proposed software tool, QUEST, in Chapter 4, we utilized BDDs as an underlying data structure which is good for compact representations of abstractions and obtained controllers, but it is found to be slower compared to other data structures used in some of the existing state-space discretization based tools such as hash tables in CoSyMa [MGG13] and sparse matrices in SCOTS [RZ16b]. On the other hand, most of the operations in QUEST are highly parallelizable, so one can leverage parallel computations. By utilizing these improvement options, there is a lot of scope for improving the speed of QUEST. In addition, the current implementation is only capable of handling invariance and reachability specifications which can be further generalized to more complex LTL specifications.

Synthesis via control barrier functions

In chapter 5, we provided a controller synthesis technique for discrete-time stochastic control systems enforcing a fragment of temporal logic specifications, namely, LTL over finite traces (LTL_F). Note that the approach can easily be extended to synthesize policies for continuous-time stochastic control systems enforcing LTL_F specifications by excluding the next operator with some minor modifications (see our paper [AJZ19] for similar modifications). However, the results may become more conservative in this case since an efficient computation of the temporal horizon $T(\cdot)$ as in (5.5.3) is not possible and one needs to consider the worst-case $T = N$ for each safety task. It is also interesting to extend similar results for different classes of systems such as retarded (stochastic) systems and switched (stochastic) systems. In addition, the chapter also provides two approaches to search for control barrier functions under some suitable assumptions which are found to be computationally expensive for large systems. More investigation is needed to develop efficient techniques for computing control barrier functions. To address the issue of computational overload for large-scale systems, in Chapter 6, we provided the compositional framework for the construction of control barrier functions assuming small-gain type conditions. However, the current results are limited to non-stochastic discrete-time control systems. One could extend these results for continuous-time control systems and for stochastic control systems. One can also employ and study the effectiveness of other existing compositionality approaches such as dissipativity. All the proposed results in the thesis assume the availability of full-state measurements. It is also an interesting direction for research to develop control barrier functions based synthesis approaches for the systems considering partial-information of states. One can refer to our recent

works [JJZ20, JJZ21] for similar results, where we provide a synthesis of controllers for partially-observed jump-diffusion systems using control barrier functions.

Synthesis for unknown dynamical systems

Chapter 7 utilized a data-driven approach based on Gaussian processes (GPs) to learn the unknown model with a probabilistic guarantee on accuracy. The learned model is then used to synthesize controller enforcing complex logic specifications given by deterministic co-Büchi automata with some probabilistic guarantee via barrier functions. However, learning GPs need some smoothness assumption over unknown dynamics given by a bound over RKHS norm (See Assumption 7.2.1) which is in general difficult to compute for unknown systems. The recent results on providing performance bounds for the scenario approach [ESL14] and data-driven stability analysis of black-box switched linear systems [KBJT19] motivate another potential direction that will allow us to learn control barrier functions directly from data without learning approximate unknown dynamics.

Bibliography

[ACE+19] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada. Control barrier functions: Theory and applications. In *18th European Control Conference (ECC)*, pages 3420–3431, June 2019.

[AJJ+20] P. Ashok, M. Jackermeier, P. Jagtap, J. Křetínský, M. Weininger, and M. Zamani. Dtcontrol: Decision tree learning algorithms for controller representation. New York, NY, USA, 2020. Association for Computing Machinery.

[AJZ19] M. Anand, P. Jagtap, and M. Zamani. Verification of switched stochastic systems via barrier certificates. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 4373–4378. IEEE, 2019.

[AKM17] M. Althoff, M. Koschi, and S. Manzinger. CommonRoad: Composable benchmarks for motion planning on roads. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 719 – 726, 2017.

[Ang02] D. Angeli. A Lyapunov approach to incremental stability properties. *IEEE Transactions on Automatic Control*, 47(3):410–421, March 2002.

[APLS08] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.

[AS03] J. S. Aujla and F. C. Silva. Weak majorization inequalities and convex functions. *Linear Algebra and Its Applications*, 369:217–233, August 2003.

[AXGT17] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8):3861–3876, 2017.

[BD18] A. Bisoffi and D. V. Dimarogonas. A hybrid barrier certificate approach to satisfy linear temporal logic specifications. In *2018 Annual American Control Conference (ACC)*, pages 634–639. IEEE, 2018.

[BKH19] T. Beckers, D. Kulić, and S. Hirche. Stable Gaussian process based tracking control of Euler–Lagrange systems. *Automatica*, 103:390–397, 2019.

[BKL08] C. Baier, J-P. Katoen, and K. G. Larsen. *Principles of model checking*. MIT press, 2008.

[BML+10] B. N. Bond, Z. Mahmood, Y. Li, R. Sredojevic, A. Megretski, V. Stojanovi, Y. Avniel, and L. Daniel. Compact modeling of nonlinear analog circuits using system identification via semidefinite programming and incremental stability certification. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 29(8):1149–1162, August 2010.

[Bry92] R. E. Bryant. Symbolic boolean manipulation with ordered binary-decision diagrams. *ACM Computing Surveys (CSUR)*, 24(3):293–318, 1992.

[BS96] D. P. Bertsekas and S. E. Shreve. *Stochastic Optimal Control: The Discrete-Time Case*. Athena Scientific, 1996.

[BSP08] M. Bandyopadhyay, T. Saha, and R. Pal. Deterministic and stochastic analysis of a delayed allelopathic phytoplankton model within fluctuating environment. *Nonlinear Analysis: Hybrid systems*, 2(3):958–970, May 2008.

[BYG17] C. Belta, B. Yordanov, and E. A. Gol. *Formal methods for discrete-time dynamical systems*, volume 89. Springer, 2017.

[CGSS13] A. Cimatti, A. Griggio, B. J. Schaafsma, and R. Sebastiani. The MATHSAT5 SMT solver. In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 93–107. Springer, 2013.

[CH19] A. Capone and S. Hirche. Backstepping for partially unknown nonlinear systems using Gaussian processes. *IEEE Control Systems Letters*, 3(2):416–421, 2019.

[CKHV14] G. Chowdhary, H. A. Kingravi, J. P. How, and P. A. Vela. Bayesian nonparametric adaptive control using Gaussian processes. *IEEE transactions on neural networks and learning systems*, 26(3):537–550, 2014.

[COMB19] R. Cheng, G. Orosz, R. M. Murray, and J. W. Burdick. End-to-end safe reinforcement learning through barrier functions for safety-critical continuous control tasks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3387–3395, 2019.

[CPR13] A. Chaillet, A. Y. Pogromsky, and B. S. Rüffer. A Razumikhin approach for the incremental stability of delayed nonlinear systems. In *Proceedings of 52nd IEEE Conference on Decision and Control*, pages 1596–1601, December 2013.

[DB10] F. Dörfler and F. Bullo. Synchronization and transient stability in power networks and non-uniform Kuramoto oscillators. In *Proceedings of the 2010 American Control Conference*, pages 930–937, June 2010.

[DEK07] C. Dax, J. Eisinger, and F. Klaedtke. Mechanizing the powerset construction for restricted classes of ω-automata. In *International Symposium*

on Automated Technology for Verification and Analysis, pages 223–236. Springer, 2007.

[DGV13] G. De Giacomo and M. Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *International Joint Conference on Artificial Intelligence*, volume 13, pages 854–860, 2013.

[DGV15] G. De Giacomo and M. Y. Vardi. Synthesis for LTL and LDL on finite traces. In *International Joint Conference on Artificial Intelligence*, volume 15, pages 1558–1564, 2015.

[DLLF+16] A. Duret-Lutz, A. Lewkowicz, A. Fauchille, T. Michaud, E. Renault, and L. Xu. Spot 2.0: A framework for LTL and ω-automata manipulation. In *International Symposium on Automated Technology for Verification and Analysis*, pages 122–129. Springer, 2016.

[dMB08] L. de Moura and N. Bjørner. Z3: An efficient SMT solver. In C. R. Ramakrishnan and J. Rehof, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[DRW07] S. Dashkovskiy, B. Rüffer, and F. Wirth. An ISS small gain theorem for general networks. *"Mathematics of Control, Signals, and Systems*, 19(2):93–122, May 2007.

[DRW10] S. Dashkovskiy, B. Rüffer, and F. Wirth. Small gain theorems for large scale systems and construction of ISS Lyapunov functions. *SIAM Journal on Control and Optimization*, 48(6):4089–4118, 2010.

[EA87] I. V. Evstigneev and V. I. Arkin. *Stochastic models of control and economic dynamics*. Academic Press, Ltd., United Kingdom, 1987.

[Eav72] B. C. Eaves. Homotopies for computation of fixed points. *Mathematical Programming*, 3(1):1–22, 1972.

[EK91] G. B. Ermentrout and N. Kopell. Multiple pulse interactions and averaging in systems of coupled neural oscillators. *Journal of Mathematical Biology*, 29(3):195–217, Jan 1991.

[ESL14] P. M. Esfahani, T. Sutter, and J. Lygeros. Performance bounds for the scenario approach and an extension to a class of non-convex programs. *IEEE Transactions on Automatic Control*, 60(1):46–58, 2014.

[FJKG10] C. Finucane, G. Jing, and H. Kress-Gazit. LTLMoP: Experimenting with language, temporal logic and robot control. In *International Conference on Intelligent Robots and Systems (IROS)*, pages 1988–1993. IEEE, 2010.

[FMPS18] S. S. Farahani, R. Majumdar, V. S. Prabhu, and S. Soudjani. Shrinking horizon model predictive control with signal temporal logic constraints under stochastic disturbances. *IEEE Transactions on Automatic Control*, 64(8):3324–3331, 2018.

[GB94] L. El Ghaoui and V. Balakrishnan. Synthesis of fixed-structure controllers via numerical optimization. In *Proceedings of 1994 33rd IEEE Conference on Decision and Control*, volume 3, pages 2678–2683, December 1994.

[Gir14] A. Girard. Approximately bisimilar abstractions of incrementally stable finite or infinite dimensional systems. In *53rd IEEE Conference on Decision and Control*, pages 824–829. IEEE, 2014.

[GKC13] S. Gao, S. Kong, and E. M. Clarke. dReal: An SMT solver for nonlinear theories over the reals. In *International Conference on Automated Deduction*, pages 208–214. Springer, 2013.

[GP07] A. Girard and G. J. Pappas. Approximation metrics for discrete and continuous systems. *IEEE Transactions on Automatic Control*, 52(5):782–798, May 2007.

[GPT09] A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Transactions on Automatic Control*, 55(1):116–126, 2009.

[HCL+17] C. Huang, X. Chen, W. Lin, Z. Yang, and X. Li. Probabilistic safety verification of stochastic hybrid systems using barrier certificates. *ACM Transactions on Embedded Computing Systems*, 16(5s):186, 2017.

[HJJ+95] J. G. Henriksen, J. Jensen, M. Jørgensen, N. Klarlund, R. Paige, T. Rauhe, and A. Sandholm. MONA: Monadic second-order logic in practice. In *International Workshop on Tools and Algorithms for the Construction and Analysis of Systems*, pages 89–110. Springer, 1995.

[HLL96] O. Hernández-Lerma and J. B. Lasserre. *Discrete-time Markov control processes*, volume 30 of *Applications of Mathematics*. Springer, 1996.

[HM09] L. Huang and X. Mao. On input-to-state stability of stochastic retarded systems with Markovian switching. *IEEE Transactions on Automatic Control*, 54(8):1898–1902, August 2009.

[HSSG12] A. Hamadeh, G. B. Stan, R. Sepulchre, and J. Goncalves. Global state synchronization in networks of cyclic feedback systems. *IEEE Transactions on Automatic Control*, 57(2):478–483, February 2012.

[HWM14] M. B. Horowitz, E. M. Wolff, and R. M. Murray. A compositional approach to stochastic optimal control with co-safe temporal logic specifications. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1466–1473. IEEE, 2014.

[Iwa16] R. Iwankiewicz. Dynamic response of mechanical systems to impulse process stochastic excitations: Markov approach. In *Journal of Physics: Conference Series*, volume 721, pages 12010–12024. IOP Publishing, 2016.

[Jan18] M. Jankovic. Control barrier functions for constrained control of linear systems with input delay. In *2018 Annual American Control Conference (ACC)*, pages 3316–3321, 2018.

[JJZ20] N. Jahanshahi, P. Jagtap, and M. Zamani. Synthesis of stochastic systems with partial information via control barrier functions. *21st IFAC World Congress*, 2020.

[JJZ21] N. Jahanshahi, P. Jagtap, and M. Zamani. Synthesis of partially observed jump-diffusion systems via control barrier functions. *IEEE Control Systems Letters*, 5(1):253–258, 2021.

[JMW96] Z-P. Jiang, I. M. Y. Mareels, and Y. Wang. A Lyapunov formulation of the nonlinear small-gain theorem for interconnected ISS systems. *Automatica*, 32(1):1211 – 1215, 1996.

[JP09] A. A. Julius and G. J. Pappas. Approximations of stochastic hybrid systems. *IEEE Transaction on Automatic Control*, 54(6):1193–1203, June 2009.

[JPZ20] P. Jagtap, G. J. Pappas, and M. Zamani. Control barrier functions for unknown nonlinear systems using gaussian processes. In *IEEE 59th Conference on Decision and Control (CDC)*, pages 4373–4378. IEEE, 2020.

[JSZ18] P. Jagtap, S. Soudjani, and M. Zamani. Temporal logic verification of stochastic systems using barrier certificates. In *International Symposium on Automated Technology for Verification and Analysis*, pages 177–193. Springer, 2018.

[JSZ20a] P. Jagtap, S. Soudjani, and M. Zamani. Formal synthesis of stochastic systems via control barrier certificates. *IEEE Transactions on Automatic Control*, 2020.

[JSZ20b] P. Jagtap, A. Swikir, and M. Zamani. Compositional construction of control barrier functions for interconnected control systems. (22):11, 2020.

[JZ16a] P. Jagtap and M. Zamani. Backstepping design for incremental stability of stochastic Hamiltonian systems. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 5367–5372. IEEE, 2016.

[JZ16b] P. Jagtap and M. Zamani. On incremental stability of time-delayed stochastic control systems. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 577–581. IEEE, 2016.

[JZ17a] P. Jagtap and M. Zamani. Backstepping design for incremental stability of stochastic Hamiltonian systems with jumps. *IEEE Transactions on Automatic Control*, 63(1):255–261, 2017.

[JZ17b] P. Jagtap and M. Zamani. QUEST: A tool for state-space quantization-free synthesis of symbolic controllers. In *International Conference on Quantitative Evaluation of Systems*, pages 309–313. Springer, 2017.

[JZ19] P. Jagtap and M. Zamani. Symbolic models for retarded jump–diffusion systems. *Automatica*, page 108666, 2019.

[JZst] P. Jagtap and M. Zamani. QUEST: A tool for state-space quantization-free synthesis of symbolic controllers. In *International Conference on Quantitative Evaluation of Systems*, pages 309–313. Springer International Publishing, September 2017, `https://www.hcs.ei.tum.de/en/software/quest/`.

[KB06] J. Klein and C. Baier. Experiments with deterministic ω-automata for formulas of linear temporal logic. *Theoretical Computer Science*, 363(2):182–195, 2006.

[KBJT19] J. Kenanian, A. Balkan, R. M. Jungers, and P. Tabuada. Data driven stability analysis of black-box switched linear systems. *Automatica*, 109:108533, 2019.

[KBTK18] T. Koller, F. Berkenkamp, M. Turchetta, and A. Krause. Learning-based model predictive control for safe exploration. In *IEEE Conference on Decision and Control (CDC)*, pages 6059–6066. IEEE, 2018.

[KKK95] M. Krstic, P. V. Kokotovic, and I. Kanellakopoulos. *Nonlinear and adaptive control design*. John Wiley & Sons, Inc., 1995.

[KLMJ07] D. J. Klein, P. Lee, K. A. Morgansen, and T. Javidi. Integration of communication and control using discrete time Kuramoto models for multivehicle coordination over broadcast networks. In *2007 46th IEEE Conference on Decision and Control*, pages 13–19, Dec 2007.

[KM13] I. Kolmanovsky and T. Maizenberg. Stochastic optimal control of jump diffusion excited energy harvesters. In *2013 American Control Conference*, pages 5049–5055, June 2013.

[KMSRG04] J. Kocijan, R. Murray-Smith, C. E. Rasmussen, and A. Girard. Gaussian process model based predictive control. In *Proceedings of the 2004 American control conference*, volume 3, pages 2214–2219. IEEE, 2004.

[Koc16] J. Kocijan. *Modelling and control of dynamic systems using Gaussian process models*. Springer, 2016.

[KRZ18] M. Khaled, M. Rungger, and M. Zamani. Sense: Abstraction-based synthesis of networked control systems. In *Proceedings of the 1st International Workshop on Methods and Tools for Rigorous System Design*, volume 272 of *Electronic Proceedings in Theoretical Computer Science*, pages 65–78. Open Publishing Association, 2018.

[KS91] I. Karatzas and S. E. Shreve. *Brownian Motion and Stochastic Calculus*, volume 113 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 2nd edition, 1991.

[Kus67] H. J. Kushner. *Stochastic Stability and Control, volume 33 of Mathematics in Science and Engineering*. Academic Press, New York, 1967.

[KV99] O. Kupferman and M. Y. Vardi. Model checking of safety properties. In *International Conference on Computer Aided Verification*, pages 172–183. Springer, 1999.

[LAB15] M. Lahijanian, S. B. Andersson, and C. Belta. Formal verification and synthesis for discrete-time stochastic systems. *IEEE Transactions on Automatic Control*, 60(8):2031–2045, 2015.

[Lav19] A. Lavaei. *Automated verification and control of large-scale stochastic cyber-physical systems: Compositional techniques*. PhD thesis, Technische Universität München, 2019.

[LCGG13] E. Le Corronc, A. Girard, and G. Goessler. Mode sequences as symbolic states in abstractions of incrementally stable switched systems. In *52nd IEEE Conference on Decision and Control (CDC)*, pages 3225–3230. IEEE, 2013.

[LD19a] L. Lindemann and D. V. Dimarogonas. Control barrier functions for multi-agent systems under conflicting local signal temporal logic tasks. *IEEE Control Systems Letters*, 3(3):757–762, July 2019.

[LD19b] L. Lindemann and D. V. Dimarogonas. Control barrier functions for signal temporal logic tasks. *IEEE Control Systems Letters*, 3(1):96–101, 2019.

[LD19c] L. Lindemann and D. V. Dimarogonas. Decentralized control barrier functions for coupled multi-agent systems under signal temporal logic tasks. In *2019 18th European Control Conference (ECC)*, pages 89–94, June 2019.

[Löd01] C. Löding. Efficient minimization of deterministic weak ω-automata. *Information Processing Letters*, 79(3):105 – 109, 2001.

[LS98] W. Lohmiller and J. J. Slotine. On contraction analysis for non-linear systems. *Automatica*, 34(6):683 – 696, 1998.

[LSZ18] A. Lavaei, S. Soudjani, and M. Zamani. Compositional synthesis of finite abstractions for continuous-space stochastic control systems: A small-gain approach. *IFAC-PapersOnLine*, 51(16):265–270, 2018.

[LSZ19] A. Lavaei, S. Soudjani, and M. Zamani. Compositional construction of infinite abstractions for networks of stochastic control systems. *Automatica*, 107:125–137, 2019.

[Mar03] R. Martí. *Multi-Start Methods*, pages 355–368. Springer US, Boston, MA, 2003.

[MGG13] S. Mouelhi, A. Girard, and G. Gössler. CoSyMA: A tool for controller synthesis using multi-scale abstractions. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 83–88. ACM, 2013.

[MGW17] P. J. Meyer, A. Girard, and E. Witrant. Compositional abstraction and safety synthesis using overlapping symbolic models. *IEEE Transactions on Automatic Control*, 63(6):1835–1841, 2017.

[MJDT10] M. Mazo Jr, A. Davitian, and P. Tabuada. PESSOA: A tool for embedded controller synthesis. In *International Conference on Computer Aided Verification*, pages 566–569. Springer, 2010.

[MP12] Z. Manna and A. Pnueli. *The temporal logic of reactive and concurrent systems: Specification*. Springer Science & Business Media, 2012.

[MPS95] O. Maler, A. Pnueli, and J. Sifakis. On the synthesis of discrete controllers for timed systems. In *Annual Symposium on Theoretical Aspects of Computer Science*, pages 229–242. Springer, 1995.

[MZ12] R. Majumdar and M. Zamani. Approximately bisimilar symbolic models for digital control systems. In P. Madhusudan and Sanjit A. Seshia, editors, *Computer Aided Verification*, volume 7358 of *Lecture Notes in Computer Science (LNCS)*, pages 362–377. Springer Berlin Heidelberg, July 2012.

[NA18] P. Nilsson and A. D. Ames. Barrier functions: Bridging the gap between planning from specifications and safety-critical control. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 765–772. IEEE, 2018.

[Øks02] B. Øksendal. *Stochastic differential equations: An introduction with applications*. Springer, 5th edition, November 2002.

[ØS05] B. Øksendal and A. Sulem. *Applied Stochastic Control of Jump Diffusions*. Universitext. Springer-Verlag, Berlin, 2005.

[Par03] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical programming*, 96(2):293–320, 2003.

[PGT08] G. Pola, A. Girard, and P. Tabuada. Approximately bisimilar symbolic models for nonlinear control systems. *Automatica*, 44(10):2508–2516, October 2008.

[PJP07] S. Prajna, A. Jadbabaie, and G. J. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, 2007.

[PPD16] G. Pola, P. Pepe, and M. D. Di Benedetto. Symbolic models for networks of control systems. *IEEE Transactions on Automatic Control*, 61(11):3663–3668, November 2016.

[PPDB15] G. Pola, P. Pepe, and M. D. Di Benedetto. Symbolic models for time-varying time-delay systems via alternating approximate bisimulation. *International Journal of Robust and Nonlinear Control*, 25(14):2328–2347, 2015.

[PPDBT10] G. Pola, P. Pepe, M. D. Di Benedetto, and P. Tabuada. Symbolic models for nonlinear time-delay systems using approximate bisimulations. *Systems & Control Letters*, 59(6):365–373, June 2010.

[PPP02] S. Prajna, A. Papachristodoulou, and P. A. Parrilo. Introducing SOSTOOLS: a general purpose sum of squares programming solver. In *Proceedings of the 41st IEEE Conference on Decision and Control*, volume 1, pages 741–746, 2002.

[PR16] V. I. Paulsen and M. Raghupathi. *An introduction to the theory of reproducing kernel Hilbert spaces*, volume 152. Cambridge University Press, 2016.

[Pra06] S. Prajna. Barrier certificates for nonlinear model validation. *Automatica*, 42(1):117–126, 2006.

[PTS09] Q. C. Pham, N. Tabareau, and J. J. Slotine. A contraction theory approach to stochastic incremental stability. *IEEE Transactions on Automatic Control*, 54(4):816–820, April 2009.

[PWN06] A. V. Pavlov, N. Wouw, and H. Nijmeijer. *Uniform output regulation of nonlinear systems: A convergent dynamics approach.* Springer Science & Business Media, 2006.

[Rat06] S. Ratschan. Efficient solving of quantified inequality constraints over the real numbers. *ACM Transactions on Computational Logic (TOCL)*, 7(4):723–748, 2006.

[Rat17] S. Ratschan. Simulation based computation of certificates for safety of dynamical systems. In *International Conference on Formal Modeling and Analysis of Timed Systems*, pages 303–317. Springer, 2017.

[RNC+03] S. J. Russell, P. Norvig, J. F. Canny, J. M. Malik, and D. D. Edwards. *Artificial intelligence: a modern approach*, volume 2. Prentice hall Upper Saddle River, 2003.

[RS15] H. Ravanbakhsh and S. Sankaranarayanan. Counter-example guided synthesis of control Lyapunov functions for switched systems. In *54th Annual Conference on Decision and Control (CDC)*, pages 4232–4239. IEEE, 2015.

[RS17] H. Ravanbakhsh and S. Sankaranarayanan. A class of control certificates to ensure reach-while-stay for switched systems. In D. Fisman and S. Jacobs, editors, *Proceedings Sixth Workshop on Synthesis, Heidelberg, Germany, 22nd July 2017*, volume 260 of *Electronic Proceedings in Theoretical Computer Science*, pages 44–61. Open Publishing Association, 2017.

[Ruf07] B. S. Ruffer. Monotone dynamical systems, graphs, and stability of large scale interconnected systems. *Ph.D. thesis, Fachbereich 3, Mathematik und Informatik, Universität Bremen, Germany*, 2007.

[RWR17] G. Reissig, A. Weber, and M. Rungger. Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62(4):1781–1796, April 2017.

[RZ16a] M. Rungger and M. Zamani. Compositional construction of approximate abstractions of interconnected control systems. *IEEE Transactions on Control of Network Systems*, 5(1):116–127, 2016.

[RZ16b] M. Rungger and M. Zamani. SCOTS: A tool for the synthesis of symbolic controllers. In *Proceedings of the 19th International Conference on Hybrid Systems: Computation and Control*, pages 99–104. ACM, 2016.

[SA13] S. Soudjani and A. Abate. Adaptive and sequential gridding procedures for the abstraction and verification of stochastic processes. *SIAM Journal on Applied Dynamical Systems*, 12(2):921–956, 2013.

[SA15] P. S. Skardal and A. Arenas. Control of coupled oscillator networks with application to microgrid technologies. *Science advances*, 1(7):e1500339, 2015.

[SAA16] S. Soudjani, D. Adzkiya, and A. Abate. Formal verification of stochastic max-plus-linear systems. *IEEE Transactions on Automatic Control*, 61(10):2861–2876, Oct 2016.

[SAM15] S. Soudjani, A.Abate, and R. Majumdar. Dynamic Bayesian networks as formal abstractions of structured stochastic processes. In *26th International Conference on Concurrency Theory (CONCUR 2015)*, volume 42 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 169–183. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2015.

[SCE18] M. Srinivasan, S.l Coogan, and M. Egerstedt. Control of multi-agent systems with finite time control barrier certificates and temporal logic. In *2018 IEEE Conference on Decision and Control (CDC)*, pages 1991–1996. IEEE, 2018.

[SGA15] S. Soudjani, C. Gevaerts, and A. Abate. FAUST^2: Formal Abstractions of Uncountable-STate STochastic processes. In C. Baier and C. Tinelli, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 272–286, Berlin, Heidelberg, 2015. Springer Berlin Heidelberg.

[SGZ18] A. Swikir, A. Girard, and M. Zamani. From dissipativity theory to compositional synthesis of symbolic models. In *Proceedings of the 4th Indian Control Conference*, pages 30–35, 2018.

[Sha13] L. Shaikhet. *Lyapunov functionals and stability of stochastic functional differential equations*. Springer Science & Business Media, 2013.

[SJZG18] A. Saoud, P. Jagtap, M. Zamani, and A. Girard. Compositional abstraction-based synthesis for cascade discrete-time control systems. *IFAC-PapersOnLine*, 51(16):13–18, 2018.

[SJZG21] A. Saoud, P. Jagtap, M. Zamani, and A. Girard. Compositional abstraction-based synthesis for interconnected systems: An approximate composition approach. *IEEE Transactions on Control of Network Systems (TCNS)*, 2021.

[SKF08] M. W. Seeger, S. M. Kakade, and D. P. Foster. Information consistency of nonparametric Gaussian process methods. *IEEE Transactions on Information Theory*, 54(5):2376–2382, 2008.

[SKKS12] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger. Information-theoretic regret bounds for Gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

[Sko09] A. Skorokhod. *Asymptotic methods in the theory of stochastic differential equations*. American Mathematical Soc., 2009.

[Som04] F. Somenzi. CUDD: CU decision diagram package-release 2.4.0. *University of Colorado at Boulder*, 2004.

[Sou14] S. Soudjani. *Formal abstractions for automated verification and synthesis of stochastic systems*. 2014.

[SRK$^+$14] I. Saha, R. Ramaithitima, V. Kumar, G. J. Pappas, and S. A. Seshia. Automated composition of motion primitives for multi-robot systems from safe LTL specifications. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1525–1532, 2014.

[SS07] G. B. Stan and R. Sepulchre. Analysis of interconnected oscillators by dissipativity theory. *IEEE Transactions on Automatic Control*, 52(2):256–270, February 2007.

[ST12] J. Steinhardt and R. Tedrake. Finite-time regional verification of stochastic non-linear systems. *The International Journal of Robotics Research*, 31(7):901–923, 2012.

[ST15] R. Sebastiani and P. Trentin. OptiMathSAT: A tool for optimization modulo theories. In *International conference on computer aided verification*, pages 447–454. Springer, 2015.

[Stu99] J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization methods and software*, 11(1-4):625–653, 1999.

[SZ19a] A. Swikir and M. Zamani. Compositional synthesis of finite abstractions for networks of systems: A small-gain approach. *Automatica*, 107:551–561, 2019.

[SZ19b] A. Swikir and M. Zamani. Compositional synthesis of finite abstractions for networks of systems: A small-gain approach. *Automatica*, 107(11):551 – 561, 2019.

[SZ19c] A. Swikir and M. Zamani. Compositional synthesis of symbolic models for networks of switched systems. *IEEE Control Systems Letters*, 3(4):1056–1061, Oct 2019.

[Tab09] P. Tabuada. *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media, 2009.

[TMKA13] I. Tkachev, A. Mereacre, J-P. Katoen, and A. Abate. Quantitative automata-based controller synthesis for non-autonomous stochastic hybrid systems. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 293–302. ACM, 2013.

[UBKH17] J. Umlauft, T. Beckers, M. Kimmel, and S. Hirche. Feedback linearization using Gaussian processes. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 5249–5255. IEEE, 2017.

[UPH18a] J. Umlauft, L. Pöhler, and S. Hirche. An uncertainty-based control Lyapunov approach for control-affine systems modeled by Gaussian process. *IEEE Control Systems Letters*, 2(3):483–488, July 2018.

[UPH18b] J. Umlauft, L. Pöhler, and S. Hirche. An uncertainty-based control Lyapunov approach for control-affine systems modeled by Gaussian process. *IEEE Control Systems Letters*, 2(3):483–488, 2018.

[WA07] P. Wieland and F. Allgöwer. Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, 40(12):462–467, 2007.

[WCS12] Z. Wu, M. Cui, and P. Shi. Backstepping control in vector form for stochastic Hamiltonian systems. *SIAM Journal on Control and Optimization*, 50(2):925–942, April 2012.

[WD13] Y. Wang and F. J. Doyle. Exponential synchronization rate of Kuramoto oscillators in the presence of a pacemaker. *IEEE Transactions on Automatic Control*, 58(4):989–994, April 2013.

[WR06] C. K. Williams and C. E. Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

[WTE18] L. Wang, E. A. Theodorou, and M. Egerstedt. Safe learning of quadrotor dynamics using barrier certificates. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2460–2465. IEEE, 2018.

[WTL15] T. Wongpiromsarn, U. Topcu, and A. Lamperski. Automata theory meets barrier certificates: Temporal logic verification of nonlinear systems. *IEEE Transactions on Automatic Control*, 61(11):3344–3355, 2015.

[WTO+11] T. Wongpiromsarn, U. Topcu, N. Ozay, H. Xu, and R. M. Murray. TuLiP: A software toolbox for receding horizon temporal logic planning. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pages 313–314. ACM, 2011.

[YTB19] G. Yang, R. Tron, and C. Belta. Continuous-time signal temporal logic planning with control barrier function. *arXiv preprint arXiv:1903.03860*, 2019.

[ZA14] M. Zamani and A. Abate. Approximately bisimilar symbolic models for randomly switched stochastic systems. *Systems & Control Letters*, 69:38–46, July 2014.

[ZA17] M. Zamani and M. Arcak. Compositional abstraction for networks of control systems: A dissipativity approach. *IEEE Transactions on Control of Network Systems*, 5(3):1003–1015, 2017.

[ZAG15] M. Zamani, A. Abate, and A. Girard. Symbolic models for stochastic switched systems: A discretization and a discretization-free approach. *Automatica*, 55:183–196, May 2015.

[ZCA13] B. G. Zhang, L. Chen, and K. Aihara. Incremental stability analysis of stochastic hybrid systems. *Nonlinear Analysis: Real World Applications*, 14(2):1225–1234, April 2013.

[ZMEM+14] M. Zamani, P. M. Esfahani, R. Majumdar, A. Abate, and J. Lygeros. Symbolic control of stochastic systems via approximately bisimilar finite abstractions. *IEEE Transactions on Automatic Control*, 59(12):3135–3150, December 2014.

[ZPMT11] M. Zamani, G. Pola, M. Mazo, and P. Tabuada. Symbolic models for nonlinear control systems without stability assumptions. *IEEE Transactions on Automatic Control*, 57(7):1804–1809, 2011.

[ZPV19] S. Zhu, G. Pu, and M. Y. Vardi. First-order vs. second-order encodings for LTL_f-to-automata translation. *arXiv preprint arXiv:1901.06108*, 2019.

[ZT11] M. Zamani and P. Tabuada. Backstepping design for incremental stability. *IEEE Transactions on Automatic Control*, 56(9):2184–2189, September 2011.

[ZTA16] M. Zamani, I. Tkachev, and A. Abate. Towards scalable synthesis of stochastic control systems. *Discrete Event Dynamic Systems*, pages 1–29, 2016.

[ZvdWM13] M. Zamani, N. van de Wouw, and R. Majumdar. Backstepping controller synthesis and characterizations of incremental stability. *Systems & Control Letters*, 62(10):949–962, October 2013.