Formal Analysis of Control Systems via Inductive Approaches: Tackling Scalability, Conservatism, and Complex Specifications

Mahathi Anand

München, 2023

Formal Analysis of Control Systems via Inductive Approaches: Tackling Scalability, Conservatism, and Complex Specifications

Mahathi Anand

Dissertation
an der Fakultät Für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität
München

vorgelegt von
Mahathi Anand
aus Hyderabad, Indien

München, den 12/06/2023

Erstgutachter: Prof. Dr. Majid Zamani
Zweitgutachter: Prof. Dr. Sadegh Soudjani
Tag der muendlichen Prüfung: 29/09/2023

Eidesstattliche Versicherung

Hiermit erkläre ich, Mahathi Anand, an Eides statt, dass die vorliegende Dissertation ohne unerlaubte Hilfe gemäß Promotionsordnung vom 12.07.2011, §8, Abs. 2 Pkt. 5, angefertigt worden ist.

München, 12.06.2023

Mahathi Anand

Contents

List of Figures

List of Tables

Abstract

This dissertation is concerned with the formal analysis of complex control systems via inductive approaches using barrier certificates. In general, safety-critical applications such as air traffic networks, autonomous vehicles, power grids, medical devices, and robotic equipment, are expected to satisfy complex logic specifications including but not limited to safety, reachability, and security. Due to several factors such as the continuous-state evolution of systems' trajectories, large systems' sizes, disturbances, etc., verification and synthesis of control systems against such high-level logic specifications is a challenging task.

An interesting yet simple way to tackle the verification and synthesis problem for logic specifications is to utilize inductive approaches based on *barrier certificates*. Barrier certificates take the form of inductive invariants and provide sufficient conditions for the satisfaction of safety or reachability specifications. Therefore, the verification and synthesis problem is reduced to the discovery of suitable barrier certificates. However, finding suitable barrier certificates can be a difficult problem due to several factors. First, the computation of barrier certificates is not scalable to large-scale systems. Second, the conditions imposed by barrier certificates are restrictive, making it difficult to search for one. Third, barrier certificate-based methods are limited to the analysis of safety or reachability specifications. As a result, they are not directly applicable to complicated logic tasks such as those expressed by ω-regular properties or (in)finite strings over automata, as well as security specifications such as those expressed by hyperproperties. In this regard, the dissertation focuses on alleviating the aforementioned issues and provides novel techniques to verify and synthesize controllers for (possibly large-scale and stochastic) control systems against the aforementioned specifications.

The first part of the thesis proposes a compositional framework for scalable construction of *control barrier certificates* for large-scale discrete-time stochastic control systems. In particular, we show that by considering the large-scale system as an interconnected one composed of several subsystems, one may construct control barrier certificates for the interconnected system by searching for so-called *control sub-barrier certificates* for subsystems and utilizing some compositionality conditions based on *small-gain* and *dissipativity* approaches. Correspondingly, one may also synthesize controllers that can be applied to the interconnected system in a decentralized manner, so that the large-scale system satisfies safety specifications over (in)finite time horizons with some probability lower bounds.

In the second part of the thesis, we propose a new notion of *k-inductive barrier certificates* for the verification of (stochastic) discrete-time dynamical systems against safety and reachability specifications. In particular, we illustrate that due to the restrictive nature of the traditional barrier

certificate conditions, it is not always possible to find suitable barrier certificates even when the system is guaranteed to satisfy the desired specifications. Then, we extend the k-induction principle utilized in software verification to propose several notions of k-inductive barrier certificates that relax the traditional barrier certificate conditions. As a result, larger classes of functions may act as barrier certificates, making them easier to find. In the context of non-stochastic systems, we propose two notions of k-inductive barrier certificates and provide formal guarantees for safety specifications. In the case of stochastic systems, we propose one notion of k-inductive barrier certificates for safety and two notions for tackling reachability specifications. Then, we obtain probabilistic guarantees for the satisfaction of safety and reachability specifications over infinite time horizons, respectively.

The last part of the thesis is concerned with the analysis of (stochastic) control systems against complex logic specifications beyond safety and reachability. First, we consider the synthesis problem for (possibly large-scale) stochastic control systems against *trace properties*, which describe specifications over individual traces of the system. Examples of such properties include ω-regular languages or (in)finite words over automata. We provide an automata-theoretic approach to decompose such complex specifications into sequential safety specifications. We then utilize the probability guarantees obtained for the safety specifications and combine them to obtain probability lower bounds for the satisfaction of original specifications. We provide such guarantees over both finite and infinite time horizons. Secondly, we consider the verification problem for non-stochastic systems against specifications that can be expressed over *sets of traces*, called hyperproperties. Hyperproperties can express many security and planning specifications that cannot be considered using ω-regular languages. In this context, we provide an automata-theoretic approach to decompose hyperproperties into smaller verification conditions called *conditional invariances*. Then, we introduce a new notion of so-called *augmented barrier certificates* constructed on the augmented system (*i.e.* self-composition of the system) to provide guarantees for the satisfaction of the conditional invariances. These guarantees may then be combined to achieve the satisfaction of original hyperproperties.

We also present computational techniques to search for suitable barrier certificates and controllers based on sum-of-squares (SOS) programming and satisfiability modulo theories (SMT) solvers. To demonstrate the effectiveness of our results, we consider several case studies such as room temperature control in buildings, networked Kuramoto oscillators, RLC circuits, and vehicle models.

Zusammenfassung

Diese Dissertation befasst sich mit der formalen Analyse komplexer Regelkreise, wobei induktive Ansätze unter Verwendung von *Barrierezertifikaten* zum Einsatz kommen. Im Allgemeinen wird von sicherheitskritischen Anwendungen wie Flugverkehrsnetzen, autonomen Fahrzeugen, Stromnetzen, medizinischen Geräten und Roboteranlagen erwartet, dass sie komplexe formale Spezifikationen erfüllen, um etwa die Betriebs- und Informationssicherheit zu gewährleisten oder anderweitige Ziele wie beispielsweise bestimmte Erreichbarkeitseigenschaften einzuhalten. Aufgrund verschiedener Faktoren wie der kontinuierlichen Zeitentwicklung der Systemtrajektorien, der Größe der Systeme, Störungen usw. ist die Verifizierung und Synthese von Steuersystemen anhand von solch allgemeinen logischen Spezifikationen eine anspruchsvolle Aufgabe.

Ein interessanter und dennoch einfacher Weg, das Verifikations- und Syntheseproblem für formale Spezifikationen anzugehen, ist die Verwendung von induktiven Ansätzen, die auf *Barrierezertifikaten* basieren. Diese haben die Form von induktiven Invarianten und liefern hinreichende Bedingungen für die Erfüllung von Sicherheits- oder Erreichbarkeitsanforderungen. Daher reduziert sich das Verifikations- und Syntheseproblem auf die Entdeckung geeigneter Barrierezertifikate. Die Suche nach geeigneten Barrierezertifikaten kann jedoch aufgrund mehrerer Faktoren ein schwieriges Problem darstellen. Erstens skaliert die Berechnung von Barrierezertifikaten nicht ohne weiteres auf große Systeme. Zweitens stellen die von Barrierezertifikaten auferlegten Bedingungen eine starke Einschränkung dar, was die Suche nach einem solchen Zertifikat erschwert. Drittens sind auf Barrierezertifikaten basierende Methoden auf die Analyse von Sicherheits- oder Erreichbarkeitsspezifikationen limitiert. Infolgedessen sind sie nicht direkt auf komplizierte logische Aufgaben anwendbar, wie z.B. solche, die durch ω-reguläre Eigenschaften oder (un)endliche Zeichenketten über Automaten ausgedrückt werden, oder auf bestimmte Sicherheitsspezifikationen, etwa wenn sie durch Hypereigenschaften ausgedrückt werden. In dieser Hinsicht konzentriert sich die Dissertation auf die Linderung der oben genannten Probleme und stellt neuartige Techniken zur Verifikation und Synthese von Reglern für (möglicherweise große und stochastische) Regelkreise hinsichtlich der oben genannten Spezifikationen zur Verfügung.

Im ersten Teil der Dissertation wird ein kompositorischer Rahmen für die skalierbare Konstruktion von *Regelungsbarrierezertifikaten* für große, zeitdiskrete und stochastische Regelkreise vorgeschlagen. Insbesondere zeigen wir, dass man durch die Zerlegung des Systems in mehrere, zusammenhängende Subsysteme Regelungsbarrierezertifikate für das Gesamtsystem konstruieren kann, indem man nach sogenannten *Regelungsunterbarrierezertifikaten* für die Subsysteme sucht. Hierzu lassen sich bestimmte Kompositionalitätsbedingungen auf der Basis von *small-gain* und *dissipativity* Ansätzen formulieren. Dementsprechend kann man auch Regler synthetisieren, die

dezentral auf die zusammenhängenden Komponenten des Systems angewendet werden können, so dass das Gesamtsystem die Sicherheitsspezifikationen über (un)endliche Zeithorizonte mit einigen Wahrscheinlichkeitsuntergrenzen erfüllt.

Im zweiten Teil der Dissertation stellen wir das neuartige Konzept der *k-induktiven Barrierezertifikate* für die Verifikation von (stochastischen) zeitdiskreten dynamischen Systemen bezüglich Sicherheits- und Erreichbarkeitsspezifikationen vor. Insbesondere zeigen wir, dass es aufgrund der restriktiven Natur der traditionellen Barrierezertifikatsbedingungen nicht immer möglich ist, geeignete Barrierezertifikate zu finden, selbst wenn das System garantiert die gewünschten Spezifikationen erfüllt. Im Anschluss erweitern wir das k-Induktionsprinzip, das in der Softwareverifikation verwendet wird, indem wir mehrere Konzepte für k-induktive Barrierezertifikate vorschlagen, die die traditionellen Barrierezertifikatsbedingungen lockern. Infolgedessen können größere Klassen von Funktionen als Barrierezertifikate fungieren, wodurch sie leichter zu finden sind. Im Zusammenhang mit nicht-stochastischen Systemen führen wir zwei Ausprägungen von k-induktiven Barrierezertifikaten ein und geben formale Garantien für Sicherheitsspezifikationen. Für stochastische Systeme stellen wir ein Konzept für k-induktive Barrierezertifikate für Sicherheit und zwei Konzepte für die Behandlung von Erreichbarkeitsspezifikationen vor. Danach erarbeiten wir probabilistische Garantien für die Erfüllung von Sicherheits- und Erreichbarkeitsanforderungen über unendliche Zeithorizonte.

Der letzte Teil der Dissertation befasst sich mit der Analyse von (stochastischen) Regelkreisen bezüglich komplexer logischer Spezifikationen jenseits von Sicherheit und Erreichbarkeit. Zunächst betrachten wir das Syntheseproblem für (möglicherweise großräumige) stochastische Regelkreise im Hinblick auf *Spureigenschaften*, also Spezifikationen über einzelne Spuren des Systems. Beispiele für solche Eigenschaften sind ω-reguläre Sprachen oder (un)endliche Wörter über Automaten. Wir bieten einen automaten-theoretischen Ansatz, um solche komplexen Spezifikationen in sequenzielle Sicherheitsspezifikationen zu zerlegen. Wir verwenden sogleich die für die Sicherheitsspezifikationen erhaltenen Wahrscheinlichkeitsgarantien und kombinieren sie, um untere Wahrscheinlichkeitsschranken für die Erfüllung der ursprünglichen Spezifikationen zu erhalten. Wir geben derartige Garantien sowohl für endliche als auch für unendliche Zeithorizonte. Im Anschluss betrachten wir das Verifikationsproblem für nicht-stochastische Systeme bezüglich Spezifikationen, die über *Mengen von Spuren*, so genannte Hypereigenschaften, ausgedrückt werden können. Hypereigenschaften können viele Sicherheits- und Planungsspezifikationen ausdrücken, die mittels ω-regulären Sprachen nicht betrachtet werden können. In diesem Zusammenhang stellen wir einen automaten-theoretischen Ansatz zur Verfügung, um Hypereigenschaften in kleinere Verifikationsbedingungen, sogenannte *bedingten Invarianzen*, zu zerlegen. Darauf aufbauend führen wir das Konzept des *erweiterten Barrierezertifikats* ein, welches auf dem (mittels Selbstkomposition) erweiterten System konstruiert wird, um Garantien für die Erfüllung der bedingten Invarianzen zu geben. Diese Garantien können dann wiederum kombiniert werden, um die Erfüllung der ursprünglichen Hypereigenschaften zu erreichen.

Acknowledgments

This dissertation is the result of the last four years of doctoral research at the Chair for Software and Computational Systems Lab, Department of Computer Science, University of Munich (LMU Munich). This dissertation would not have been possible without the continuous, generous support of those involved in my life both professionally and personally, and I would like to take this opportunity to extend my thanks to everyone.

Most importantly, I would like to express my gratitude to my advisor, Prof. Dr. Majid Zamani for presenting me with the opportunity to pursue my doctoral research, and for guiding me throughout my journey with his consistent support and patience. I sincerely appreciate his encouragement to pursue an academic career and for introducing me to this interesting field of research. I am forever grateful to him for making me a good researcher and inspiring me to be a better one still.

I wish to sincerely thank the research training group ConVeY (DFG GRK 2428), as well as the principal investigators and doctoral researchers within ConVeY for providing a fruitful interdisciplinary research environment to grow as a researcher. I want to thank Prof. George Pappas for providing me with a great opportunity to visit his group at the University of Pennsylvania, Philadelphia, USA, for his valuable advice and for introducing me to new topics of research. I also wish to thank Dr. Sadegh Soudjani for being the external examiner for my dissertation.

I am grateful to my collaborators, Prof. Ashutosh Trivedi, Prof. Abolfazl Lavaei, and Vishnu Murali, for their involvement in my doctoral research without whom this thesis would have been incomplete. I sincerely appreciate the involvement of my colleagues at the Hybrid Control Systems Lab. The insightful discussions during the group meetings and the wonderful time shared together will be forever memorable. Specifically, I would like to thank my former colleague Prof. Pushpak Jagtap for encouraging me to pursue research and for guiding me throughout. I also wish to appreciate my colleagues Niloofar and Siyuan for providing me with the necessary support that helped me towards my academic pursuits.

I am genuinely indebted to my friends and family who rendered me with much-needed encouragement, love and endless support throughout my journey. I wish to appreciate my friends Akash and Sanjay for their presence and support despite being continents away. I also want to thank Adyasha, Sadwini, and Pavan for being there for me and giving me a good time in Munich. Finally, no words can express how grateful I am to Amma, Appa and Anirudh for believing in me even when I lacked confidence in myself and for healing me with unconditional love and motivation. This dissertation is dedicated to them for providing me with endless freedom and support to pursue my dreams throughout my life.

Chapter 1

Introduction

1.1 Motivation and Contributions

Research problems studied in classical control theory usually involve checking complex continuous-space mathematical models against simple properties such as stability or invariance. On the other hand, in the field of formal methods, simple mathematical models such as finite transition systems are checked against complex logic tasks. As such, many real-world applications employ complex dynamical systems to perform complex tasks. For example, consider iRobot's self-cleaning vacuum Roomba i7+ that is required to navigate continuous spaces and clean the floor, and then return to the charging dock and empty the contents of Roomba's bin directly into the dock. As another example, consider an air traffic control system where it is required for a network of aircraft to keep a safe distance from each other and avoid collisions while following the flight path. Moreover, many of these applications, like the air traffic system, are safety-critical, and failure to perform the necessary tasks can lead to catastrophic consequences. Therefore, enforcing and providing formal guarantees for such complex tasks has gained considerable attention in the past few years.

Traditionally, this problem is solved by simply bridging the gap between the problems studied in control theory and formal methods via finite discretizations of complex control systems. Specifically, finite abstractions (*i.e.* symbolic models) are obtained for the complex systems, and then, machinery from the field of formal methods like model-checking [15] is utilized to provide guarantees over the original systems. These approaches have been quite popular in the last few years and there have been several results in this direction including but not limited to [122, 114, 76, 17, 135, 85, 72, 80]. Unfortunately, to obtain such abstractions, one needs to discretize or quantize the state set of the concrete system, resulting in an exponential blow-up with respect to state dimensions.

Discretization-free approaches using barrier certificates [97] are an interesting alternative to abstraction-based methods. Barrier certificates are real-valued functions over the states of the system such that their existence guarantees the satisfaction of safety or reachability specifications. As such, barrier certificates are comparable to inductive invariants [125], *i.e.*, their value remains within a certain level set along the reachable states of the system. Then, the certification of safety or reachability properties via barrier certificates can be established through inductive proofs.

Unfortunately, searching for the existence of barrier certificates is in general a difficult problem. Relevant literature such as [97, 62, 63] utilize existing tools like sum-of-squares optimization and satisfiability modulo theory solvers to compute suitable barrier certificates by restricting the barrier certificate functions to a specific parametric form, such as exponential or polynomial, and then searching for their corresponding coefficients. However, these methods are still not very scalable to large-scale control systems. Moreover, the barrier certificate conditions defined in the existing literature are inherently restrictive, and as a result, one is not able to find barrier certificates even for systems that trivially satisfy the concerned properties. Lastly, while barrier certificate-based approaches are suited for simple safety and reachability specifications, they are not directly applicable to more complex logic tasks, such as those expressible by temporal logic specifications or (in)finite strings over automata.

This dissertation focuses on alleviating the aforementioned limitations of barrier-certificate-based approaches in order to formally analyze complex control systems against safety, reachability, and complex logic specifications. In particular, we are concerned with the analysis of different classes of systems (e.g. stochastic and non-stochastic) and develop techniques to:

- Synthesize suitable controllers enforcing safety specifications on large-scale stochastic control systems by obtaining the so-called control barrier certificates in a compositional manner, thus alleviating the scalability issues associated with the construction of control barrier certificates;
- Alleviate the conservatism imposed by barrier certificate conditions via k-induction so that larger classes of functions may act as barrier certificates, making them easier to find;
- Obtain qualitative and/or quantitative (*i.e.*, probabilistic) guarantees for the satisfaction of safety, reachability, and other complex logic (hyper)properties for non-stochastic and/or stochastic (control) systems, respectively.

The first part of the thesis is reserved for potentially mitigating the scalability issues pertaining to the synthesis of suitable controllers against safety specifications for large-scale discrete-time stochastic control systems via barrier certificate-based approaches. In particular, we are concerned with the synthesis of controllers along with the so-called *control barrier certificates* so that safety specifications may be satisfied by stochastic control systems with some probability lower bounds. Since the monolithic construction of control barrier certificates is not scalable to large-scale systems, we consider a compositional framework for the same by considering large-scale systems as interconnected ones consisting of smaller subsystems. Then, we propose a notion of so-called *control sub-barrier certificates* which can be obtained for the subsystems along with corresponding local controllers. Then, by utilizing some compositionality conditions based on *small-gain* and *dissipativity*-based theories, one can construct the control barrier certificates for the large-scale systems by utilizing the control sub-barrier certificates. Finally, the control barrier certificates are used to compute probability lower bounds on the satisfaction of safety specifications, and the local controllers are applied to the large-scale system in a decentralized manner so that the system satisfies the safety specifications with the aforementioned probability lower bounds. Such bounds are obtained for both *finite* as well as *infinite* time horizons.

The second part of the thesis is concerned with alleviating the restrictiveness of the standard barrier certificate conditions. We leverage the inductive nature of barrier certificates and weaken the conditions imposed on them by utilizing k-induction instead of standard induction. By doing so, a larger class of functions may act as barrier certificates, making it easier to find them. In this part of the thesis, we utilize the so-called *k-inductive barrier certificates* to provide qualitative safety verification guarantees for discrete-time non-stochastic dynamical systems (*i.e.*, systems without control inputs) as well as probabilistic safety and reachability verification guarantees for discrete-time stochastic dynamical systems. For the safety verification of non-stochastic dynamical systems, we propose two alternative definitions for k-inductive barrier certificates, demonstrate their applicability over standard ones, compare the two notions and discuss their expressibility over one another. On the other hand, we obtain probability lower bounds on the safety satisfaction of stochastic dynamical systems by utilizing a single notion of k-inductive barrier certificates for stochastic safety. Finally, probabilistic reachability guarantees for stochastic dynamical systems are achieved by considering two separate notions, one which computes probability lower bounds on reach-and-avoid specifications, while the other definition provides us reachability guarantees with probability 1. We also utilize some simple examples to demonstrate the effectiveness of k-inductive barrier certificates for stochastic systems. Note that for stochastic systems, probability guarantees are obtained over *infinite* time horizons.

The last part of the thesis focuses on extending the applicability of barrier certificate-based approaches beyond simple safety and reachability specifications. First, we consider complex specifications defined over individual execution traces of the system such as linear temporal logic, ω-regular properties, or (in)finite strings over automata. Consequently, we provide a controller synthesis procedure via control barrier certificates for the probabilistic satisfaction of (possibly large-scale) stochastic control systems against the aforementioned specifications. In order to extend the previously obtained control barrier certificate-based results for such specifications, we take an automata-theoretic approach. Specifically, we consider the automata associated with the complex specification and discharge the specification into a sequence of smaller safety synthesis tasks. Then, by utilizing control barrier certificates to compute probability bounds on the smaller safety tasks and combining them, we obtain probabilistic satisfaction guarantees over the original specifications. Correspondingly, we also propose a switching controller structure for the system in order to ensure the probabilistic satisfaction of the concerned specifications. Note that the probability guarantees here may be provided for both *finite* time horizons (*e.g.* finite automata) as well as *infinite* time horizons (*e.g.* ω-regular properties or ω-automata).

Secondly, for the first time, we extend the applicability of barrier certificate approaches for verifying discrete-time dynamical systems against *hyperproperties*. Hyperproperties [31] describe specifications that require quantification over multiple execution traces of the system, and as such, cannot be described by linear temporal logic or ω-regular properties. However, they can specify many relevant properties required by the continuous-space systems considered in our thesis, like opacity or optimality. In particular, we consider hyperproperties specified using hyper-temporal logic (HyperLTL), which utilizes universal and existential quantifiers in its syntax to reason about multiple trace executions. Due to the presence of such quantification, standard barrier certificate-based approaches cannot be extended to handle these specifications. Therefore, we propose a new notion of *augmented barrier certificates* defined over an augmented

system by taking the product of the original system with itself (*i.e.*, self-composition of the system), which provide us with sufficient conditions ensuring the satisfaction of some so-called *conditional invariances*. Then, inspired by our previously proposed results, we take an automata-theoretic approach to extend the applicability of our reasoning to hyperproperties by breaking down the automata associated with the specifications into smaller verification (*i.e.*, satisfaction of conditional invariances) tasks and utilizing augmented barrier certificates to solve these smaller tasks.

Note that our theoretical contributions are supported by suitable procedures to construct appropriate barrier certificates as well as controllers, where applicable. Such approaches include the use of sum-of-squares programming and satisfiability modulo theories. Moreover, we demonstrate the effectiveness of our results by applying the approaches to suitable case studies, including room temperature regulation networks, networks of Kuramoto oscillators, RLC circuits, and vehicle models.

1.2 Outline of the Thesis

The thesis is divided into six chapters including the current one which serves as an introduction to our contributions. The rest is structured as follows:

- **Chapter 2** introduces some preliminary concepts relevant to the thesis. It includes some mathematical notations that follow throughout the thesis, basic concepts borrowed from control theory such as system definitions and (control) barrier certificates, as well as some concepts borrowed from the formal methods community, like the definitions of the complex logic specifications considered in the thesis.

- **Chapter 3** through **Chapter 5** present the main technical contributions achieved in the thesis. For clarity in the presentation, the three chapters follow the same structure. First, the introduction provides the necessary motivation for the proposed results. This is followed by a brief literature review and a statement of contributions. The subsequent sections provide the necessary technical details on the problem considered and the proposed techniques to solve the problem. This is then followed by suitable case studies to illustrate the effectiveness of the approaches. The chapter is then concluded with a brief summary of our results. The following provides a quick overview of the technical contributions of the chapters:

  - **Chapter 3** proposes scalable construction of control barrier certificates for the *controller synthesis* of *discrete-time large-scale stochastic control systems* using two different compositionality techniques using small-gain and dissipativity-based approaches. Utilizing these control barrier certificates, *probabilistic guarantees* on the satisfaction of *safety specifications* are obtained.
  - **Chapter 4** introduces new notions of k-inductive barrier certificates to relax the traditional barrier certificate conditions. Utilizing these notions, we obtain *qualitative safety verification* for *discrete-time non-stochastic dynamical systems*, as well as

probabilistic safety and *reachability verification* for *discrete-time stochastic dynamical systems*.

- **Chapter 5** extends the applicability of control barrier certificate-based approaches for the *controller synthesis* of *discrete-time stochastic control systems* to specifications that can be represented as *deterministic finite automata* as well as *ω-regular properties*. Moreover, this chapter also proposes an approach for the *formal verification* of *discrete-time dynamical systems* against *hyperproperties*.

- **Chapter 6** provides a summary of the results established in the thesis and suggests some interesting directions for future research.

For ease in understanding the contributions of each chapter, the highlighted terms in the thesis outline represent the main problem solved. For instance, Chapter 3 is concerned with the controller synthesis of large-scale stochastic control systems to obtain probabilistic safety guarantees, and so on. Note that technical results established in the thesis are based on the publications [6, 4, 5, 8, 9, 10]. Individual contributions made by the author for the relevant publications are specifically highlighted in the beginning of each chapter.

Chapter 2

Preliminaries

In this chapter, we introduce some mathematical notations that will be used throughout the thesis. Moreover, we also introduce some preliminary concepts relevant to the thesis, including basic concepts borrowed from the field of control theory, computer science, and mathematics.

2.1 Notations

The set of real numbers, integers, and non-negative integers are denoted by $\mathbb{R}, \mathbb{Z}$, and $\mathbb{N}$, respectively. These notations are annotated with subscripts to restrict them in the usual way, *e.g.*, we use $\mathbb{R}_{\geq 0}$ to represent the set of positive real numbers. In addition, we use $\mathbb{R}^n$ to denote the real space of dimension n. Given N vectors $e_i \in \mathbb{R}^{n_i}$, the notation $e = [e_1; \ldots; e_N]$ denotes the concatenated vector of dimension $\sum_i n_i$. Correspondingly, we use $e(i)$ to refer to the i^{th} dimension of e, *i.e.*, $e(i) = e_i$. For a vector $e \in \mathbb{R}^n$, we denote the infinity norm by $\|e\|$ and Euclidean norm by $\|e\|_2$. Furthermore, for a matrix $M \in R^{n \times n}$, we use $\|M\|_F$ to denote the Frobenius norm of M. The identity matrix in $\mathbb{R}^{n \times n}$ is denoted by $\mathbb{I}_n$. Similarly, we use $\mathbf{0}_n$ and $\mathbb{1}_n$ to denote the column vector in $\mathbb{R}^n$ with all elements equal to 0 and 1, respectively.

For a finite set A, $|A|$ denotes the cardinality of A. We use $\varnothing$ to denote the empty set. The complement of a set B with respect to any set A is denoted by $A \setminus B = \{a \in A, a \notin B\}$. The power set of A is denoted by 2^A. The boundary and closure of A are denoted by ∂A and $\bar{A}$, respectively. Moreover, for any set A, A^n denotes the n-ary Cartesian power of A, *i.e.*, $A^n = \{(a_1, \ldots, a_n) \mid a_i \in A, i \in \{1, \ldots, N\}\}$. For two sets A and B, a function $f : A \to B$ is a mapping from A to B. The identity function on the set A is denoted by id_A. Given three sets A, B, and C, and functions $f : A \to B$ and $g : B \to C$, the composition of functions f and g is denoted by $g \circ f : A \to C$. For a function $f : A \to A$, we use f_n to denote the n^{th} iterate of f, defined as $f_0 = \mathrm{id}_A$ and $f_n = f_{n-1} \circ f$. For n functions $f_{(i)} : A_i \to B_i, i \in \{1, \ldots, N\}$, their Cartesian product $\prod_{i=1}^n f_{(i)} : \prod_{i=1}^n A_i \to \prod_{i=1}^n B_i$ is given by $\prod_{i=1}^n f_{(i)}(a_1, \ldots, a_n) = [f_{(1)}(a_1); \ldots; f_{(n)}(a_n)]$. Similarly, the n-ary Cartesian product of a function f, denoted by f^n, is the Cartesian product of f applied n times. A function $f : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ is said to be a class $\mathcal{K}$ function if it is continuous, strictly increasing, and $f(0) = 0$. A class $\mathcal{K}$ function $f(\cdot)$ belongs to the class $\mathcal{K}_\infty$ if $f(s) \to \infty$ as $s \to \infty$. Finally, we denote the disjunction ($\vee$) and conjunction ($\wedge$) of a Boolean function

$f : S \to \{0, 1\}$ over a (possibly infinite) index set S by $\bigvee_{s\in S} f(s)$ and $\bigwedge_{s\in S} f(s)$, respectively.

An alphabet Σ is a finite set of letters. An ω-sequence $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots)$ is an infinite concatenation of letters, i.e., for all $i \geq 0$ we have $\sigma_i \in \Sigma$. A finite sequence is such a sequence but with a finite length. We write Σ^* and Σ^ω for the set of finite and ω-sequences over Σ, and we let $\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$. For a sequence $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots) \in \Sigma^\omega$, let $\boldsymbol{\sigma}(i)$ be the i-th element of $\boldsymbol{\sigma}$ and $\boldsymbol{\sigma}(i, \infty)$ be the ω-sequence $(\sigma_i, \sigma_{i+1}, \ldots) \in \Sigma^\omega$ of $\boldsymbol{\sigma}$ starting from the i-th position. We let $\mathsf{zip} : (\Sigma^\omega)^p \to (\Sigma^p)^\omega$ denote a function that maps a p-tuple of sequences to a sequence of p-tuples, i.e.

$$(\boldsymbol{\sigma_1}, \boldsymbol{\sigma_2}, \ldots, \boldsymbol{\sigma_p}) \to (\boldsymbol{\sigma_1}(0), \boldsymbol{\sigma_2}(0), \ldots, \boldsymbol{\sigma_p}(0))(\boldsymbol{\sigma_1}(1), \boldsymbol{\sigma_2}(1), \ldots, \boldsymbol{\sigma_p}(1)) \ldots$$

and $\mathsf{unzip} : (\Sigma^p)^\omega \to (\Sigma^\omega)^p$ denotes the inverse of zip, i.e.

$$\boldsymbol{\sigma} \to ((\boldsymbol{\sigma}(0)(1), \boldsymbol{\sigma}(1)(1), \ldots), \ldots, (\boldsymbol{\sigma}(0)(p), \boldsymbol{\sigma}(1)(p), \ldots)).$$

2.2 System Definitions

In this thesis, we consider two classes of control systems defined over continuous state sets and discrete time, namely, non-stochastic control systems and stochastic control systems, respectively. In this section, we present the definitions of both classes of systems and discuss their behavior in detail. We first define the non-stochastic case.

Definition 1. *A non-stochastic discrete-time control system is a tuple*

$$\mathfrak{S} = (X, U, f), \tag{2.1}$$

where

- $X \subseteq \mathbb{R}^n$ *is the (possibly infinite) state set of the system;*
- $U \subseteq \mathbb{R}^m$ *is the (possibly infinite) input set of the system;*
- $f : X \times U \to X$ *is the transition function of the system that characterizes the dynamics of the system via the following difference equation*

$$\mathbf{x}(t + 1) = f(\mathbf{x}(t), \nu(t)), \tag{2.2}$$

where $t \in \mathbb{N}$, $\mathbf{x} : \mathbb{N} \to X$, and $\nu : \mathbb{N} \to U$ are the state and input sequences, respectively.

Given an initial condition $\mathbf{x}(0) = x_0$ and an input sequence ν, the state sequence of the dt-CS $\mathfrak{S}$ is obtained via (2.2) and is denoted by $\mathbf{x}_{x_0,\nu}$. The state sequence may be finite (*e.g.*, $\mathbf{x}_{x_0,\nu,n} = (x_0, \mathbf{x}(1), \ldots, \mathbf{x}(n))$, $n \in \mathbb{N}$), or infinite (*e.g.*, $\mathbf{x}_{x_0,\nu} = (x_0, \mathbf{x}(1), \ldots)$).

We now present stochastic control systems. Before doing so, some preliminary notions on probability spaces are in order. We consider the probability space $(\Omega, \mathcal{F}_\Omega, \mathbb{P}_\Omega)$, where Ω is the sample space, $\mathcal{F}_\Omega$ is a sigma-algebra on Ω consisting subsets of Ω as events, and $\mathbb{P}_\Omega$ is the

probability measure that assigns probability to those events. Random variables in this paper are assumed to be measurable functions of the form $X : \Omega \to S_X$. Any random variable X induces a probability measure on S_X as $Prob\{A\} = \mathbb{P}_\Omega\{X^{-1}(A)\}$ for any $A \in \mathcal{F}_X$, where $\mathcal{F}_X$ is the sigma-algebra on X. The topological space S_X is a Borel space if it is homeomorphic to a Borel subset of a Polish space, *i.e.*, a separable and completely metrizable space. The Borel sigma-algebra generated from Borel space S_X is denoted by $\mathcal{B}(S_X)$ and the map $f : S_X \to Y$ is measurable whenever it is Borel measurable.

Definition 2. *A stochastic discrete-time control system is a tuple*

$$\mathfrak{S} = (X, U, \varsigma, f), \tag{2.3}$$

where

- $X \subseteq \mathbb{R}^n$ *is a Borel space as the state set of the system. The tuple* $(X, \mathcal{B}(X))$ *is the measurable state space where* $\mathcal{B}(X)$ *denotes the Borel sigma-algebra on the state space;*
- $U \subseteq \mathbb{R}^m$ *is the Borel space as the input set of the system. The tuple* $(U, \mathcal{B}(U))$ *denotes the Borel sigma-algebra on the input space;*
- $\varsigma := \{\varsigma(t) : \Omega \to \mathcal{V}_\varsigma,\ t \in \mathbb{N}\}$ *is a sequence of independent and identically distributed (i.i.d.) random variables from a sample space* Ω *to the measurable space* $(\mathcal{V}_\varsigma, \mathcal{F}_\varsigma)$*;*
- $f : X \times U \times \mathcal{V}_\varsigma \to X$ *is a measurable function that characterizes the state evolution of* $\mathfrak{S}$ *through the following difference equation:*

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \nu(t), \varsigma(t)), \tag{2.4}$$

  where $\mathbf{x}(t) : \Omega \to X$ *is the state at time* $t \in \mathbb{N}$ *and* $\nu(t) : \Omega \to U$ *is the control input at time* $t \in \mathbb{N}$*.*

We associate with U a set $\mathcal{U}$ which consists of a collection of input sequences $\{\nu(t) : \Omega \to U,\ t \in \mathbb{N}\}$ such that $\nu(t)$ is independent of the random variable $\varsigma(s)$ is for all $t, s \in \mathbb{N}$ and $s \geq t$. For a given initial state $\mathbf{x}(0) = x_0$, and $\nu(\cdot) \in \mathcal{U}$, a random sequence $\mathbf{x}_{x_0,\nu} : \Omega \times \mathbb{N} \to X$ is the solution process of $\mathfrak{S}$ under the influence of ν starting from x_0 that is obtained from (2.4). Similar to the non-stochastic case, the solution process may be finite or infinite.

In some parts of the thesis, we deal with the controller synthesis problem, which refers to finding a suitable controller such that the system $\mathfrak{S}$ satisfies a desired property, such as safety. In this case, the control of dt-CS or dt-SCS $\mathfrak{S}$ in (3.1) is enforced by a controller $\varpi : X \to U$ where the control input $\nu(t) = \varpi(x(t))$ at any time t depends on the state at time t. Consequently, we denote the state sequence starting from the initial condition $\mathbf{x}(0) = x_0$ under the controller ϖ as $\mathbf{x}_{x_0,\varpi}$. In other parts of the thesis, we consider the verification problem, which assumes either that the controller is absent, *i.e.*, $\nu(t) = 0, \forall t \in \mathbb{N}$, or that the controller is designed a priori. In this case, the tuples in (2.1) and (2.3) are reduced to $\mathfrak{S} = (X, f)$, and $\mathfrak{S} = (X, \varsigma, f)$, respectively. Moreover, for the sake of simplicity in presentation, we use the term discrete-time non-stochastic dynamical systems (dt-DS) and the term discrete-time stochastic dynamical

systems (dt-SS), respectively, to refer to systems without the explicit mention of control inputs. Moreover, the state sequences of the dt-DS and the solution processes of the dt-SS are written without the explicit mention of the input sequence or the controller in their subscripts. Finally, the verification problem entails determining whether the system $\mathfrak{S}$ satisfies the desired specification.

Remark 1. *Note that we abuse the notation to utilize a unified representation $\mathfrak{S}$ for all the classes of systems that we consider in the thesis (dt-DS, dt-CS, dt-SS or dt-SCS). The meaning of the notation will be made clear with context.*

2.3 Safety and Reachability

An important problem in the field of control theory, especially in the case of safety-critical control systems, is to provide proof of correctness of system behavior. In this context, the safety problem is to certify whether the system trajectories (*i.e.* state sequences or solution processes) avoid some undesirable or unsafe configurations. Formally, given a set of initial states X_0 and a set of unsafe states X_u for the system $\mathfrak{S}$, a trajectory $\mathbf{x}_{x_0,\nu}$ starting from $x_0 \in X_0$ under the influence of control sequence ν is *safe* if it never visits the states in X_u, *i.e.*, we have that $\mathbf{x}_{x_0}(t) \notin X_u$, for all time $t \in \mathbb{N}$. In the context of non-stochastic control systems, one may be able to provide absolute guarantees for safety. However, in the case of stochastic control systems, one is concerned with obtaining probabilistic guarantees for safety, *i.e.*, tight lower bounds on the probability that the system avoids X_u. We now present the safety verification and synthesis problem in both non-stochastic and stochastic control systems, respectively.

Problem 1 (Safety Verification for dt-DS)**.** *Given a dt-DS $\mathfrak{S} = (X, f)$, a set of initial states $X_0 \subseteq X$, and a set of unsafe states $X_u \subseteq X$, determine whether the state sequences $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ starting from $x_0 \in X_0$, satisfy $\mathbf{x}_{x_0}(t) \notin X_u$, for all $t \in \mathbb{N}$.*

Problem 2 (Safe Controller Synthesis for dt-CS)**.** *Given a dt-CS $\mathfrak{S} = (X, U, f)$, a set of initial states $X_0 \in X$, and a set of unsafe states $X_u \subseteq X$, synthesize a controller ϖ such that the state sequences $\mathbf{x}_{x_0,\varpi}$ starting from $x_0 \in X_0$ satisfy $\mathbf{x}_{x_0,\varpi}(t) \notin X_u$, for all $t \in \mathbb{N}$.*

Problem 3 (Probabilistic Safety Verification for dt-SS)**.** *Given a dt-SS $\mathfrak{S} = (X, \varsigma, f)$, the sets of initial and unsafe states X_0 and X_u, respectively, compute a constant $0 \leq \varkappa \leq 1$ such that the system is safe with a probability bound of at least $\varkappa$, i.e.,*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin X_u \text{ for all } t \in \mathbb{N} \mid x_0 \in X_0\} \geq \varkappa. \tag{2.5}$$

Problem 4 (Probabilistic Safety Synthesis for dt-SCS)**.** *Given a dt-SS $\mathfrak{S} = (X, U, \varsigma, f)$, the sets of initial and unsafe states X_0 and X_u, respectively, compute a controller ϖ along with a constant $0 \leq \varkappa \leq 1$ such that the system is safe with a probability bound of at least $\varkappa$, i.e.,*

$$\mathbb{P}\{\mathbf{x}_{x_0,\varpi}(t) \notin X_u \text{ for all } t \in \mathbb{N} \mid x_0 \in X_0\} \geq \varkappa. \tag{2.6}$$

Another relevant specification concerned with dynamical systems is the reachability specification, which requires the system to eventually reach a desired set of states. Formally, given a set of initial states X_0 and a set of target states X_R, state sequences $\mathbf{x}_{x_0}$ of a dynamical system $\mathfrak{S}$ starting from some state $x_0 \in X_0$ is said to *reach* X_R if it eventually visits some states in X_R, *i.e.* we have that $\mathbf{x}_{x_0}(t) \in X_R$, for some time $t \in \mathbb{N}$. One can define the following problems concerning the verification of reachability properties, similar to those presented for safety.

Problem 5 (Reachability Verification for dt-DS)**.** *Given a dt-DS $\mathfrak{S} = (X, f)$, a set of initial states $X_0 \subseteq X$, and a set of target states $X_R \subseteq X$, determine whether the state sequences $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ starting from $x_0 \in X_0$ eventually reach X_R, i.e., if there exists $t \in \mathbb{N}$ such that $\mathbf{x}_{x_0}(t) \in X_R$.*

Problem 6 (Probabilistic Reachability Verification for dt-SS)**.** *Given a dt-SS $\mathfrak{S} = (X, \varsigma, f)$, a set of initial states and target states $X_0, X_R \subseteq X$, respectively, compute a constant $0 \leq \varkappa \leq 1$ such that $\mathfrak{S}$ satisfies the reachability property with a probability bound of at least $\varkappa$, i.e.,*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_R \text{ for some } t \in \mathbb{N} \mid x_0 \in X_0\} \geq \varkappa. \tag{2.7}$$

Remark 2. *We do not present problems pertaining to controller synthesis for reachability specifications as they are out of the scope of this thesis. However, it must be noted that similar problems may be formulated for such tasks.*

2.4 Specifications Beyond Safety and Reachability

In computer science literature, complex logic specifications beyond safety and reachability properties are usually defined over infinite trace executions of the system. We consider here trace properties that are subsets of infinite traces defined over the set of atomic propositions $\mathcal{AP}$, where each trace is an ω-sequence of letters from the set $2^{\mathcal{AP}}$. Such specifications may be expressed using temporal logic formulas like linear temporal logic (LTL) [15] or simply as infinite languages over automata (*e.g.*, Büchi automata, Streett automata, etc.) [124]. Note that these specifications may also be considered over finite traces rather than infinite traces, and correspondingly, one may utilize finite fragments of temporal logic specifications, such as LTL_f [38], or finite languages over automata like (non)deterministic finite automata [59]. In the following, we present the syntax and semantics of LTL.

2.4.1 Linear Temporal Logic (LTL)

Consider a set of atomic propositions $\mathcal{AP}$, which are input symbols that are relevant to the underlying system, and the alphabet $\Sigma = 2^{\mathcal{AP}}$ characterized by the subsets of these propositions. We refer to an infinite sequence (ω-sequence) of letters from Σ as an infinite trace. We write Σ^ω for the set of all infinite traces over Σ.

Syntax. An LTL formula over $\mathcal{AP}$ can be built from the following production rules:

$$\psi \;::=\; p \;\mid\; \neg\psi \;\mid\; \psi \vee \psi \;\mid\; \mathsf{X}\psi \;\mid\; \psi \,\mathsf{U}\, \psi,$$

where $p \in \mathcal{AP}$, X and U are the next and until operators, respectively. Other popular temporal operators such as globally (G), eventually (F) and release (R) can be derived from these minimal ones in a standard manner.

Semantics. Given an infinite trace σ and an LTL formula ψ, the formula ψ is valid for σ, i.e. $\sigma \models \psi$, if:

- $\psi = p$ and $p \in \sigma(0)$;
- $\psi = \neg\psi$ and $\sigma \not\models \psi$;
- $\psi = \psi_1 \vee \psi_2$ and $\sigma \models \psi_1$ or $\sigma \models \psi_2$;
- $\psi = \mathsf{X}\psi$ and $\sigma(1, \infty) \models \psi$;
- $\psi = \psi_1 \,\mathsf{U}\, \psi_2$ and $\sigma(i, \infty) \models \psi_2$ for some $i \geq 0$ and for all $0 \leq j < i$, we have that $\sigma(j, \infty) \models \psi_1$.

We refer the interested readers to [15] for more details on syntax and semantics of LTL properties. The syntax of LTL_f is defined similarly to LTL, but its semantics are interpreted over finite traces rather than infinite ones. *i.e.*, over Σ^* [38].

2.4.2 Temporal Logic for Hyperproperties (HyperLTL)

LTL specifications can express many relevant safety and mission-related properties. However, they can only express trace properties, *i.e.*, properties of individual execution traces. Unfortunately, they cannot specify properties over sets of execution traces, which is essential for many relevant security and planning specifications. On the other hand, HyperLTL was developed as an extension of LTL to describe collective behaviour relating multiple execution traces simultaneously. HyperLTL utilizes trace variables to specify the execution traces and uses the $\forall$ and $\exists$ quantifiers preceding a quantifier-free formula to specify on which traces the atomic propositions must hold. In the following, we present the syntax and semantics of HyperLTL in detail.
Syntax. We consider HyperLTL with the syntax:

$$\begin{aligned} \phi &::= \exists\pi.\phi \mid \forall\pi.\phi \mid \psi \\ \psi &::= p_\pi \mid \neg\psi \mid \psi \vee \psi \mid \mathsf{X}\psi \mid \psi \,\mathsf{U}\, \psi. \end{aligned}$$

The key distinction over LTL formulae is the introduction of trace quantifiers $\exists$ and $\forall$. The quantifier $\exists\pi$ stands for "for some trace π" while the quantifier $\forall\pi$ stands for "for all traces π", respectively. The variable ψ generates standard LTL formulae with the exception that atomic propositions can refer to distinct trace variables. Hence, for every proposition $p \in \mathcal{AP}$ and trace variable π, we use p_π to express that proposition p is referring to the trace π. A trace variable occurs free in a HyperLTL formula, if it is not bounded by any trace quantifier, *i.e.*, if no atomic propositions corresponding to the trace variable occur in ψ. A HyperLTL formula with no free variable is called closed.

Semantics. Since HyperLTL formulae express the properties of multiple trace variables, one requires assigning these trace variables to specific traces for reasoning about the satisfaction of the formula. Let $\mathcal{V} = \{\pi_1, \pi_2, \ldots\}$ be an infinite set of trace variables. The semantics of a HyperLTL formula ψ is defined over a set T of traces and a trace valuation function $\Pi : \mathcal{V} \to \Sigma^\omega$ that maps all the free trace variables occurring in the formula ψ to traces in the set Σ^ω. We use $\Pi[\pi \to \sigma]$ to express the trace valuation function Π' that agrees with Π for all trace variables except π and $\Pi'(\pi) = \sigma$. We define the trace valuation suffix $\Pi[i, \infty]$ as $\pi \mapsto \Pi(\pi)(i, \infty)$, i.e. $\Pi[i, \infty]$ maps π to the i-suffix of the trace mapped to π by Π. We say that a HyperLTL formula ψ is satisfiable over a given set T of traces and trace valuation function $\Pi : \mathcal{V} \to \Sigma^\omega$, and we write $\Pi \models_T \phi$ if one of the following holds:

- $\phi = \exists \pi.\psi$ and there is $\sigma \in T$ such that $\Pi[\pi \to \sigma] \models_T \psi$;
- $\phi = \forall \pi.\psi$ and for all $\sigma \in T$, we have $\Pi[\pi \to \sigma] \models_T \psi$;
- $\phi = p_\pi$ and $p \in \Pi(\pi)(0)$;
- $\phi = \neg\phi$ and $\Pi \not\models_T \phi$;
- $\phi = \psi_1 \vee \psi_2$ and $\Pi \models_T \psi_1$ or $\Pi \models_T \psi_2$;
- $\phi = \mathsf{X}\psi$ and $\Pi[1, \infty] \models_T \psi$;
- $\phi = \psi_1 \mathsf{U} \psi_2$ and there is $i \geq 0$ such that $\Pi[i, \infty] \models_T \psi_2$ and for all $0 \leq j < i$, we have that $\Pi[j, \infty] \models_T \psi_1$.

A closed HyperLTL formula ϕ is considered to be valid for a set of traces T, and we write $T \models \phi$ if the empty trace assignment satisfies the formula, *i.e.*, $\varnothing \models_T \phi$. We refer the interested readers to [30] for more details on syntax and semantics of HyperLTL properties.

2.4.3 Automata on (In)finite Traces

In general, complex logic specifications may also be represented by utilizing relevant automata, such as ω-automata to express specifications over infinite traces, or (non)deterministic finite automata to specify properties over finite traces. We present the definitions of such automata and describe their connection with LTL and HyperLTL specifications.

Definition 3. *An ω-automaton is a tuple $\mathcal{A} = (Q, q_0, \Sigma, \delta, \mathrm{Acc})$, where*

- *Q is a finite set of states;*
- *$q_0 \in Q$ is the initial state;*
- *Σ is the alphabet;*
- *$\delta \subseteq Q \times \Sigma \times Q$ is the state transition relation;*
- *Acc is an acceptance condition that varies according to the automaton considered;*

An automaton is said to be deterministic if, from every $q \in Q$, there exists only one successor state with an element $\sigma \in \Sigma$, *i.e.*, $\delta : Q \times \Sigma \to Q$. Otherwise, it is nondeterministic. An infinite sequence of input symbols $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots) \in \Sigma^\omega$ is called an infinite *word* or *trace*. An infinite *run* or *path* $\mathbf{q} = (q_0, q_1, \ldots) \in Q^\omega$ on the trace $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots)$ is an infinite sequence of states such that for every $m \geq 0$, we have $q_{m+1} = \delta(q_m, \sigma_m)$. We denote by $inf(\mathbf{q})$ the set of states in Q that is visited infinitely often during the run $\mathbf{q}$. A run $\mathbf{q}$ is said to be accepting if it satisfies the acceptance condition Acc, and the corresponding word $\sigma(\mathbf{q})$ is said to be accepted by $\mathcal{A}$. The language of $\mathcal{A}$, denoted by $\mathcal{L}(\mathcal{A})$, comprises of all the words accepted by $\mathcal{A}$.

There exist different kinds of acceptance conditions for ω-automata, such as Büchi [23], Rabin [101] , Streett [118], and Müller [86] acceptance conditions. In this thesis, we mainly work with Büchi and Streett automata. A nondeterministic Büchi automaton (NBA), denoted by $\mathcal{A}^b = (Q, q_0, \Sigma, \delta, F)$, is defined similarly to Definition 3, but with the acceptance condition defined by $F \subseteq Q$, which is a set of accepting states. Moreover, the transition relation is nondeterministic, *i.e.*, $\delta : Q \times \Sigma \to 2^Q$. An infinite run is said to be accepting iff $inf(\mathbf{q}) \cup F \neq \emptyset$. The deterministic variant, called deterministic Büchi automata (DBA) is defined similarly, but with a deterministic transition function $\delta : Q \times \Sigma \to Q$. On the other hand, a deterministic Streett automaton (DSA), denoted by $\mathcal{A}^s = (Q, q_0, \Sigma, \delta, Acc)$, has an acceptance condition that is defined by pairs of states, *i.e.*, $Acc = \{< E_1, F_1 >, < E_2, F_2 >, \ldots, < E_z, F_z >\}$, where $< E_i, F_i >$ with $E_i, F_i \subseteq Q, \forall i \in \{1, \ldots, z\}$. For simpler presentation, we define the sets $E = \{E_1, E_2, \ldots, E_z\}$ and $F = \{F_1, F_2, \ldots, F_z\}$ where $< E_i, F_i > \in Acc$, $\forall i \in \{1, \ldots, z\}$. A run $\mathbf{q}$ is said to be an accepting run for $\mathcal{A}^s$ if for all $E_i \in E$ and $F_i \in F$, $i \in \{1, \ldots, z\}$, we have $inf(\mathbf{q}) \cap E_i = \emptyset$ or $inf(\mathbf{q}) \cap F_i \neq \emptyset$. We now define deterministic finite automata (DFA), which we utilize to express specifications over finite traces.

Definition 4. *A deterministic finite automaton (DFA) is a tuple $\mathcal{A}^f = (Q, q_0, \Sigma, \delta, F)$, where Q is a finite set of states, $q_0 \in Q$ is the initial state, Σ is a finite set of input symbols called alphabet, $\delta : Q \times \Sigma \to Q$ is the transition function and $F \subseteq Q$ represents the accepting states.*

A finite trace $\boldsymbol{\sigma}_f = (\sigma_0, \ldots, \sigma_{n-1}) \in \Sigma^*$ is said to be be accepted by DFA $\mathcal{A}^f$ if there exists a corresponding finite path $\mathbf{q}_f = (q_0, q_1, \ldots, q_n) \in Q^{n+1}$ such that $q_{m+1} = \delta(q_m, \sigma_m)$, $m \in \{1, \ldots, n\}$ and $q_F \in F$. Similar to ω-automata, we utilize the notation $\mathcal{L}(\mathcal{A}^f)$ to denote the accepting language of $\mathcal{A}^f$ that consists of all the finite traces that are accepted by $\mathcal{A}^f$.

Expressiveness of specifications. We first consider the expressiveness of specifications defined over infinite traces. ω-Automata such as nondeterministic Büchi automata and deterministic Streett automata can express all ω-regular specifications, and in general are more expressive than deterministic Büchi automata, which can only represent a subset of ω-regular specifications [124]. Moreover, LTL specifications also correspond to another subset of ω-regular specifications called star-free ω-regular specifications, and any LTL specification ψ may be converted to a corresponding Büchi or Streett automaton by utilizing appropriate automata conversion tools such as SPOT [44], or `ltl2dstar` [68]. It must be noted that ω-regular properties are all *trace* properties, *i.e.*, they capture the properties of individual execution traces. On the other hand, hyperproperties are *sets of traces* properties, *i.e.*, they capture the interactions between multiple execution traces. In this context, HyperLTL, which is a formalism used to express hyperproperties, extends LTL specifications and is, therefore, more expressive than LTL. In fact, HyperLTL subsumes LTL

and can express a subset of ω-regular hyperproperties [48]. LTL to Büchi automata construction techniques may also be adapted to obtain NBA corresponding to HyperLTL specifications [30]. In particular, for a given HyperLTL specification $\phi = \mu_1\pi_1 \ldots, \mu_p\pi_p\psi$, where $\mu_i \in \{\forall, \exists\}$, for all $i \in \{1, \ldots, p\}$, one constructs an NBA $\mathcal{A}^b_\psi$ corresponding to ψ, which is a quantifier-free LTL formula. However, in this case, to accommodate for the quantification of traces, a trace for such an automaton is a p-tuple of individual traces, denoted by $\tilde{\boldsymbol{\sigma}} = (\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_p)$, i.e., $\mathsf{unzip}(\tilde{\boldsymbol{\sigma}}) \in \mathcal{L}(\mathcal{A}_\psi)$ if $(\pi_1 \mapsto \boldsymbol{\sigma}_1, \ldots, \pi_p \mapsto \boldsymbol{\sigma}_p)$ is an accepting run of $\mathcal{A}^b_\psi$.

As mentioned, specifications over finite traces may either be expressed by DFA or LTL_f specifications. In general, one prefers to consider specifications represented as DFA since they are more expressive than LTL_f specifications [38]. Moreover, any LTL_f specification can be converted to an equivalent DFA by utilizing appropriate construction tools such as MONA [57]. Note that in this thesis, we consider deterministic finite automata rather than nondeterministic finite automata (NFA). This is without any loss of generality since DFA are equally expressive and one can obtain a DFA corresponding to any NFA by means of power set construction [102].

2.5 Barrier Certificates

Having introduced all the relevant specifications that are considered in this thesis, we now provide some preliminary results available in existing literature for solving Problems (1)-(6) by utilizing barrier certificate-based approaches.

2.5.1 Barrier Certificates for Safety

In this subsection, we present barrier certificate-based approaches for the safety analysis of discrete-time (stochastic) systems, obtained by adapting results from [96, 97]. We first provide solutions to Problems 1-2.

Definition 5. *We say that a function $\mathbb{B} : X \to \mathbb{R}$ is a barrier certificate (BC) for a dt-DS $\mathfrak{S} = (X, f)$ with respect to a set of initial states $X_0 \subseteq X$, and a set of unsafe states $X_u \subseteq X$, if:*

$$\mathbb{B}(x) \leq 0, \qquad \textit{for all } x \in X_0, \tag{2.8}$$

$$\mathbb{B}(x) > 0, \qquad \textit{for all } x \in X_u, \tag{2.9}$$

$$\mathbb{B}(f(x)) - \mathbb{B}(x) \leq 0 \qquad \textit{for all } x \in X. \tag{2.10}$$

It is simple to see that barrier certificates provide sufficient conditions ensuring that the state sequences started in the initial set $X_0 \subset X$ never reach the unsafe region $X_u \subseteq X$. Since condition (2.10) requires the barrier certificate to be non-increasing at every time step, it ensures that the state sequences never cross the level set $\mathbb{B}(x)$=0. In other words, the state sequences of the dt-DS $\mathfrak{S}$ always remain in the safe regions. Figure 2.1 demonstrates the safety verification using barrier certificates. For additional information and detailed proofs, we refer the interested readers to [96].

Control barrier certificates (CBCs) for synthesizing controllers enforcing safety specifications may be obtained in a similar manner, defined as follows.

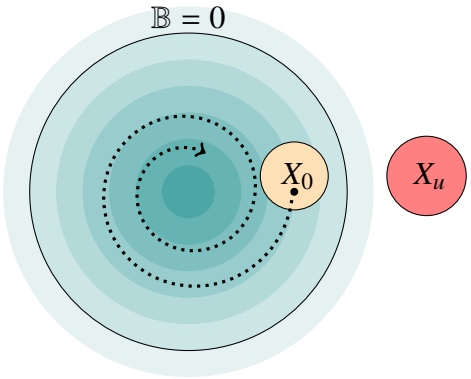


Figure 2.1: Safety verification using barrier certificates.

Definition 6. *Consider a dt-CS $\mathfrak{S} = (X, U, f)$, a set of initial states $X_0 \subseteq X$ and a set of unsafe states $X_u \subseteq X$. We say that $\mathbb{B} : X \to \mathbb{R}$ is a control barrier certificate (CBC) for dt-CS $\mathfrak{S}$ if:*

$$\mathbb{B}(x) \leq 0, \qquad \text{for all } x \in X_0, \tag{2.11}$$
$$\mathbb{B}(x) > 0, \qquad \text{for all } x \in X_u, \tag{2.12}$$

and for all $x \in X$, there exists $u \in U$ such that:

$$\mathbb{B}(f(x,u)) - \mathbb{B}(x) \leq 0. \tag{2.13}$$

The proof follows similarly to that of Definition 5, but with control input being selected according to condition (2.13).

In the context of stochastic systems, barrier certificates take the form of *supermartingale* functions. In probability theory, a supermartingale is a sequence of random variables whose conditional expectation of the next value at any time instant is always smaller than the current value, irrespective of the prior values. In this regard, barrier certificates for stochastic systems are non-negative real-valued functions that satisfy the supermartingale condition, *i.e.*, the expected value of the barrier certificate is non-increasing at every time step. In the following, we introduce barrier certificates for safety verification [97] and control barrier certificates for synthesis [63], and utilize these definitions to provide probabilistic guarantees for safety satisfaction, consequently solving Problems 3-4.

Definition 7. *We say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a* barrier certificate *for the dt-SS $\mathfrak{S} = (X, \varsigma, f)$ with respect to a set of initial states $X_0 \subseteq X$, a set of unsafe states $X_u \subseteq X$ if there exists a constant $0 \leq \varepsilon \leq 1$ such that the following conditions hold:*

$$\mathbb{B}(x) \leq \varepsilon, \qquad \text{for all } x \in X_0, \tag{2.14}$$
$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in X_u, \tag{2.15}$$
$$\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x] - \mathbb{B}(x) \leq 0, \qquad \text{for all } x \in X. \tag{2.16}$$

We now utilize Definition 7 to provide probabilistic bounds with which the dt-SS satisfies the safety specification.

Theorem 1. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$. Let $\mathbb{B}$ be a barrier certificate satisfying conditions* (2.14)-(2.16) *for some $0 \leq \varepsilon \leq 1$. Then the probability that the solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X_0$ does not reach unsafe region X_u is bounded by*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin X_u \text{ for all } t \in \mathbb{N} \mid x_0\} \geq 1 - \varepsilon. \tag{2.17}$$

From Theorem 1, it can be easily inferred that the existence of a barrier certificate according to Definition 7 guarantees a solution to Problem 3 with a probability of $\varkappa = 1 - \varepsilon$.

Proof. According to condition (2.15), $X_u \subseteq \{x \in X \mid \mathbb{B}(x) \geq 1\}$. Therefore, it follows that

$$\mathbb{P}\{x(t) \in X_u \text{ for some } t \in \mathbb{N} \mid x_0\} \leq \mathbb{P}\{\sup_{t \in \mathbb{N}} \mathbb{B}(x(t)) \geq 1 \mid x_0\}.$$

Now, due to condition (2.16), we have that $\mathbb{B}$ is a non-negative supermartingale, and from [70, Theorem 12, Chapter II] it follows that

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_u \text{ for some } t \in \mathbb{N} \mid x_0\} \leq \varepsilon.$$

By means of complementation, we obtain the lower bound of (2.17). □

Definition 8. *Consider a dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$, a set of initial states $X_0 \subseteq X$, and a set of unsafe states $X_u \subseteq X$. We say that $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a control barrier certificate (CBC) for dt-SCS $\mathfrak{S}$ if:*

$$\mathbb{B}(x) \leq \varepsilon, \qquad \text{for all } x \in X_0, \tag{2.18}$$
$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in X_u, \tag{2.19}$$

and for all $x \in X$, there exists $u \in U$ such that:

$$\mathbb{E}[\mathbb{B}(f(x, u, \varsigma)) \mid x, \nu] - \mathbb{B}(x) \leq 0. \tag{2.20}$$

Theorem 2. *Consider a dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$. Let $\mathbb{B}$ be a control barrier certificate satisfying conditions* (2.18)-(2.20)*. Then the probability that the solution process $\mathbf{x}_{x_0,\varpi}$ starting from an initial condition $x_0 \in X_0$ under a controller ϖ, obtained via the satisfaction of condition* (2.20)*, satisfies the safety specification with the probability bounds given by:*

$$\mathbb{P}\{\mathbf{x}_{x_0,\varpi}(t) \notin X_u \text{ for all } t \in \mathbb{N} \mid x_0\} \geq 1 - \varepsilon. \tag{2.21}$$

The proof of Theorem 2 follows similarly to that of Theorem 1 under the controller ϖ. It must be mentioned that while supermartingale conditions are necessary in Definitions 7 and Definitions 8 to provide probabilistic guarantees over infinite time horizons, it is possible to relax these conditions by adding an offset of $c > 0$ in the right-hand side of conditions (2.14) and (2.20), respectively. This condition is known as a c-martingale condition [117]. However, this comes at the cost of providing probabilistic guarantees over finite time horizons. We state the following corollary for the verification problem in a dt-SS $\mathfrak{S}$. Note that this result may be similarly adapted to the controller synthesis case as well, yielding identical probabilistic guarantees.

Corollary 1. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$. Let $\mathbb{B}$ be a c-martingale barrier certificate for $\mathfrak{S}$. Then the lower bound on the probability that the solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X_0$ does not reach the unsafe regions within time horizon $[0, T_d)$ is obtained as*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin X_u \text{ for all } t \in [0, T_d) \mid x_0\} \geq 1 - (\varepsilon + cT). \tag{2.22}$$

Remark 3. *Note that one may also reformulate condition* (2.16) *to write*

$$\mathbb{E}[\mathbb{B}(f(x, \varsigma)) \mid x] - \kappa(\mathbb{B}(x)) \leq 0, \tag{2.23}$$

where $\kappa \in \mathcal{K}_\infty$, with $\kappa \leq \mathrm{id}$ for the verification problem. When $\kappa = \mathrm{id}$, we recover condition (2.16)*. This condition may also be suitably adapted for the controller synthesis problem.*

Remark 4. *In all of the aforementioned definitions, it is implicitly assumed that $X_0 \wedge X_u = \varnothing$. When this is not the case, there does not exist a (control) barrier certificate due to the conflict between the first two conditions. In such a case, the system can start from the unsafe regions and will trivially violate the safety property.*

2.5.2 Barrier Certificates for Reachablility

In this subsection, we utilize barrier certificates to provide verification guarantees over reachability specifications and consequently provide solutions to Problems 5-6. However, to do this, one needs the following assumption.

Assumption 1. *For a (stochastic) dynamical system $\mathfrak{S}$ that starts from the initial condition $x_0 \in X$, we have that $\mathbf{x}_{x_0}(t) \in X$ for all $t \in \mathbb{N}$.*

Remark 5. *The relevance of Assumption 1 has been demonstrated in [58]. Intuitively, this assumption is natural in many physical applications where state variables are naturally constrained to a compact set and do not leave this set in their operating envelope. For stochastic dynamical systems, this assumption can be supported by analyzing the stopped process (see [58]). Given a solution process $\mathbf{x}_{x_0}$ of dt-SS $\mathfrak{S}$, we define a stopped process $\bar{\mathbf{x}}_{x_0}$ as*

$$\bar{\mathbf{x}}_{x_0}(t) = \begin{cases} \mathbf{x}_{x_0}(t), & \text{for } t < \tau, \\ \mathbf{x}_{x_0}(\tau - 1), & \text{for } t \geq \tau, \end{cases}$$

where $\tau \in \mathbb{N}$ is the first exit time of $\mathbf{x}_{x_0}$ from X.

We now state the following definition of barrier certificates for reachability verification of discrete-time dynamical systems, adapted from the continuous-time version presented in [99, Theorem 3.5].

Definition 9. *Consider a dt-DS $\mathfrak{S} = (X, f)$, the set of initial states $X_0 \subseteq X$, and the set of target states $X_R \subseteq X$. Let $\mathfrak{S}$ satisfy Assumption 1. We say that a function $\mathbb{B} : X \to \mathbb{R}$ is a barrier*

certificate for $\mathfrak{S}$ with respect to X_0 and X_R if $\mathbb{B}$ is bounded from below on X and the following conditions hold:

$$\mathbb{B}(x) \leq 0, \qquad \text{for all } x \in X_0, \tag{2.24}$$
$$\mathbb{B}(x) > 0, \qquad \text{for all } x \in \partial X \setminus \partial X_R, \tag{2.25}$$
$$\mathbb{B}(f(x)) - \mathbb{B}(x) \leq -\delta, \qquad \text{for all } x \in \overline{X \setminus X_R}, \tag{2.26}$$

where $\delta > 0$ is a small positive number used to ensure a strict decrease of $\mathbb{B}$.

Existence of a BC as in Definition 9 guarantees that the dt-DS $\mathfrak{S} = (X, f)$ satisfies the reachability specification, *i.e.*, for a state run $\mathbf{x}_{x_0}$ starting from $x_0 \in X_0$, there exists some $T \in \mathbb{N}$ such that $\mathbf{x}_{x_0}(T) \in X_R$. To see this, consider the fact that the state runs of $\mathfrak{S}$ must eventually leave $X \setminus X_R$ in finite time since $\mathbb{B}$ is bounded from below. Now, suppose that $\mathbf{x}_{x_0}$ leaves this set without entering X_R. This can only happen if the $\mathbf{x}_{x_0}$ enters the boundary set $\partial X \setminus \partial X_R$. However, due to conditions (2.24) and (2.26), we have that $\mathbb{B}(\mathbf{x}_{x_0}(t)) \leq 0, \forall t \in \mathbb{N}$. This results in a contradiction with condition (2.25), and therefore $\mathbf{x}_{x_0}$ cannot reach $\partial X \setminus \partial X_R$. Moreover, since $\mathfrak{S}$ is forward invariant in X, $\mathbf{x}_{x_0}$ must enter X_R, thus satisfying the reachability specification. We now present a similar definition of barrier certificates for stochastic dynamical systems to provide probabilistic guarantees over reachability specifications.

Definition 10. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$ that satisfies Assumption 1. Then, we say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a barrier certificate for the dt-SS $\mathfrak{S}$ with respect to a set of initial states $X_0 \subseteq X$ and a set of target states $X_R \subseteq X$ if there exist constants $0 \leq \varepsilon \leq 1$ and $\delta > 0$ such that the following conditions hold:*

$$\mathbb{B}(x) \leq \varepsilon, \qquad \text{for all } x \in X_0, \tag{2.27}$$
$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in \partial X \setminus \partial X_R, \tag{2.28}$$
$$\mathbb{E}[\mathbb{B}(f(x, \varsigma)) \mid x] - \mathbb{B}(x) \leq -\delta, \qquad \text{for all } x \in \overline{X \setminus X_R}. \tag{2.29}$$

This definition can then be utilized to obtain a lower bound on the probability that a dt-SS $\mathfrak{S}$ satisfies the reachability specification over *unbounded-time* horizons.

Theorem 3. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$ satisfying Assumption 1. Let $\mathbb{B}$ be a barrier certificate for $\mathfrak{S}$ satisfying conditions* (2.27)-(2.29) *with some $0 \leq \varepsilon \leq 1$. Then the probability that the solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X_0$ reaches the target region X_R is bounded by*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_R \text{ for some } t \in \mathbb{N} \mid x_0\} \geq 1 - \varepsilon. \tag{2.30}$$

Proof. The solution processes of $\mathfrak{S}$ may either reach the boundary $\partial X \setminus \partial X_R$ without entering X_R, or may never reach $\partial X \setminus \partial X_R$ after reaching X_R. Now, due to conditions (2.27)-(2.29) and Theorem 1 with $X_u = \partial X \setminus \partial X_R$, one has a lower bound on the probability that the solution process $\mathbf{x}_{x_0}$ starting from x_0 does not reach the boundary set $\partial X \setminus \partial X_R$ as

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin \partial X \setminus \partial X_R \text{ for all } t \in \mathbb{N} \mid x_0\} \geq 1 - \varepsilon.$$

Now, under the condition that the solution processes do not enter $\partial X \setminus \partial X_R$, we can provide an almost sure guarantee that the solution process reaches the target set X_R. This is due to condition (2.29), which imposes a stronger supermartingale condition that requires a strict decrease in the expected value of the barrier certificate. Since the barrier certificate is bounded below (due to non-negativity), by the virtue of Doob's martingale convergence theorem [43], we have that the barrier certificate convergences almost surely to a state x where $\mathbb{B}(x)$ reaches its minimum value. Let $x \in \overline{X \setminus X_R}$. Then by condition (2.29), the expected value of the barrier certificate must strictly decrease. However, this is not possible, and therefore, $x \notin \overline{X \setminus X_R}$ and the solution process $\mathbf{x}_{x_0}$ must leave $\overline{X \setminus X_R}$. Since the probability of not leaving $\overline{X \setminus X_R}$ via the boundary set $\partial X \setminus \partial X_R$ is greater than $1 - \epsilon$, the solution process $\mathbf{x}_{x_0}$ must leave the set $\overline{X \setminus X_R}$ by entering X_R with probability greater than $1 - \varepsilon$. Therefore, we obtain the probability bound of (2.30). □

Remark 6. *Note that barrier certificates as in Definitions 9 and 10 provide reach-while-avoid guarantees. They ensure that the system avoids the set $\partial X \setminus \partial X_R$ while reaching the set X_R. Given a set of unsafe states $X_u \subseteq X$, one can replace $\partial X \setminus \partial X_R$ with X_u in conditions* (2.25) *and* (2.29) *to give guarantees of reaching the target set of states X_R while avoiding X_u.*

We next formulate another definition of barrier certificates for reachability in the case of stochastic systems when no unsafe region to avoid is provided. This formulation can provide stronger almost-sure guarantees and is analogous to supermartingale ranking functions used in [24, Definition 4.3.2].

Definition 11. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$. Suppose Assumption 1 holds for $\mathfrak{S}$. Then, we say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a barrier certificate for $\mathfrak{S}$ with respect to a set of initial states $X \setminus X_R$ and a set of target states $X_R \subseteq X$ if there exist constants $\varepsilon, \delta > 0$, such that the following conditions hold:*

$$\mathbb{B}(x) \geq \varepsilon, \qquad \text{for all } x \in X \setminus X_R, \tag{2.31}$$

$$\mathbb{B}(x) < \varepsilon, \qquad \text{for all } x \in X_R, \tag{2.32}$$

$$\mathbb{E}[\mathbb{B}(f(x, \varsigma)) \mid x] - \mathbb{B}(x) \leq -\delta, \qquad \text{for all } x \in X \setminus X_R. \tag{2.33}$$

Definition 11 can then be utilized to show that the dt-SS $\mathfrak{S}$ satisfies reachability specification with probability 1.

Theorem 4. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$ satisfying Assumption 1. Let $\mathbb{B}$ be a barrier certificate for $\mathfrak{S}$ satisfying conditions* (2.31)-(2.33) *with some $\varepsilon > 0$. Then a solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X \setminus X_R$ reaches the target region X_R with probability* 1*, i.e.,*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_R \text{ for some } t \in \mathbb{N} \mid x_0\} = 1. \tag{2.34}$$

Proof. Since the barrier certificate $\mathbb{B}$ is a non-negative supermartingale that is strictly decreasing due to condition (2.33), from Doob's martingale convergence theorem [43] it follows that the barrier certificate almost surely converges to some state x such that $\mathbb{B}(x)$ reaches its minimum value. Moreover, due to Assumption 1 and conditions (2.31) and (2.32), we have $x \in X_R$. Therefore, we have that a solution process $\mathbf{x}_{x_0}$ starting from $x_0 \in X_0$ eventually reaches X_R almost surely, which implies that $\mathbf{x}_{x_0}(t) \in X_R$ for some $t \in \mathbb{N}$ with probability 1, as obtained in (2.34). □

The set of initial states for a system admitting barrier certificates as in Definition 11 can be anywhere within the set X. If a solution process starts in X_R, the system trivially satisfies the reachability specification. For a solution process that starts in $X \setminus X_R$, the existence of a barrier certificate guarantees convergence and reachability into X_R. However, in the case of barrier certificates as defined in Definition 10, one requires condition $X_0 \cap (\partial X \setminus \partial X_R) \neq \emptyset$. The reasoning for this is similar to that of Remark 4. It is also important to note that conditions (2.29) and (2.33) in Definitions 10 and 11, respectively, impose a stronger supermartingale condition than the one for safety. These conditions require a strict decrease in barrier certificate values, which ensures the convergence of the barrier certificate so that it eventually reaches the target set.

Remark 7. *Note that we abuse the notation to utilize a unified representation $\mathbb{B}$ for (control) barrier certificates for both safety and reachability specifications, as well as for both non-stochastic and stochastic (control) systems. In the remainder of the thesis, the meaning of the notation will be made clear with context.*

Comparison with inductive invariants. Safety verification of software and hardware systems is traditionally performed by utilizing inductive invariants [125, 33]. Inductive invariants are properties that hold universally along the reachable set of the state space, starting from the initial state. As the name suggests, such properties can be proven via mathematical induction. Barrier certificates are a continuous-state analogue to inductive invariants. In the case of non-stochastic systems, the property that $\mathbb{B}(x) \leq 0$ is an inductive invariant, since it holds true in the initial set of the system, and continues to remain true at every state that can be reached by the system. For stochastic dynamical systems, barrier certificates take the form of expectation invariants [25], *i.e.*, they employ *supermartingale* conditions to ensure that they are non-increasing in expectation at each time step, which can provide lower bounds for the probabilities of satisfying safety and reachability properties over potentially *unbounded* time horizons.

Chapter 3

Compositional Construction of Control Barrier Certificates

3.1 Introduction

Many safety-critical systems in the real world do not evolve in a deterministic manner due to disturbances. As a result, they are modeled as stochastic control systems. Formal verification and synthesis of such systems against logic specifications has therefore received a lot of attention in recent years [122, 71, 73, 97]. While existing methods perform well over lower-dimensional systems, the problem is especially more challenging in large-scale stochastic control systems that model many application scenarios such as power networks, air traffic control, etc. For example, consider the control barrier certificate-based approach for synthesis against safety specifications, as presented in Chapter 2. Typically, the search for control barrier certificates is performed by restricting them to a certain parametric form (*e.g.*, polynomial functions of a fixed degree) and then searching for their corresponding coefficients under certain assumptions (*e.g.*, coefficients of the polynomial functions). Although lower-dimensional systems usually admit simple control barrier certificates and the corresponding search is relatively easy by utilizing existing numerical tools (*e.g.*, sum-of-squares optimization), this is extremely difficult, if not impossible, for large-scale systems. To tackle such computational complexity involved in the context of large-scale stochastic control systems, in this chapter, we propose a compositional framework for the construction of control barrier certificates. To do so, we consider the large-scale stochastic control system as an interconnected one composed of several smaller subsystems, and search for so-called *control sub-barrier certificates* for subsystems, along with corresponding local controllers. These control sub-barrier certificates may be compiled together under certain *compositionality conditions* to obtain control barrier certificates for the interconnected system. By doing so, we are able to establish lower bounds on the probability with which the large-scale stochastic control system remains in safe regions over (in)finite time horizons.

3.1.1 Related Literature

Large-scale Stochastic Control Systems

Existing results on the verification and controller synthesis of large-scale stochastic systems have primarily focused on abstraction-based techniques. Existing approaches include obtaining probabilistic guarantees against safety and reachability specifications in the context of discrete-time stochastic hybrid systems [1], an abstraction-based framework for verification and synthesis of discrete-time stochastic systems [71, 73]. These methods unfortunately suffer from the state-explosion problem, which makes them unsuitable for large systems. These issues have been partly alleviated by utilizing adaptive gridding-based approaches [115] or by leveraging incremental stability properties to obtain input-set abstractions for stochastic control systems [136]. More recently, compositional frameworks have been proposed for obtaining finite abstractions of large-scale discrete-time stochastic control systems in [72, 120]. Instead of obtaining finite abstractions for large-scale systems monolithically, such approaches decompose large-scale systems into smaller subsystems and obtain finite abstractions for the subsystems instead.

Control Barrier certificates for Stochastic Systems

Control barrier certificate-based approaches are naturally discretization-free, which makes them more scalable compared to abstraction-based techniques. Existing results in this direction include safety verification of continuous-time stochastic hybrid systems [97, 61, 129]. A verification approach for Markov decision processes using barrier certificates is proposed by [2]. Verification via barrier certificates for switched stochastic systems was proposed in [3]. Control barrier functions for stochastic affine control systems under incomplete information was presented in [28, 29]. Verification and control for finite-time safety of stochastic systems using barrier functions are discussed by [110, 109]. Recently, verification and synthesis of discrete-time stochastic control systems against logic properties in finite-time horizons via control barrier certificates are presented by [62], and [63], respectively. Despite being discretization-free, these methods are intractable for large-scale control systems.

Compositional Control Barrier Certificates

We would like to mention that some recent works in the literature investigate the compositional construction of control barrier certificates for different classes of systems than we consider in this thesis. For instance, compositional construction of control barrier functions via max-small gain compositionality conditions was presented in [64] for non-stochastic interconnected systems. Compositional construction of control barrier certificates for large-scale stochastic control systems are also presented in [87, 88, 89], but these results are reserved for continuous-time systems. Moreover, [88, 89] rely on a different compositional scheme than the one presented here, namely sum-type small gain conditions. Unfortunately, those conditions are formulated in terms of "almost" linear gains and require subsystems to have a (nearly) linear behaviour, making it much more conservative than our proposed approach here.

3.1.2 Contributions

In Chapter 2, we introduced safety specifications and a control barrier certificate-based approach for the synthesis of controllers ensuring safety specifications for stochastic control systems. In this chapter, we focus on the control of large-scale stochastic control systems for safety specifications. To deal with large-scale stochastic control systems, we introduce some compositional frameworks for the construction of control barrier certificates and corresponding controllers. In particular, we provide two different compositional methodologies (*i.e.*, max-type small-gain, and dissipativity approaches) for the construction of control barrier certificates for interconnected systems composed of several smaller subsystems. The proposed techniques reduce the construction of control barrier certificates for the interconnected system to that of control sub-barrier certificates for the subsystems. Correspondingly, local controllers are computed for the subsystems. Then, by leveraging some sufficient max-type small-gain and dissipativity-type compositionality conditions, we obtain the control barrier certificates for the interconnected system by utilizing the control sub-barrier certificates for the subsystems. Moreover, the local controllers are applied to the interconnected system in a decentralized fashion to the interconnected system. Finally, the proposed approach enables the satisfaction of safety specifications with some (tight) probability lower bounds.

This chapter is organized into three sections. Section 3.2 introduces the decomposition of a large-scale stochastic control system into control subsystems. In Section 3.3, we present the compositional framework for the construction of control barrier certificates for interconnected discrete-time stochastic control systems by utilizing max-type small-gain type compositionality conditions. Moreover, by utilizing the obtained control barrier certificates and corresponding controllers, we establish probability lower bounds on the satisfaction of safety specifications over *finite time horizons*. We also provide two different approaches for the computation of suitable control sub-barrier certificates for the subsystems, one based on sum-of-squares (SOS) optimization, and the other based on counterexample-guided inductive synthesis (CEGIS). Finally, we demonstrate our results with the help of two case studies: a room temperature regulation problem in a building consisting of 1000 rooms, and a *fully-connected* Kuramoto oscillator network with 300 oscillators.

On the other hand, Section 3.4 proposes a different compositional framework for the construction of control barrier certificates for large-scale stochastic control systems by utilizing dissipativity-type compositionality conditions. This framework utilizes the interconnection topology together with the dissipativity properties of the subsystems. Depending on the structure of the interconnections (*e.g.*, skew-symmetric), one is able to satisfy the compositionality condition without any restrictions on the number of subsystems or corresponding gains. The main goal here is to synthesize controllers and correspondingly compute probability lower bounds for the satisfaction of safety specifications over *infinite time horizons*.

Conventionally, in order to satisfy the compositionality condition, the required parameters for finding suitable control sub-barrier certificates are pre-selected and the compositionality condition is checked a posteriori. While this method can provide tractable results in certain scenarios for systems with specific interconnection structures, it is not particularly useful in large-scale networks where structural properties of the subsystems (*e.g.*, dissipativity properties)

are not apparent. Besides, since control sub-barrier certificates are not optimized with respect to the compositionality condition, obtained results can be conservative. In order to provide scalable, less conservative results, we employ a distributed optimization method based on an alternating direction method of multipliers (ADMM) algorithm which allows us to break down a large optimization problem into several smaller sub-problems which can be easier to handle. The solution to the optimization problem provides us with suitable control sub-barrier certificates along with local controllers, allowing the computation of control barrier certificates for the interconnected system. For systems with polynomial dynamics, we show that the ADMM algorithm can be utilized in conjunction with sum-of-squares (SOS) optimization in order to obtain control sub-barrier certificates and corresponding local controllers. Finally, we present the theoretical comparisons between the dissipativity-based compositionality framework and the small-gain one. We also demonstrate the effectiveness of our proposed results by applying them to a room temperature network in a circular building containing 300 rooms.

We must mention that the results presented in this chapter appear in our publications [4, 5, 6]. The former two results have been published at the 21st IFAC World Congress and in the journal Transactions of Automatic Control, respectively. The latter has been accepted in the journal Nonlinear Analysis: Hybrid Systems. These are joint works with Abolfazl Lavaei and Majid Zamani. The author of the thesis has established the results and written the drafts. Abolfazl Lavaei contributed to initial discussions, some results and experiments, revision of the drafts as well as mentoring. Majid Zamani supervised the work.

3.2 Stochastic Control Subsystems

In this chapter, we focus on the safe control large-scale stochastic control systems (dt-SCS) $\mathfrak{S} = (X, U, \varsigma, f)$, as defined in Definition 2 that can be considered as an interconnected system consisting of several smaller stochastic control subsystems, defined as follows.

Definition 12. *A discrete-time stochastic control subsystem (or simply, subsystem) is a tuple*

$$\mathfrak{S} = (X, U, W, \varsigma, f, Y, h), \tag{3.1}$$

where,

- $X \subseteq \mathbb{R}^n$ *is a Borel space as the state set of the system. The tuple* $(X, \mathcal{B}(X))$ *is the measurable state space where* $\mathcal{B}(X)$ *denotes the Borel sigma-algebra on the state space;*
- $U \subseteq \mathbb{R}^m$ *and* $W \subseteq \mathbb{R}^p$ *are Borel spaces as* external *and* internal *input sets of the system. The tuples* $(U, \mathcal{B}(U))$ *and* $(W, \mathcal{B}(W))$ *are the measurable external and internal input sets, respectively, with* $\mathcal{B}(U)$ *and* $\mathcal{B}(W)$ *denoting their respective Borel sigma-algebras;*
- $\varsigma := \{\varsigma(t) : \Omega \to \mathcal{V}_\varsigma,\ t \in \mathbb{N}\}$ *is a sequence of independent and identically distributed (i.i.d.) random variables from a sample space* Ω *to the measurable space* $(\mathcal{V}_\varsigma, \mathcal{F}_\varsigma)$*;*
- $f : X \times U \times W \times \mathcal{V}_\varsigma \to X$ *is a measurable function that characterizes the state evolution of* $\mathfrak{S}$*;*

- *$Y \subseteq \mathbb{R}^r$ is a Borel space as the internal output set of the system;*
- *$h : X \to Y$ is a measurable function that maps a state $x \in X$ to its output $y = h(x)$.*

We associate sets $\mathcal{U}$ and $\mathcal{W}$ to respectively sets U and W as collections of external and internal input sequences $\{\nu(t) : \Omega \to U,\ t \in \mathbb{N}\}$ and $\{w(t) : \Omega \to W,\ t \in \mathbb{N}\}$. Both $\nu(t)$ and $w(t)$ are independent of the random variable $\varsigma(s)$ for all $s, t \in \mathbb{N}$ and $s \geq t$. It should be noted that $\nu(t)$ is the external input of subsystem $\mathfrak{S}$ that should be designed to enforce the property of interest, whereas $w(t)$ is the internal input that is used to provide interconnections between several subsystems in a large-scale interconnected system, as will be explained later. The state evolution of subsystem $\mathfrak{S}$ for a given initial state $x(0) \in X$, and input sequences $\{\nu(t) : \Omega \to U,\ t \in \mathbb{N}\}$ and $\{w(t) : \Omega \to W,\ t \in \mathbb{N}\}$ is characterized by:

$$\mathfrak{S} : \begin{cases} \mathbf{x}(t+1) = f(\mathbf{x}(t), \nu(t), w(t), \varsigma(t)), \\ y(t) = h(\mathbf{x}(t)), \end{cases} \tag{3.2}$$

for any $t \in \mathbb{N}$. For a given initial state $x_0 \in X$, $\nu(\cdot) \in \mathcal{U}$ and $w(\cdot) \in \mathcal{W}$, a random sequence $\mathbf{x}_{x_0,\nu,w} : \Omega \times \mathbb{N} \to X$ denotes the solution process of $\mathfrak{S}$ under the influence of the external input ν, the internal input w, and started from the initial state x_0. Finally, we consider that the control of subsystem $\mathfrak{S}$ is enforced by a controller similar to that defined in Section 2.2 of Chapter 2.

The main focus of this chapter is on the control of large-scale systems without any internal inputs or outputs as in Definition 2 which can be regarded as a composition of smaller subsystems as in Definition 12. Note that although there is an output in the definition of subsystem in (3.2), full state information is assumed to be available for the large-scale system (*i.e.,* its output map is identity) for the sake of controller synthesis. Hence, we consider the large-scale dt-SCS $\mathfrak{S}$ without any output set or output map, as in Definition 2. More precisely, the role of the output in (3.2) is mainly for the sake of interconnecting subsystems, which will be discussed in more detail later in this chapter. However, for the sake of clarity, we present Figure 3.1 to illustrate a dt-SCS of Definition 2, which does not consist of any internal inputs or outputs, that is constructed by interconnecting smaller subsystems of Definition 12 consisting of both internal inputs and outputs. In particular, the two subsystems $\mathfrak{S}_1$ and $\mathfrak{S}_2$ are connected together through the internal inputs w_1, w_2 and outputs y_1, y_2, respectively. Notice that the external inputs u_1 and u_2 of subsystems $\mathfrak{S}_1, \mathfrak{S}_2$ also serve as the external input to the interconnected dt-SCS $\mathfrak{S}$ as $u = [u_1; u_2]$. Therefore, by designing appropriate controllers for the subsystems in a decentralized fashion, one subsequently obtains the overall controller for the interconnected dt-SCS $\mathfrak{S}$.

In the following, we seek to provide a solution to Problem 4 for large-scale dt-SCS $\mathfrak{S}$ as in Definition 2 by utilizing barrier certificate-based approaches. In particular, we want to synthesize control barrier certificates along with suitable controllers in order to ensure the satisfaction of safety specifications with some probabilities. However, doing so in a monolithic fashion such as that presented in Chapter 2 may become computationally intractable. In order to overcome this challenge, we present two different compositional frameworks based on small-gain and dissipativity theories for the construction of control barrier certificates by considering the large-scale dt-SCS $\mathfrak{S}$ as an interconnection of several smaller subsystems as in Definition 12 and computing the so-called control sub-barrier certificates and local controllers for the subsystems.

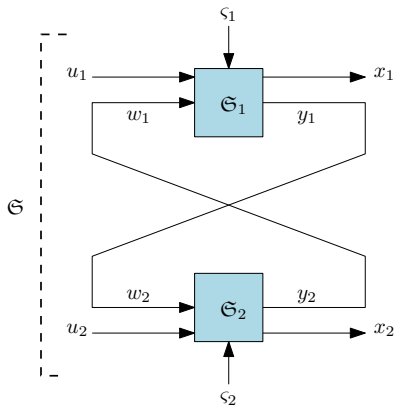


Figure 3.1: Illustration of an interconnected dt-SCS $\mathfrak{S}$ composed of dt-SCS $\mathfrak{S}_1$, $\mathfrak{S}_2$.

Then, by leveraging compositionality conditions (*i.e.* small-gain or dissipativity conditions), the control barrier certificate and corresponding controller for the interconnected dt-SCS $\mathfrak{S}$ can be constructed from the control sub-barrier certificates and local controllers of the subsystems.

3.3 Small-Gain Approach

In this section, we provide our first compositional framework for the construction of control barrier certificates for large-scale dt-SCS $\mathfrak{S}$ as in Definition 2, via *small-gain* type compositionality conditions. To do this, we first present the definition of control sub-barrier certificates for subsystems as in Definition 12, which are useful for constructing CBCs for the interconnected dt-SCS $\mathfrak{S}$. Then, we derive sufficient max-type small gain compositionality conditions that we utilize to construct CBCs of the interconnected system. The obtained CBC is then utilized to establish lower bounds on the probability that the interconnected system $\mathfrak{S}$ avoids unsafe regions over *finite time horizons*, thereby allowing verification and synthesis of safety properties. Finally, we demonstrate the efficacy of our results for safety problems with the help of two case studies: a room temperature network in a circular building consisting of 1000 rooms, as well as a network of Kuramoto oscillators with 100 oscillators.

3.3.1 Control (Sub-)Barrier Certificates

In this subsection, we first define the notion of control (sub)-barrier certificates for both subsystems as well as interconnected discrete-time stochastic control systems, which will be later utilized to obtain probabilistic guarantees on the satisfaction of safety specifications over interconnected systems.

Definition 13. *Consider a subsystem $\mathfrak{S} = (X, U, W, \varsigma, f, Y, h)$, and sets $X_0, X_u \subseteq X$, respectively. A function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is said to be a control sub-barrier certificate (CSBC) for $\mathfrak{S}$ if there exist functions $\alpha, \kappa \in \mathcal{K}_\infty$, with $\kappa < \mathcal{I}_d$, $\rho \in \mathcal{K}_\infty \cup \{0\}$, and constants $\eta, c \in \mathbb{R}_{\geq 0}$ and $\beta \in \mathbb{R}_{>0}$, such*

that

$$\mathbb{B}(x) \geq \alpha(\|h(x)\|^2), \qquad \textit{for all } x \in X, \tag{3.3}$$
$$\mathbb{B}(x) \leq \eta, \qquad \textit{for all } x \in X_0, \tag{3.4}$$
$$\mathbb{B}(x) \geq \beta, \qquad \textit{for all } x \in X_u, \tag{3.5}$$

and $\forall x \in X$, $\exists u \in U$, *such that* $\forall w \in W$,

$$\mathbb{E}\Big[\mathbb{B}(f(x,u,w,\varsigma)) \;\Big|\; x,u,w\Big] \leq \max\Big\{\kappa(\mathbb{B}(x)), \rho(\|w\|^2), c\Big\}. \tag{3.6}$$

We now present a similar definition for control barrier certificates for interconnected systems.

Definition 14. *Consider an interconnected dt-SCS* $\mathfrak{S} = (X, U, \varsigma, f)$. *A function* $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ *is called a control barrier certificate (CBC) for* $\mathfrak{S}$ *with respect to initial and unsafe sets* $X_0, X_u \subseteq X$, *respectively, if*

$$\mathbb{B}(x) \leq \eta, \qquad \textit{for all } x \in X_0, \tag{3.7}$$
$$\mathbb{B}(x) \geq \beta, \qquad \textit{for all } x \in X_u, \tag{3.8}$$

and $\forall x \in X$, $\exists u \in U$, *such that*

$$\mathbb{E}\Big[\mathbb{B}(f(x,u,\varsigma)) \;\Big|\; x,u\Big] \leq \max\Big\{\kappa(\mathbb{B}(x)), c\Big\}, \tag{3.9}$$

for a function $\kappa \in \mathcal{K}_\infty$, *with* $\kappa < \mathcal{I}_d$, *and constants* $\eta, c \in \mathbb{R}_{\geq 0}$ *and* $\beta \in \mathbb{R}_{>0}$, *with* $\beta > \eta$.

Now, by employing Definition 14, we now provide the following theorem that enables us to quantify a lower bound on the probability that the interconnected dt-SCS $\mathfrak{S}$ satisfies the safety specification in a finite time horizon.

Theorem 5. *Let* $\mathfrak{S} = (X, U, \varsigma, f)$ *be an interconnected dt-SCS. Suppose* $\mathbb{B}$ *is a CBC for* $\mathfrak{S}$ *and there exists a constant* $0 < \hat{\kappa} < 1$ *such that function* $\kappa \in \mathcal{K}_\infty$ *in* (3.9) *satisfies* $\kappa(s) \leq \hat{\kappa} s$, $\forall s \in \mathbb{R}_{\geq 0}$. *Then the probability that the solution process of* $\mathfrak{S}$ *starts from any initial state* $x_0 \in X_0$ *and remains safe from the region* X_u, *under a controller* ϖ *obtained via condition* (3.9), *within finite time steps* $t \in [0, T_d)$ *is lower bounded as*

$$\mathbb{P}\Big\{\mathbf{x}_{x_0,\varpi}(t) \notin X_u \textit{ for all } t \in [0,T_d)\Big\} \geq 1 - \varepsilon, \tag{3.10}$$

where,

$$\varepsilon = \begin{cases} 1 - (1 - \frac{\eta}{\beta})(1 - \frac{c}{\beta})^{T_d}, & \textit{if } \beta \geq \frac{c}{1-\hat{\kappa}}, \\ \frac{\eta}{\beta}\hat{\kappa}^{T_d} + \frac{c}{(1-\hat{\kappa})\beta}(1 - \hat{\kappa}^{T_d}), & \textit{if } \beta < \frac{c}{1-\hat{\kappa}}. \end{cases}$$

Proof. According to the condition (3.8), $X_u \subseteq \{x \in X \;|\; \mathbb{B}(x) \geq \beta\}$. Then we have

$$\mathbb{P}\Big\{\mathbf{x}_{x_0,\varpi} \in X_u \text{ for all } t \in [0,T_d) \;\Big|\; a\Big\} \leq \mathbb{P}\Big\{\sup_{0 \leq k < T_d} \mathbb{B}(\mathbf{x}_{x_0,\varpi}(t)) \geq \beta \;\Big|\; x_0\Big\}. \tag{3.11}$$

By applying [70, Theorem 3, Chapter III] to (3.11), and employing respectively conditions (3.9) and (3.7), one obtains the following probability upper bound:

$$\mathbb{P}\Big\{ \sup_{0\leq k<T_d} \mathbb{B}(\mathbf{x}_{x_0,\varpi}(t)) \geq \beta \;\Big|\; x_0 \Big\} \leq \varepsilon.$$

The proposed bounds in (3.10) is then obtained by means of complementation. □

Remark 8. *The condition $\beta > \eta$ is required in Definition 14 for interconnected dt-SCS for the sake of providing meaningful probabilistic guarantees over the satisfaction of safety specifications given by Theorem 5. However, the same condition is not necessarily required in Definition 13 for the subsystems. This is due to the fact that CSBCs in Definition 13 are useless on their own to ensure the safety of the interconnected system and instead are only used to compositionally construct CBCs as in Definition 14.*

Remark 9. *Note that CBC $\mathbb{B}$ satisfying the condition* (3.9) *with $c = 0$ is a non-negative supermartingale similar to condition* (2.23)*. Although the supermartingale property on $\mathbb{B}$ allows one to provide probabilistic guarantees for* infinite time *horizons (see Theorem 2), it is restrictive in the sense that a supermartingale CBC $\mathbb{B}$ may not exist in general [117]. We, therefore, employ a more general c-martingale-type condition in this setting, at the cost of providing probabilistic guarantees for finite time horizons.*

In the next section, we describe interconnected stochastic control systems as a composition of several stochastic subsystems and provide compositional conditions under which a CBC of an interconnected system can be constructed from CSBCs of subsystems.

3.3.2 Compositional Construction of CBC

Suppose we are given N control subsystems

$$\mathfrak{S}_i = (X_i, U_i, W_i, \varsigma_i, f_{(i)}, Y_i, h_{(i)}), \quad i \in \{1, \ldots, N\}, \tag{3.12}$$

where $X_i \in \mathbb{R}^{n_i}$, $U_i \in \mathbb{R}^{m_i}$, $W_i \in \mathbb{R}^{p_i}$, and $Y_i \in \mathbb{R}^{r_i}$, whose internal inputs and outputs are partitioned as

$$\begin{aligned} w_i &= [w_{i1}; \ldots; w_{i(i-1)}; w_{i(i+1)}; \ldots; w_{iN}], \\ y_i &= [y_{i1}; \ldots; y_{iN}], \end{aligned} \tag{3.13}$$

and their output spaces and functions are of the form

$$Y_i = \prod_{j=1}^{N} Y_{ij}, \quad h_{(i)}(x_i) = [h_{(i1)}(x_i); \ldots; h_{(iN)}(x_i)]. \tag{3.14}$$

We call outputs $y_{ii} = x_i$ as *external* ones, whereas outputs y_{ij} with $i \neq j$ are *internal* ones which are used to interconnect stochastic control subsystems. If there exists a connection from $\mathfrak{S}_j$ to $\mathfrak{S}_i$, then $w_{ij} = y_{ji}$. Otherwise, the connecting output is considered identically zero, *i.e.,* $h_{(ji)} \equiv 0$.

Remark 10. *The term "internal" is utilized to refer to those inputs and outputs of subsystems that affect the behavior of other subsystems, i.e., an internal input of a subsystem is affected by an internal output of another one. The term "external" is employed to describe those inputs and outputs that are not used for constructing the interconnection. We assume that one has full-state information in order to synthesize controllers, i.e.,* $h_{(ii)}(x_i) = x_i$*. In the absence of full-state information, the controller synthesis becomes more challenging since one requires the existence of an estimator with some given accuracy. See [65] for a detailed discussion. Under this assumption, we are able to formulate CSBCs and controllers directly over the actual states of the system.*

We now provide the formal definition of our interconnection framework.

Definition 15. *Consider* $N \in \mathbb{N}_{\geq 1}$ *stochastic control subsystems* $\mathfrak{S}_i = (X_i, U_i, W_i, \varsigma_i, f_{(i)}, Y_i, h_{(i)})$*,* $i \in \{1, \ldots, N\}$*, with the input-output partition as in* (3.13) *and* (3.14)*. The interconnected discrete-time stochastic control system* $\mathfrak{S} = (X, U, \varsigma, f)$ *is composed of* $\mathfrak{S}_i$*,* $\forall i \in \{1, \ldots, N\}$*, denoted by* $\mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ *such that* $X := \prod_{i=1}^{N} X_i$*,* $U := \prod_{i=1}^{N} U_i$*,* $\varsigma := [\varsigma_1; \ldots; \varsigma_N]$*, and* $f := \prod_{i=1}^{N} \hat{f}_{(i)}$*, subjected to:*

$$\forall i, j \in \{1, \ldots, N\}, i \neq j: \quad w_{ji} = y_{ij}, \quad Y_{ij} \subseteq W_{ji}.$$

Utilizing the above definition for the interconnected dt-SCS $\mathfrak{S}$ and the subsystems, we now provide a compositional framework for the construction of CBC for the interconnected dt-CS $\mathfrak{S}$ by using CSBCs for the subsystems $\mathfrak{S}_i$. For each subsystem $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$, suppose there exists CSBC $\mathbb{B}_i$ as defined in Definition 13 with respect to sets X_{0_i} and X_{u_i} such that $X_0 = \prod_i^N X_{0_i}$ and $X_u = \prod_{i=1}^{N} X_{u_i}$, respectively, with functions $\alpha_i, \kappa_i \in \mathcal{K}_\infty$, with $\kappa_i < \mathcal{I}_d$, $\rho_i \in \mathcal{K}_\infty \cup \{0\}$, and constants $\eta_i, c_i \in \mathbb{R}_{\geq 0}$ and $\beta_i \in \mathbb{R}_{>0}$. Now we present the following small-gain assumption that is essential for the compositional construction of CBC for $\mathfrak{S}$.

Assumption 2. *Assume that* $\mathcal{K}_\infty$ *functions* κ_{ij} *defined as*

$$\kappa_{ij}(s) := \begin{cases} \kappa_i(s), & \text{if } i = j, \\ \rho_i(\alpha_j^{-1}(s)), & \text{if } i \neq j, \end{cases}$$

satisfy

$$\kappa_{i_1 i_2} \circ \kappa_{i_2 i_3} \circ \cdots \circ \kappa_{i_{r-1} i_r} \circ \kappa_{i_r i_1} < \mathcal{I}_d, \tag{3.15}$$

for all sequences $(i_1, \ldots, i_r) \in \{1, \ldots, N\}^r$ *and* $r \in \{1, \ldots, N\}$*.*

The small-gain condition (2) implies the existence of $\mathcal{K}_\infty$ functions $\varrho_i > 0$ [104, Theorem 5.5], satisfying

$$\max_{i,j} \left\{ \varrho_i^{-1} \circ \kappa_{ij} \circ \varrho_j \right\} < \mathcal{I}_d, \quad i, j = \{1, \ldots, N\}. \tag{3.16}$$

Remark 11. *Note that* (3.15) *is a standard small-gain assumption employed for investigating the stability of large-scale interconnected systems via ISS Lyapunov functions [35, 36]. This condition is automatically satisfied if each* κ_{ij} *is less than identity (*i.e., $\kappa_{ij} < \mathcal{I}_d, \forall i, j \in \{1, \ldots, N\}$*).*

In the following theorem, we show that one can construct a CBC of $\mathfrak{S}$ using CSBC of $\mathfrak{S}_i$ if Assumption 2 holds and $\max_i \varrho_i^{-1}$ is concave (in order to employ Jensen's inequality [26]).

Theorem 6. *Consider the interconnected dt-SCS $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ induced by $N \in \mathbb{N}_{\geq 1}$ stochastic control subsystems $\mathfrak{S}_i$. Suppose that each $\mathfrak{S}_i$ admits a CSBC $\mathbb{B}_i$ as defined in Definition 13 with respect to sets $X_{0_i}, X_{u_i} \subseteq X_i$. If Assumption 2 holds and*

$$\max_i \left\{\varrho_i^{-1}(\beta_i)\right\} > \max_i \left\{\varrho_i^{-1}(\eta_i)\right\}, \tag{3.17}$$

then function $\mathbb{B}(x)$ defined as

$$\mathbb{B}(x) := \max_i \left\{\varrho_i^{-1}(\mathbb{B}_i(x_i))\right\}, \tag{3.18}$$

is a CBC for the interconnected system $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ with respect to initial and unsafe sets $X_0 = \prod_{i=1}^N X_{0_i}$ and $X_u = \prod_{i=1}^N$, respectively, provided that $\max_i \varrho_i^{-1}$ for ϱ_i as in (3.16) is concave.

Proof. We first show that conditions (3.7) and (3.8) in Definition 14 hold. For any $x = [x_1; \ldots; x_N] \in X_0 = \prod_{i=1}^N X_{0_i}$ from (3.4), we have

$$\mathbb{B}(x) = \max_i \left\{\varrho_i^{-1}(\mathbb{B}_i(x_i))\right\} \leq \max_i \left\{\varrho_i^{-1}(\eta_i)\right\} = \eta,$$

and similarly for any $x = [x_1; \ldots; x_N] \in X_u = \prod_{i=1}^N X_{u_i}$ and from (3.5), one has

$$\mathbb{B}(x) = \max_i \left\{\varrho_i^{-1}(\mathbb{B}_i(x_i))\right\} \geq \max_i \left\{\varrho_i^{-1}(\beta_i)\right\} = \beta,$$

satisfying conditions (3.7) and (3.8) with $\eta = \max_i \left\{\varrho_i^{-1}(\eta_i)\right\}$ and $\beta = \max_i \left\{\varrho_i^{-1}(\beta_i)\right\}$.

Now we show that the condition (3.9) holds as well. Let $\kappa(s) = \max_{i,j}\{\varrho_i^{-1} \circ \kappa_{ij} \circ \varrho_j(s)\}$. It follows from (3.15) that $\kappa < \mathcal{I}_d$. Moreover, $\beta > \eta$ according to (3.17). Since $\max_i \varrho_i^{-1}$ is concave, one can readily acquire the chain of inequalities in (3.20) using Jensen's inequality, and by defining the constant c as

$$c := \max_i \varrho_i^{-1}(c_i).$$

Hence $\mathbb{B}(x)$ is a CBC for the interconnected system $\mathfrak{S}$ which completes the proof. □

So far, we have discussed the construction of CBC for the interconnected dt-SCS $\mathfrak{S}$ by utilizing CSBCs for subsystems. In the next section, we provide two systematic approaches to search for CSBCs and their corresponding local controllers.

3.3.3 Computation of CSBC and Corresponding Controllers

In this subsection, we provide suitable methods to search for CSBC and synthesize corresponding local controllers satisfying simple safety specifications for subsystems $\mathfrak{S}_i$. We propose two different approaches: one is based on the sum-of-squares (SOS) optimization problem and another one relies on counter-example guided inductive synthesis (CEGIS) framework.

$$
\begin{aligned}
\mathbb{E}\Big[\mathbb{B}(f(x,u,\varsigma))\,\big|\,x,u\Big] &= \mathbb{E}\Big[\max_i\Big\{\varrho_i^{-1}(\mathbb{B}_i(f_{(i)}(x_i,u_i,w_i,\varsigma_i)))\Big\}\,\big|\,x,u,w\Big]\\
&\leq \max_i\Big\{\varrho_i^{-1}(\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i,u_i,w_i,\varsigma_i))\,\big|\,x,u,w\Big])\Big\}\\
&= \max_i\Big\{\varrho_i^{-1}(\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i,u_i,w_i,\varsigma_i))\,\big|\,x_i,u_i,w_i\Big])\Big\}\\
&\leq \max_i\Big\{\varrho_i^{-1}(\max\{\kappa_i(\mathbb{B}_i(x_i)),\rho_i(\|w_i\|^2),c_i\})\Big\}\\
&= \max_i\Big\{\varrho_i^{-1}(\max\{\kappa_i(\mathbb{B}_i(x_i)),\rho_i(\max_{j,j\neq i}\{\|w_{ij}\|^2\}),c_i\})\Big\}\\
&= \max_i\Big\{\varrho_i^{-1}(\max\{\kappa_i(\mathbb{B}_i(x_i)),\rho_i(\max_{j,j\neq i}\{\|y_{ji}\|^2\}),c_i\})\Big\}\\
&= \max_i\Big\{\varrho_i^{-1}(\max\{\kappa_i(\mathbb{B}_i(x_i)),\rho_i(\max_{j,j\neq i}\{\|h_{(j)}(x_j)\|^2\}),c_i\})\Big\}\\
&\leq \max_i\Big\{\varrho_i^{-1}(\max\{\kappa_i(\mathbb{B}_i(x_i)),\rho_i(\max_{j,j\neq i}\{\alpha_j^{-1}(\mathbb{B}_j(x_j))\}),c_i\})\Big\}\\
&= \max_{i,j}\Big\{\varrho_i^{-1}(\max\{\kappa_{ij}(\mathbb{B}_j(x_j)),c_i\})\Big\}\\
&= \max_{i,j}\Big\{\varrho_i^{-1}(\max\{\kappa_{ij}\circ\varrho_j\circ\varrho_j^{-1}(\mathbb{B}_j(x_j)),c_i\})\Big\} \qquad (3.19)\\
&\leq \max_{i,j,l}\Big\{\varrho_i^{-1}(\max\{\kappa_{ij}\circ\varrho_j\circ\varrho_l^{-1}(\mathbb{B}_l(x_l)),c_i\})\Big\}\\
&= \max_{i,j}\Big\{\varrho_i^{-1}(\max\{\kappa_{ij}\circ\varrho_j\circ\mathbb{B}(x),c_i\})\Big\} = \max\Big\{\kappa(\mathbb{B}(x)),c\Big\}. \qquad (3.20)
\end{aligned}
$$

Sum-of-Squares Optimization Problem

Here, we propose to reformulate conditions (3.3)-(3.6) as a sum-of-squares (SOS) optimization problem [93] by restricting CSBC to be a non-negative polynomial that can be represented as a sum of squares of different polynomials. However, to do so, we need to raise the following assumption.

Assumption 3. *The stochastic control subsystem $\mathfrak{S}_i$ has a continuous state set $X_i \subseteq \mathbb{R}^{n_i}$, and continuous external and internal input sets $U_i \subseteq \mathbb{R}^{m_i}$ and $W_i \subseteq \mathbb{R}^{p_i}$. Its vector field $f_{(i)} : X_i \times U_i \times W_i \times \mathcal{V}_{\varsigma_i} \to X_i$ is a polynomial function of the state x_i, the external input u_i, and the internal input w_i. We also assume that the output map $h_{(i)} : X_i \to Y_i$ and $\mathcal{K}_\infty$ functions α_i and ρ_i are polynomial.*

Under Assumption 3, one can reformulate conditions (3.3)-(3.6) as an SOS optimization problem to search for a polynomial CSBC $\mathbb{B}_i$ and a polynomial controller $\varpi(\cdot)$ for the subsystem $\mathfrak{S}_i$. Then, one can utilize the computed CSBCs and the compositionality results presented in the last subsection in order to obtain the CBC for the interconnected system $\mathfrak{S}$.

Lemma 1. *Suppose Assumption 3 holds and sets X_i , X_{0_i}, X_{u_i} can be defined by vectors of polynomial inequalities $X_i = \{x_i \in \mathbb{R}^{n_i} \mid g_i(x_i) \geq 0\}$, $X_{0_i} = \{x_i \in \mathbb{R}^{n_i} \mid g_{0_i}(x_i) \geq 0\}$, and $X_{u_i} = \{x_i \in \mathbb{R}^{n_i} \mid g_{u_i}(x_i) \geq 0\}$, where the inequalities are provided element-wise. Similarly, let the internal input set W_i be defined by vectors of a polynomial inequality $W_i = \{w_i \in \mathbb{R}^{p_i} \mid g_{w_i}(w_i) \geq 0\}$. Suppose for a given control subsystem $\mathfrak{S}_i$, there exists a sum-of-squares polynomial $\mathbb{B}_i(x_i)$, constants $\eta_i, \bar{c}_i \in \mathbb{R}_{\geq 0}$, $\beta_i \in \mathbb{R}_{>0}$, functions $\bar{\rho}_i \in \mathcal{K}_\infty \cup \{0\}$, $\alpha_i, \bar{\kappa}_i \in \mathcal{K}_\infty$, with $\bar{\kappa}_i < \mathcal{I}_d$, vectors of sum-of-squares polynomials $\lambda_{0_i}(x_i), \lambda_{u_i}(x_i)$, $\lambda_i(x_i)$, $\hat{\lambda}_i(x_i)$, $\lambda_{w_i}(w_i)$ and polynomials $\lambda_{u_{ji}}(x_i)$ corresponding to the j^{th} input in $u_i = (u_{1i}, \ldots, u_{m_{ii}}) \in U_i \subseteq \mathbb{R}^{m_i}$ of appropriate dimensions such that the following expressions are sum-of-squares polynomials:*

$$\mathbb{B}_i(x_i) - \lambda_i^T(x_i) g_i(x_i) - \alpha_i(h_{(i)}^T(x_i) h_{(i)}(x_i)), \tag{3.21}$$

$$-\mathbb{B}_i(x_i) - \lambda_{0_i}^T(x_i) g_{0_i}(x_i) + \eta_i, \tag{3.22}$$

$$\mathbb{B}_i(x_i) - \lambda_{u_i}^T(x_i) g_{u_i}(x_i) - \beta_i, \tag{3.23}$$

$$\begin{aligned} -\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \mid x_i, u_i, w_i\Big] + \bar{\kappa}_i(\mathbb{B}_i(x_i)) + \bar{\rho}_i(\frac{w_i^T w_i}{p_i}) + \bar{c}_i \\ - \sum_{j=1}^{m_i} (u_{ji} - \lambda_{u_{ji}}(x_i)) - \hat{\lambda}_i^T(x_i) g_i(x_i) - \lambda_{w_i}^T(w_i) g_{w_i}(w_i), \end{aligned} \tag{3.24}$$

where p_i is the dimension of the internal input w_i. Then $\mathbb{B}_i(x_i)$ is a CSBC satisfying conditions (3.3)-(3.6) *and $\varpi_i(x_i) = [\lambda_{u_{1i}}(x_i); \ldots; \lambda_{u_{m_{ii}}}(x_i)]$, $i \in \{1, \ldots, N\}$, is the corresponding controller for the subsystem $\mathfrak{S}_i$. The parameters satisfying the conditions are given by*

$$\begin{aligned} \kappa_i &= \mathcal{I}_d - (\mathcal{I}_d - \pi_i) \circ (\mathcal{I}_d - \bar{\kappa}_i), \\ \rho_i &= (\mathcal{I}_d + \bar{\delta}_i) \circ (\mathcal{I}_d - \bar{\kappa}_i)^{-1} \circ \pi_i^{-1} \circ \bar{\pi}_i \circ \bar{\rho}_i, \\ c_i &= (\mathcal{I}_d + \bar{\delta}_i^{-1}) \circ (\mathcal{I}_d - \bar{\kappa}_i)^{-1} \circ \pi_i^{-1} \circ \bar{\pi}_i \circ (\bar{\pi}_i - \mathcal{I}_d)^{-1}(\bar{c}_i), \end{aligned}$$

where $\bar{\delta}_i, \pi_i, \bar{\pi}_i$ are some arbitrarily chosen $\mathcal{K}_\infty$ functions so that $\mathcal{I}_d - \pi_i \in \mathcal{K}_\infty$ and $\bar{\pi}_i - \mathcal{I}_d \in \mathcal{K}_\infty$.

Proof. Since $\mathbb{B}_i(x_i)$ and $\lambda_i(x_i)$ in (3.21) are sum-of-squares, we have $0 \leq \mathbb{B}_i(x_i) - \lambda_i^T(x_i) g_i(x_i) - \alpha_i(h_{(i)}^T(x_i) h_{(i)}(x_i))$. Since $\|h_{(i)}(x_i)\|^2 \leq h_{(i)}^T(x_i) h_{(i)}(x_i)$, we have $0 \leq \mathbb{B}_i(x_i) - \lambda_i^T(x_i) g_i(x_i) - \alpha_i(\|h_{(i)}(x_i)\|^2)$. Since the term $\lambda_i^T(x_i) g_i(x_i)$ is non-negative over X, the condition (3.21) implies the condition (3.3). Similarly, we can show that (3.22) and (3.23) imply conditions (3.4) and (3.5), respectively. Now we proceed with showing that the condition (3.24) implies (3.6), as well. By selecting external inputs $u_{ji} = \lambda_{u_{ji}}(x_i)$ and since the terms $\hat{\lambda}_i^T(x_i) g_i(x_i), \lambda_{w_i}^T(w_i) g_{w_i}(w_i)$ are non-negative over the set X, we have $\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \Big| x_i, u_i, w_i\Big] \leq \bar{\kappa}_i(\mathbb{B}_i(x_i)) + \bar{\rho}_i(\frac{w_i^T w_i}{p_i}) + \bar{c}_i$ implying that $\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \Big| x_i, u_i, w_i\Big] \leq \bar{\kappa}_i(\mathbb{B}_i(x_i)) + \bar{\rho}_i(\|w_i\|^2) + \bar{c}_i$, since $w_i^T w_i \leq p_i \|w_i\|^2$. By employing a similar argument as the one in [121, Theorem 1], the additive form of the right-hand side of the above inequality can be converted into a max form as

$$\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \mid x_i, u_i, w_i\Big] \leq \max\Big\{\kappa_i(\mathbb{B}_i(x_i)), \rho_i(\|w_i\|^2), c_i\Big\},$$

where,

$$\kappa_i = \mathcal{I}_d - (\mathcal{I}_d - \pi_i) \circ (\mathcal{I}_d - \bar{\kappa}_i),$$
$$\rho_i = (\mathcal{I}_d + \bar{\delta}_i) \circ (\mathcal{I}_d - \bar{\kappa}_i)^{-1} \circ \pi_i^{-1} \circ \bar{\pi}_i \circ \bar{\rho}_i,$$
$$c_i = (\mathcal{I}_d + \bar{\delta}_i^{-1}) \circ (\mathcal{I}_d - \bar{\kappa}_i)^{-1} \circ \pi_i^{-1} \circ \bar{\pi}_i \circ (\bar{\pi}_i - \mathcal{I}_d)^{-1}(\bar{c}_i),$$

with $\bar{\delta}_i, \pi_i, \bar{\pi}_i$ being some arbitrarily chosen $\mathcal{K}_\infty$ functions so that $\mathcal{I}_d - \pi_i \in \mathcal{K}_\infty$, $\bar{\pi}_i - \mathcal{I}_d \in \mathcal{K}_\infty$. Hence this implies that the function $\mathbb{B}_i(x_i)$ is a CSBC and the proof is completed. □

Counter-Example Guided Inductive Synthesis

This approach involves finding a CSBC of a given parametric form,*e.g.,* polynomials, by utilizing satisfiability modulo theories (SMT) solvers such as Z3 [39], dReal [51] or MathSat [27]. Unlike SOS optimization, this framework does not require any restrictions on the underlying dynamics and is applicable under the following assumption on the underlying dynamics of the system. However, the following assumption is essential.

Assumption 4. *Each subsystem $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$, has a compact state set X_i, a compact internal input set W_i and a compact external input set U_i.*

Under Assumption 4, we propose the following lemma to reformulate conditions (3.3)-(3.6) as a satisfiability problem.

Lemma 2. *Consider a stochastic control subsystem $\mathfrak{S}_i = (X_i, U_i, W_i, \varsigma_i, f_{(i)}, Y_i, h_{(i)}), i \in \{1, \ldots, N\}$, satisfying Assumption 4. Suppose there exist a function $\mathbb{B}_i(x_i)$, constants $\eta_i, c_i \in \mathbb{R}_{\geq 0}, \beta_i \in \mathbb{R}_{>0}$, functions $\rho_i \in \mathcal{K}_\infty \cup \{0\}, \alpha_i, \kappa_i \in \mathcal{K}_\infty$, with $\kappa_i < \mathcal{I}_d$ such that*

$$\bigwedge_{x_i \in X_i} \big(\mathbb{B}_i(x_i) \geq \alpha_i(\|h_{(i)}(x_i)\|^2)\big) \bigwedge_{x_i \in X_{0_i}} \big(\mathbb{B}_i(x_i) \leq \eta_i\big) \bigwedge_{x_i \in X_{u_i}} \big(\mathbb{B}_i(x_i) \geq \beta_i\big) \bigwedge_{x_i \in X_i} \bigvee_{\nu_i \in U_i} \bigwedge_{w_i \in W_i} \Big(\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \,\Big|\, x_i, u_i, w_i\Big] \leq \max\{\kappa_i(\mathbb{B}_i(x_i)), \rho_i(\|w_i\|^2), c_i\}\Big).$$

Then $\mathbb{B}_i(x_i)$ is a CSBC satisfying conditions (3.3)-(3.6).

3.3.4 Case Studies

In this section, we demonstrate our proposed results using two physical case studies. The first case study presents a room temperature regulation problem in a circular building with 1000 rooms, where the temperatures of the room are required to be maintained within a certain range. The second one is a fully-interconnected Kuramoto network with 100 nonlinear oscilattors where we synthesize controllers to ensure that the angular positions of the oscillators do not cross some specified bounds.

Room Temperature Network

We first apply our approaches to a room temperature network in a circular building consisting of $N = 1000$ rooms. The model of this case study is borrowed from [85] by including stochasticity as additive noise. The evolution of the temperature $T(\cdot)$ in the interconnected system is governed by the following dynamics

$$\mathfrak{S} : T(t+1) = AT(t) + \mu T_H \nu(t) + \iota T_E + 0.1\varsigma(t),$$

where $A \in \mathbb{R}^{N\times N}$ is a matrix with diagonal elements given by $\bar{a}_{ii} = (1 - 2\epsilon - \iota - \mu\nu_i(t))$, off-diagonal elements $\bar{a}_{i,i+1} = \bar{a}_{i+1,i} = \bar{a}_{1,n} = \bar{a}_{n,1} = \epsilon$, $i \in \{1, \ldots, n-1\}$, and all other elements are identically zero. Parameters $\epsilon = 0.005$, $\iota = 0.06$, and $\mu = 0.145$ are conduction factors between rooms $i \pm 1$ and i, the external environment and the room i, and the heater and the room i, respectively. Outside temperatures are the same for all rooms: $T_{ei} = -15\,°\mathsf{C}$, $\forall i \in \{1, \ldots, n\}$, and the heater temperature is $T_H = 45\,°\mathsf{C}$. Moreover, $T(t) = [T_1(t); \ldots; T_n(t)]$, $\varsigma = [\varsigma_1(t); \ldots; \varsigma_n(t)]$, $\nu(t) = [\nu_1(t); \ldots; \nu_n(t)]$, and $T_E = [T_{e_1}; \ldots; T_{e_n}]$.

We consider the state set $X = [0, 50]^N$, $X_0 = [19.5, 20]^N$, and $X_u = [1, 17]^N \cup [23, 50]^N$. The main goal is to synthesize controllers such that the solution processes of the system $\mathfrak{S}$ do not violate the safety conditions by not reaching the unsafe regions in X_u for a time horizon of $T_d = 10$. This means that temperature of all rooms are safely maintained between 17°C and 23°C for all time $t \in [0, T_d)$, with an initial temperature between 19.5°C and 20°C. To do this, we consider $\mathfrak{S}$ as a network consisting of $N = 1000$ subsystems (individual rooms) $\mathfrak{S}_i$ represented by

$$\mathfrak{S}_i : \begin{cases} T_i(t+1) = \bar{a}T_i(t) + \mu T_H \nu_i(t) + \epsilon w_i(t) + \iota T_{ei} + 0.1\varsigma_i(t), \\ y_i(t) = T_i(t). \end{cases}$$

One can readily verify that $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_n)$ where $w_i(t) = [T_{i-1}(t); T_{i+1}(t)]$ (with $T_0 = T_n$ and $T_{n+1} = T_1$). We utilize the software tool SOSTOOLS [98] and the SDP solver SeDuMi [119] to compute a CSBC as described in Section 3.3.2. Based on Lemma 1, we compute a CSBC of an order 2 as $\mathbb{B}_i(T_i) = 0.7659T_i^2 - 30.24T_i + 298.5$ and the corresponding controller of an order 1 as $\varpi_i(T_i) = -0.012T_i + 0.8$, $\forall i \in \{1, \ldots, n\}$. Furthermore, the corresponding constants and functions in Definition 13 satisfying conditions (3.3)-(3.6) are computed as $\eta_i = 0.13, \beta_i = 4.4, c_i = 0.0139, \alpha_i(s) = 5 \times 10^{-5}s, \kappa_i(s) = 0.99s$, and $\rho_i(s) = 4.99 \times 10^{-5}s, \forall s \in \mathbb{R}_{\geq 0}$. Then, In order to construct a CBC for the interconnected system using CSBC of subsystems, we now check the small-gain condition (3.15) that is required for the compositionality result. By taking $\varrho_i(s) = s, \forall i \in \{1, \ldots, n\}$, the condition (3.15) and as a result the condition (3.16) is always satisfied without any restriction on the number of rooms. Moreover, the compositionality condition (3.17) is also met since $\beta_i > \eta_i, \forall i \in \{1, \ldots, n\}$. Then one can conclude that $\mathbb{B}(T) = \max_i \left\{0.7659T_i^2 - 30.24T_i + 298.5\right\}$ is a CBC for the interconnected system $\mathfrak{S}$. Accordingly, $\varpi(T) = [-0.012T_1 + 0.8; \ldots; -0.012T_{1000} + 0.8]$ is the overall controller for the interconnected system and corresponding parameters satisfying conditions (3.7)-(3.9) are obtained as $\eta = 0.13, \beta = 4.4, c = 0.0139$ and $\kappa(s) = 0.99s, \forall s \in \mathbb{R}_{\geq 0}$.

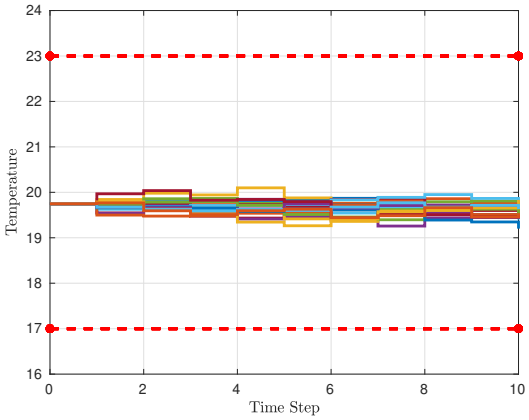


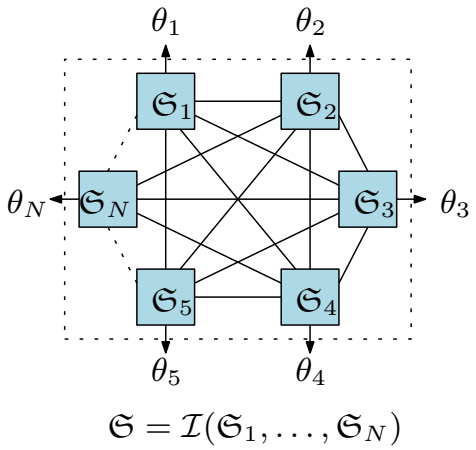
Figure 3.2: Closed-loop stage trajectories of a representative room with 10 noise realizations in a network of 1000 rooms.

We can then guarantee via Theorem 5 that the probability of satisfaction of the safety specification in the time horizon $[0, T_d]$ is at least 0.95. State trajectories of the closed-loop system for a representative room in a network of 1000 rooms with 10 noise realizations are illustrated in Fig. 3.2. Note that the lower bound on the probability of satisfaction proposed by our approach is rather conservative compared to empirical results that can be obtained by running Monte Carlo simulations for the closed-loop system with our computed controller. The reason is due to the conservative nature of barrier certificates which are chosen to be polynomials of a fixed degree but at the gain of providing a formal lower bound on the probability of satisfaction rather than just an empirical one. We should mention that the computation of CSBC and its corresponding controller for each subsystem takes almost 10 seconds with a memory usage of 3.7 MB on a machine with Microsoft Windows (Intel i7-8665U CPU with a 32 GB of RAM).

Network of Kuramoto Controllers

As our second case study, we apply our results to a network of $N = 100$ controlled Kuramoto oscillators in a fully-interconnected topology as illustrated in Figure 3.3 which can model a large number of problems in different fields, such as biology [34], smart grids [54], neural networks [92] and nanotechnology [126]. Our model is adapted from [113] by adding stochasticity as an additive noise and the dynamics of such a model is also presented in Figure 3.3. Here, $\theta = [\theta_1; \ldots; \theta_N]$ is the phase of oscillators with $\theta_i \in [0, 2\pi]$, $\forall i \in \{1, \ldots, N\}$, $\Omega = [\Omega_1; \ldots; \Omega_N] = [0.01; \ldots; 0.01]$ is the natural frequency of oscillators, $K = 0.0012$ is the coupling strength, $\tau = 0.1$ is the sampling time, $\phi(\theta(t)) = [\phi(\theta_1(t)); \ldots; \phi(\theta_N(t))]$ such that $\phi(\theta_i(t)) = \sum_{\substack{j=1 \\ i \neq j}} sin(\theta_j(t) - \theta_i(t)), \forall i \in \{1, \ldots, N\}$, $\nu(t) = [\nu_1(t); \ldots; \nu_N(t)]$, and $\varsigma(t) = [\varsigma_1(t); \ldots; \varsigma_N(t)]$.

We want to synthesize controllers such that the system $\mathfrak{S}$ satisfies the safety specification with respect to initial set $X_0 = [\frac{4\pi}{9}, \frac{5\pi}{9}]^N$ and the unsafe set $X_u = [0, \frac{\pi}{15}]^N \cup [\frac{14\pi}{15}, \pi]^N$ within time horizon $T_d = 7$. To do so, we consider the network of N nonlinear oscillators as an interconnection of N subsystems, *i.e.*, $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ where each subsystem $\mathfrak{S}_i, i \in \{1, \ldots, N\}$, can be



$$\mathfrak{S}: \theta(t+1) = \theta(t) + \tau\Omega + \tfrac{\tau K}{N}\phi(\theta(t)) + \nu(t) + 0.05\varsigma(t)$$

$$\mathfrak{S}_i: \begin{cases} \theta_i(t+1) = \theta_i(t) + \tau\Omega_i \\ \qquad + \tfrac{\tau K}{N}\phi(\theta_i(t)) \\ \qquad + \nu_i(t) + 0.05\varsigma_i(t) \\ y_i(k) = \theta_i(t) \end{cases}$$

Figure 3.3: Fully-connected Kuramoto oscillator network $\mathfrak{S}$, and dynamics corresponding to $\mathfrak{S}$ and each subsystem $\mathfrak{S}_i$.

described by dynamics as shown in Figure 3.3 (©2022 IEEE). To compute control sub-barrier certificates and the corresponding local controllers, we utilize the SOS algorithm in Section 3.3.3 and in particular, we use SOSTOOLS and SDP solver SeDuMi. Since dynamics of $\mathfrak{S}$ are not polynomial and SOS algorithm is only equipped to provide solutions for polynomial dynamics, we make an approximation to our dynamics. More precisely, in the condition (3.24), we take an upper bound on the term $\mathbb{B}_i(f_{(i)}(\theta_i, u_i, w_i, \varsigma_i))$ by replacing $sin(\cdot)$ by either 1 or -1 accordingly.

The CSBC and local controller satisfying conditions (3.3)-(3.6) for subsystem $\mathfrak{S}_i$ are given by $\mathbb{B}_i(\theta_i) = 0.001361\theta_i^8 - 0.0001877\theta_i^7 + 0.0004904\theta_i^6 - 0.03395\theta_i^5 + 0.00107\theta_i^4 - 0.1927\theta_i^3 + 1.71\theta_i^2 - 3.205\theta_i + 1.827$, $\varpi_i(\theta_i) = -0.532\theta_i^2 + 1.69$. The other parameters are obtained as $\eta_i = 0.02$, $\beta_i = 1.2$, $c_i = 0.0083$, $\alpha_i(s) = 4.7\times10^{-7}s$, $\kappa_i(s) = 0.997s$, and $\rho_i(s) = 4.49\times10^{-7}s$. By taking $\varrho_i(s) = s$, $\forall i \in \{1, \ldots, n\}$, the condition (3.15) and as a result the condition (3.16) is always satisfied without any restriction on the number of rooms. Moreover, the compositionality condition (3.17) is also met since $\beta_i > \eta_i, \forall i \in \{1, \ldots, n\}$. Then, by utilizing Theorem 6, we compute the CBC as $\mathbb{B}(\theta) = \max_i^N \mathbb{B}_i(x)$ and controller as $\varpi(\theta) = [-0.532\theta_1^2 + 1.69; \ldots; -0.532\theta_{100}^2 + 1.69\theta_{100}]$ for the interconnected system, and also parameters satisfying (3.7)-(3.9) as $\eta = 0.02, \beta = 1.2, c = 0.0083$ and $\kappa(s) = 0.997s$. By utilizing Theorem 5, we obtain the probability of satisfaction for the safety specification to be at least 0.94.

Figure 3.4 shows the evolution of solution processes within the time horizon $T_d = 7$ when starting from the initial region of X_0. It can be seen that the solution processes remain within the safe states and do not cross the unsafe regions. The CSBC computation for this example takes around 1 minute with a memory usage of 30 MB on a Microsoft Windows machine (Intel i7-8665U CPU with 32 GB of RAM).

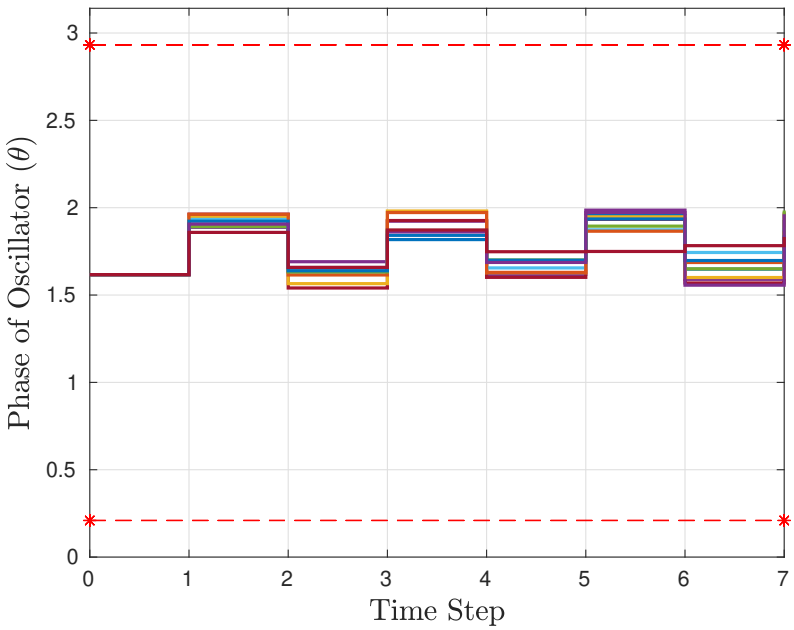


Figure 3.4: Closed-loop state trajectories of a representative oscillator in a network of 100 oscillators with 10 noise realizations with an initial state starting from X_0.

3.4 Dissipativity-based Approach

In this section, we propose a compositional framework based on dissipativity approaches for the construction of control barrier certificates for interconnected discrete-time stochastic systems. The proposed approach utilizes the interconnection topology together with dissipativity-type properties of subsystems. Depending on the structure of the interconnection (*e.g.*, skew-symmetric), one is able to satisfy this compositionality condition without any restriction on the number of subsystems or their gains. Similar to Section 3.3, we first present the definition of control sub-barrier certificates for subsystems of Definition 12. Then by utilizing compositionality conditions derived using dissipativity theory, we construct the control barrier certificates for interconnected systems. However, unlike Section 3.3, in this section, we consider the probabilistic satisfaction of safety specifications over *infinite time horizons*. In other words, we synthesize a suitable controller for an interconnected system and correspondingly compute a lower bound on the probability that the solution processes do not reach unsafe regions.

From Section 3.3, it can be observed that the compositionality condition is usually checked independently after the computation of suitable control barrier certificates. This means that the required parameters for finding suitable control barrier certificates are pre-selected and the compositionality condition is checked a posteriori. While this method can provide tractable results in certain scenarios for systems with specific interconnection structures, it is not particularly useful in large-scale networks where structural properties of the subsystems (*e.g.*, dissipativity properties) are not apparent. Besides, since control sub-barrier certificates are not optimized

with respect to the compositionality condition, obtained results can be conservative. In order to provide scalable, less-conservative results, we employ a distributed optimization method based on an alternating direction method of multipliers (ADMM) algorithm which allows us to break down a large optimization problem into several smaller sub-problems which can be easier to handle. The solution to the optimization problem provides us with suitable control sub-barrier certificates along with local controllers, allowing the computation of control barrier certificates for the interconnected system. For systems with polynomial dynamics, we show that ADMM algorithm can be utilized in conjunction with sum-of-squares (SOS) optimization in order to obtain control sub-barrier certificates and corresponding local controllers. We demonstrate the effectiveness of our proposed results by applying them to a room temperature network in a circular building containing 300 rooms.

3.4.1 Control (Sub-)Barrier Certificates

In this subsection, we introduce control (sub)-barrier certificates that utilize dissipativity-type properties of the subsystems to provide safety guarantees for the interconnected dt-SCS $\mathfrak{S}$ over infinite time horizons. We now present the definition of control sub-barrier certificates for subsystems. Note that this definition is slightly different from that presented in Section 3.3, since it utilizes dissipativity-type properties of subsystems in order to establish the dissipativty-based compositional framework that will be presented later.

Definition 16. *Consider a subsystem $\mathfrak{S} = (X, U, W, \varsigma, f, Y, h)$, and sets $X_0, X_u \subseteq X$, respectively. A function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is said to be a control sub-barrier certificate (CSBC) for $\mathfrak{S}$ if there exists a constant $\eta \in \mathbb{R}_{\geq 0}$ and a symmetric matrix $\underline{X}$ with conformal block partitions $\underline{X}^{ij}, i, j \in \{1, 2\}$ such that*

$$\mathbb{B}(x) \leq \eta, \qquad \text{for all } x \in X_0, \tag{3.25}$$

$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in X_u, \tag{3.26}$$

and $\forall x \in X$, $\exists u \in U$, such that $\forall w \in W$,

$$\mathbb{E}\Big[\mathbb{B}(f(x,u,w,\varsigma)) \;\Big|\; x,u,w\Big] - \mathbb{B}(x) \leq \begin{bmatrix} w \\ h(x) \end{bmatrix}^T \begin{bmatrix} \underline{X}^{11} & \underline{X}^{12} \\ \underline{X}^{21} & \underline{X}^{22} \end{bmatrix} \begin{bmatrix} w \\ h(x) \end{bmatrix}.$$

The definition of control barrier certificates for interconnected dt-SCS $\mathfrak{S}$ is similar to the one presented in Definition 8. However, we present it again, with some minor modifications (see condition 3.28), for the sake of completeness of this section.

Definition 17. *Consider an interconnected dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$. A function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is called a control barrier certificate (CBC) for $\mathfrak{S}$ with respect to initial and unsafe sets $X_0, X_u \subseteq X$, respectively, if there exists constants $\eta \in \mathbb{R}_{\geq 0}$ and $\beta \in \mathbb{R}_{>0}$ with $\beta > \eta$ such that*

$$\mathbb{B}(x) \leq \eta, \qquad \text{for all } x \in X_0, \tag{3.27}$$

$$\mathbb{B}(x) \geq \beta, \qquad \text{for all } x \in X_u, \tag{3.28}$$

and $\forall x \in X$, $\exists u \in U$, *such that*

$$\mathbb{E}\Big[\mathbb{B}(f(x,u,\varsigma)) \;\Big|\; x,u\Big] - \mathbb{B}(x) \leq 0. \tag{3.29}$$

We now utilize Definition 17 to quantify a lower bound on the probability that the interconnected dt-SCS $\mathfrak{S}$ reaches unsafe regions for infinite time horizons. Once again, note that this is similar to the bounds obtained in Theorem 2 but adapted to Definition 17.

Corollary 2. *For an interconnected dt-SCS* $\mathfrak{S} = (X, U, \varsigma, f)$, *let* $\mathbb{B}$ *be a CBC satisfying conditions* (3.27)-(3.29). *Then the lower bound on the probability that the solution processes of* $\mathfrak{S}$ *start from any initial state* $a \in X_0$ *and avoids entering an unsafe region* X_u, *under a controller* ϖ *obtained via condition* (3.29), *is given by*

$$\mathbb{P}\Big\{\mathbf{x}_{x_0,\varpi}(t) \notin X_u \text{ for all } \; 0 \leq t < \infty \;\Big|\; a\Big\} \geq 1 - \varepsilon, \tag{3.30}$$

where $\varepsilon = \frac{\eta}{\beta}$.

Remark 12. *Note that Remark 8 also holds in this case. In order to provide probabilistic guarantees in infinite-time horizons, CBC* $\mathbb{B}$ *in* (3.29) *is required to be a non-negative supermartingale, i.e., the value of CBC is expected to decay at every time step. This can be quite restrictive and there may not exist a CBC satisfying the supermartingale condition* (3.29). *For example, in stochastic control systems with additive noise as seen in 3.3.4, it is not possible to obtain supermartingale control barrier certificates. In such a case, it is possible to relax condition* (3.29) *by introducing a constant* $c > 0$ *in the right-hand side of* (3.29). *In this case, the CBC* $\mathbb{B}$ *is called c-martingale [117] and such a condition ensures that CBC is decaying with an offset of up to c. However, this comes at the cost of providing only finite-time horizon guarantees, as demonstrated in Section 3.3.*

Unfortunately, finding a CBC for large-scale interconnected systems can be difficult due to the computational complexity associated with the dimension of the state set. In this section, we consider a large-scale system as an interconnection of several smaller subsystems and develop a compositional scheme based on dissipativity approaches to construct the CBC of the interconnected system based on CSBCs of individual subsystems. This is explained in detail in the following section.

3.4.2 Compositional Construction of CBC

To obtain a compositional framework using dissipativity-based conditions, we consider the following definition of interconnected stochastic control systems.

Definition 18. *Suppose we are given* $N \in \mathbb{N}_{\geq 1}$ *control subsystems* $\mathfrak{S}_i = (X_i, U_i, W_i, \varsigma_i, f_{(i)}, Y_i, h_{(i)})$, $i \in \{1, \ldots, N\}$, *where* $X_i \subseteq \mathbb{R}^{n_i}$, $U_i \subseteq \mathbb{R}^{m_i}$, $W_i \subseteq \mathbb{R}^{p_i}$, $Y_i \subseteq \mathbb{R}^{r_i}$ *along with a matrix* M *that describes the coupling between the subsystems, with the constraint* $M \prod_{i=1}^{N} Y_i \subseteq M \prod_{i=1}^{N} W_i$ *to provide a well-posed interconnection. Then the interconnection of subsystems* $\mathfrak{S}_i, i \in \{1, \ldots, N\}$,

denoted by $\mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$*, is the dt-SCS* $\mathfrak{S} = (X, U, \varsigma, f)$ *such that* $X := \prod_{i=1}^{N} X_i$*,* $U := \prod_{i=1}^{N} U_i$*, and* $f := \prod_{i=1}^{N} f_{(i)}$*, with internal inputs constrained according to*

$$[w_1; \ldots; w_N] = M[h_{(1)}; \ldots; h_{(N)}]. \tag{3.31}$$

Remark 13. *The structure of the interconnected matrix M depends on the type of interconnection between subsystems. For instance, consider a circular interconnection network of N subsystems (see the case study). For such an interconnected system, M consists of elements* $m_{i,i+1} = m_{i+1,i} = m_{1,N} = m_{N,1} = 1$*,* $i \in \{1, \ldots, N-1\}$ *and all other elements are identically* 0*. On the other hand, the coupling matrix of a fully-interconnected network has its diagonal elements identically* 0 *and all the other elements are non-zero.*

Remark 14. *For the sake of controller synthesis, we assume that all subsystems* $\mathfrak{S}_i$*,* $i \in \{1, \ldots, N\}$*, have access to their full-state information. The main goal is to synthesize external inputs in order to satisfy specifications over the states of the interconnected system.*

We now provide a compositional framework for obtaining CBC for interconnected dt-SCS $\mathfrak{S}$ based on CSBCs of subsystems $\mathfrak{S}_i$. Let us assume that there exist a CSBC $\mathbb{B}_i$ as in Definition 16 for each control subsystem $\mathfrak{S}_i, i \in \{1, \ldots, N\}$, with $\eta_i \in \mathbb{R}_{\geq 0}$ and symmetric matrix $\underline{X}_i$ with conformal block partitions $\underline{X}_i^{11}, \underline{X}_i^{12}, \underline{X}_i^{21}, \underline{X}_i^{22}$. We propose the following theorem in order to provide sufficient conditions to obtain a CBC for the interconnected dt-SCS $\mathfrak{S}$ from the CSBCs of subsystems $\mathfrak{S}_i, i \in \{1, \ldots, N\}$.

Theorem 7. *Consider an interconnected dt-SCS* $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ *composed of N control subsystems* $\mathfrak{S}_i$*,* $i \in \{1, \ldots, N\}$*, with an interconnection matrix M. Let the initial and unsafe sets of* $\mathfrak{S}$ *be decomposable as* $X_0 = \prod_{i=1}^{N} X_{0_i}$ *and* $X_u = \prod_{i=1}^{N} X_{u_i}$*, respectively. Assume each control subsystem* $\mathfrak{S}_i$ *admits a CSBC* $\mathbb{B}_i$ *corresponding to the sets* X_{0_i} *and* X_{u_i} *with parameter* η_i *according to Definition 16. If*

$$\sum_{i=1}^{N} \eta_i < N, \tag{3.32}$$

$$\begin{bmatrix} M \\ I_{\tilde{r}} \end{bmatrix}^T \underline{X}^{comp} \begin{bmatrix} M \\ I_{\tilde{r}} \end{bmatrix} \leq 0, \tag{3.33}$$

then

$$\mathbb{B}(x) = \sum_{i=1}^{N} \mathbb{B}_i(x_i) \tag{3.34}$$

is a CBC for the interconnected system $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ *with respect to the sets* X_0 *and* X_u*, where*

$$\underline{X}^{comp} := \begin{bmatrix} \underline{X}_1^{11} & & & \underline{X}_1^{12} & & \\ & \ddots & & & \ddots & \\ & & \underline{X}_N^{11} & & & \underline{X}_N^{12} \\ \underline{X}_1^{21} & & & \underline{X}_1^{22} & & \\ & \ddots & & & \ddots & \\ & & \underline{X}_N^{21} & & & \underline{X}_N^{22} \end{bmatrix}, \tag{3.35}$$

and $\tilde{r} = \sum_{i=1}^{N} r_i$ *where* r_i *is the dimension of the internal output of subsystem* $\mathfrak{S}_i$.

Proof. First we show that the CBC $\mathbb{B}$ as (3.34) satisfies conditions (3.27) and (3.28). For any $x := [x_1; \ldots; x_N] \in X_0 = \prod_{i=1}^{N} X_{0_i}$ and from (3.25), we have

$$\mathbb{B}(x) = \sum_{i=1}^{N} \mathbb{B}_i(x_i) \leq \sum_{i=1}^{N} \eta_i = \eta,$$

and similarly for any $x := [x_1; \ldots; x_N] \in X_u = \prod_{i=1}^{N} X_{u_i}$ and from (3.26), one has

$$\mathbb{B}(x) = \sum_{i=1}^{N} \mathbb{B}_i(x_i) \geq N,$$

satisfying conditions (3.27) and (3.28) with $\eta = \sum_{i=1}^{N} \eta_i$ and $\beta = N$. Since $\sum_{i=1}^{N} \eta_i < N$ according to (3.32), one has $\beta > \eta$. Now we show that $\mathbb{B}(x)$ satisfies the condition (3.29) as well. For any $i \in \{1, \ldots, N\}$, let there exist $u_i \in U_i$, with $u = [u_1; \ldots; u_N] \in U$ satisfying the condition (3.27) and internal inputs given as $[w_1; \ldots; w_N] = M[h_{(1)}(x_1); \ldots; h_{(N)}(x_N)]$. Then, we can reach the chain of inequalities in (3.36) which completes the proof. □

$$
\begin{aligned}
&\mathbb{E}\Big[\mathbb{B}(f(x,u,\varsigma)\,\big|\,x,u\Big] - \mathbb{B}(x) = \mathbb{E}\Big[\sum_{i=1}^{N} \mathbb{B}_i(f_{(i)}(x_i,u_i,w_i,\varsigma_i))\,\big|\,x,u,w\Big] - \sum_{i=1}^{N} \mathbb{B}_i(x_i)\\
&= \sum_{i=1}^{N} \mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i,u_i,w_i,\varsigma_i))\,\big|\,x_i,u_i,w_i\Big] - \sum_{i=1}^{N} \mathbb{B}_i(x_i) \leq \sum_{i=1}^{N} \begin{bmatrix} w_i \\ h(x_i) \end{bmatrix}^T \begin{bmatrix} \underline{\mathrm{X}}_i^{11} & \underline{\mathrm{X}}_i^{12} \\ \underline{\mathrm{X}}_i^{21} & \underline{\mathrm{X}}_i^{22} \end{bmatrix} \begin{bmatrix} w_i \\ h(x_i) \end{bmatrix}\\
&= \begin{bmatrix} w_1 \\ \vdots \\ w_N \\ h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix}^T \begin{bmatrix} \underline{\mathrm{X}}_1^{11} & & & \underline{\mathrm{X}}_1^{12} & & \\ & \ddots & & & \ddots & \\ & & \underline{\mathrm{X}}_N^{11} & & & \underline{\mathrm{X}}_N^{12} \\ \underline{\mathrm{X}}_1^{21} & & & \underline{\mathrm{X}}_1^{22} & & \\ & \ddots & & & \ddots & \\ & & \underline{\mathrm{X}}_N^{21} & & & \underline{\mathrm{X}}_N^{22} \end{bmatrix} \begin{bmatrix} w_1 \\ \vdots \\ w_N \\ h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix}\\
&= \begin{bmatrix} M \begin{bmatrix} h_{(1)}(x_1) \\ \ldots \\ h_{(N)}(x_N) \end{bmatrix} \\ h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix}^T \underline{\mathrm{X}}^{comp} \begin{bmatrix} M \begin{bmatrix} h_{(1)}(x_1) \\ \ldots \\ h_{(N)}(x_N) \end{bmatrix} \\ h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix} = \begin{bmatrix} h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix}^T \begin{bmatrix} M \\ I_{\tilde{r}} \end{bmatrix}^T \underline{\mathrm{X}}^{comp} \begin{bmatrix} M \\ I_{\tilde{r}} \end{bmatrix} \begin{bmatrix} h_{(1)}(x_1) \\ \vdots \\ h_{(N)}(x_N) \end{bmatrix} \leq 0.
\end{aligned}
\tag{3.36}
$$

Remark 15. *Condition* (3.33) *is similar to the linear matrix inequality (LMI) that appeared in [12] as a compositional stability condition based on the dissipativity theory. It is shown in [12] that this condition holds independently of the number of subsystems in many physical applications with particular interconnection topologies, e.g., skew-symmetric.*

Conventionally, in order to satisfy the compositionality condition in (3.33), the required parameters for sub-barrier certificates (*i.e.,* conditions (3.25)-(3.27)) are pre-selected and the compositionality condition is checked a posteriori. This method is capable of providing tractable results for large-scale systems where useful knowledge of the interconnection structure and subsystem properties are available (*e.g.*, if the matrix $\underline{\mathrm{X}}$ is fixed for each subsystem [12]). However, in many cases, such information is not apparent. Therefore, obtained control sub-barrier certificates may not satisfy the compositionality condition (3.33) a posteriori and one needs to redesign them from scratch again. In order to design control sub-barrier certificates while having the compositionality condition (3.33) in mind a priori, we employ a distributed optimization method based on an alternating direction method of multipliers (ADMM) algorithm. It allows us to break down a large optimization problem into several smaller sub-problems which can be easier to handle. The solution to the optimization problem provides us with suitable control sub-barrier certificates along with local controllers, satisfying the compositionality condition (3.33) and, hence, allowing the computation of control barrier certificates for the interconnected system.

3.4.3 Compositional Certification using ADMM Algorithm

In this subsection, we discuss the ADMM algorithm [21] which allows us to decompose the condition (3.33) into local sub-problems and a global one involving the interconnection matrix M, as well as matrices $\underline{\mathrm{X}}_i$, $\forall i \in \{1,\dots,N\}$, and $\underline{\mathrm{X}}^{comp}$. Consider an interconnected system $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1,\dots,\mathfrak{S}_N)$. The task of constructing CSBC $\mathbb{B}_i$ of subsystems $\mathfrak{S}_i$, $i \in \{1,\dots,N\}$, can be formulated as a local optimization problem given by

$$\mathcal{S}_i = \Big\{(\underline{\mathrm{X}}_i, \eta_i) \,\Big|\, \exists \text{ CSBC } \mathbb{B}_i \text{ w.r.t. conditions (3.25)-(3.27)}\Big\}. \tag{3.37}$$

Verifying the compositionality condition is a global feasibility problem that can be formulated as

$$\mathcal{G} = \Big\{(\underline{\mathrm{X}}_1,\dots,\underline{\mathrm{X}}_N,\eta_1,\dots,\eta_N) \,\Big|\, \text{conditions (3.32) – (3.33) are satisfied}\Big\}. \tag{3.38}$$

We now restate Theorem 7 as a feasibility problem by utilizing $\mathcal{S}_i$ and $\mathcal{G}$ in the following lemma.

Lemma 3. *Consider an interconnected system $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1,\dots,\mathfrak{S}_N)$. If there exist matrices $\underline{X}_1,\dots,\underline{X}_N$ and constants $\eta_1,\dots,\eta_N$ such that*

$$(\underline{X}_i,\eta_i) \in \mathcal{S}_i, \quad \forall i \in \{1,\dots,N\}, \tag{3.39}$$
$$(\underline{X}_1,\dots,\underline{X}_N,\eta_1,\dots,\eta_N) \in \mathcal{G}, \tag{3.40}$$

then $\mathbb{B}(x) = \sum_i^N \mathbb{B}_i(x_i)$ is a CBC for the interconnected system, where $\mathbb{B}_i$ for $i \in \{1,\dots,N\}$ are obtained when solving the local problem $\mathcal{S}_i$.

In order to convert the feasibility problem of Lemma 3 to an ADMM form, we first define the following indicator functions:

$$\mathbf{I}_{\mathcal{S}_i}(\underline{\mathrm{X}}_i, \eta_i) = \begin{cases} 0, & (\underline{\mathrm{X}}_i, \eta_i) \in \mathcal{S}_i, \\ \infty, & \text{otherwise}, \end{cases}$$

$$\mathbf{I}_{\mathcal{G}}(\underline{\mathrm{X}}_1, \ldots, \underline{\mathrm{X}}_N, \eta_1 \ldots, \eta_N) = \begin{cases} 0, & (\underline{\mathrm{X}}_1, \ldots, \underline{\mathrm{X}}_N, \eta_1, \ldots, \eta_N) \in \mathcal{G}, \\ \infty, & \text{otherwise}. \end{cases}$$

Now, by introducing auxiliary variables $\underline{\mathrm{Z}}_i$ and ζ_i for each subsystem, we can rewrite (3.40) as an optimization problem in the ADMM form as

$$\text{ADMM}: \begin{cases} \min\limits_{d} \sum\limits_{i=1}^{N} \mathbb{I}_{\mathcal{S}_i}(\underline{\mathrm{X}}_i, \eta_i) + \mathbf{I}_{\mathcal{G}}(\underline{\mathrm{Z}}_1, \ldots, \underline{\mathrm{Z}}_N, \zeta_1, \ldots, \zeta_N), \\ \text{s.t.} \quad \underline{\mathrm{X}}_i - \underline{\mathrm{Z}}_i = 0, \quad \forall i \in \{1, \ldots, N\}, \\ \qquad \eta_i - \zeta_i = 0, \quad \forall i \in \{1, \ldots, N\}, \end{cases} \tag{3.41}$$

where $d = (\underline{\mathrm{X}}_1, \ldots, \underline{\mathrm{X}}_N, \eta_1, \ldots, \eta_N, \underline{\mathrm{Z}}_1, \ldots, \underline{\mathrm{Z}}_N, \zeta_1, \ldots, \zeta_N)$. Note that under the satisfaction of constraints in (3.41), minimizing the objective function over d results in $\mathbb{I}_{\mathcal{S}_i}(\underline{\mathrm{X}}_i, \eta_i) = 0$ and $\mathbb{I}_{\mathcal{G}}(\underline{\mathrm{X}}_1, \ldots, \underline{\mathrm{X}}_N, \eta_1 \ldots, \eta_N) = 0$, which is possible if and only if conditions (3.39)-(3.40) are fulfilled. Now, in order to decompose the large optimization problem (3.41) into smaller subproblems, one potential solution is to split the objective function. This is done by introducing the auxiliary variables $\underline{\mathrm{Z}}_i$ and ζ_i that are equivalent to $\underline{\mathrm{X}}_i$ and η_i, $i \in \{1, \ldots, N\}$. The first part of the objective function, *i.e.,* $\mathbb{I}_{\mathcal{S}_i}$, is separable by subsystems, one can find a solution parallelly by iterating over $\underline{\mathrm{X}}_i$, η_i, $\underline{\mathrm{Z}}_i$, and ζ_i, alternately with the help of new scaled dual variables Λ_i and ξ_i which can take real values over corresponding dimensions. Let us denote $\underline{\mathrm{X}}_{1:N} = \{\underline{\mathrm{X}}_1, \ldots, \underline{\mathrm{X}}_N\}$. Similarly, we use notations $\eta_{1:N}$, $\underline{\mathrm{Z}}_{1:N}$, and $\zeta_{1:N}$, respectively. Then, iterative updating of variables is performed in the following manner:

- For each $i \in \{1, \ldots, N\}$, solve the following local problem:

$$(\underline{\mathrm{X}}_i^{k+1}, \eta_i^{k+1}) = \underset{(\underline{\mathrm{X}}_i, \eta_i) \in \mathcal{S}_i}{\operatorname{argmin}} \left\{ \|\underline{\mathrm{X}}_i - \underline{\mathrm{Z}}_i^k + \Lambda_i^k\|_F^2 + (\eta_i - \zeta_i^k + \xi_i^k)^2 \right\}. \tag{3.42}$$

- The solution to (3.42) results in candidate CSBCs $\mathbb{B}_i$ together with controllers ϖ_i, as well as corresponding parameters $\underline{\mathrm{X}}_{1:N}^{k+1}, \eta_{1:N}^{k+1}$. If $(\underline{\mathrm{X}}_{1:N}^{k+1}, \eta_{1:N}^{k+1}) \in \mathcal{G}$, *i.e.*, they satisfy the conditions (3.32)-(3.33), then optimal values of $\underline{\mathrm{X}}_{1:N}^*$ and $\eta_{1:N}^*$ are found as $\underline{\mathrm{X}}_{1:N}^* = \underline{\mathrm{X}}_{1:N}^{k+1}$ and $\eta_{1:N}^* = \eta_{1:N}^{k+1}$. Correspondingly, the CBC $\mathbb{B}$ for the interconnected dt-CS $\mathfrak{S}$ can be obtained via Lemma 3. The algorithm can then be terminated. If not, we solve the following global problem:

$$(\underline{\mathrm{Z}}_{1:N}^{k+1}, \zeta_{1:N}^{k+1}) = \underset{(\underline{\mathrm{Z}}_{1:N}, \zeta_{1:N}) \in \mathcal{G}}{\operatorname{argmin}} \sum_{i=1}^{N} \left\{ \|\underline{\mathrm{X}}_i^{k+1} - \underline{\mathrm{Z}}_i + \Lambda_i^k\|_F^2 + (\eta_i^{k+1} - \zeta_i + \xi_i^k)^2 \right\}. \tag{3.43}$$

- We update our dual variables as

$$\begin{aligned}\Lambda_i^{k+1} &= \underline{\mathrm{X}}_i^{k+1} - \underline{\mathrm{Z}}_i^{k+1} + \Lambda_i^k,\\ \xi_i^{k+1} &= \eta_i^{k+1} - \zeta_i^{k+1} + \xi_i^k,\end{aligned} \tag{3.44}$$

  and return to the first step until a possible convergence.

At each iteration, solutions to the local problems (3.42) for subsystems $\mathfrak{S}_i$ provides one with $\underline{\mathrm{X}}_i$ that is closest to $\Lambda_i - \underline{\mathrm{Z}}_i$ in the Frobenius norm, and η_i that is closest to the value of $\xi_i - \zeta_i$. If $X_{1:N}$ and $\eta_{1:N}$ satisfy the compositionality conditions (3.32)-(3.33), then they satisfy global feasibility conditions in $\mathcal{G}$, and accordingly, we have $\underline{\mathrm{Z}}_{1:N} = \underline{\mathrm{X}}_{1:N}$ as well as $\zeta_{1:N} = \eta_{1:N}$. Correspondingly, the ADMM problem in (3.41) is solved and optimal parameters d are obtained. In the case that $X_{1:N}$ and $\eta_{1:N}$ do not satisfy the compositionality conditions, they are passed to the global problem (3.43) and $Z_{1:N}$ and $\zeta_{1:N}$ are updated. Accordingly, the dual variables Λ_i and ξ_i are also updated via (3.44) and passed back to the local problem (3.42). This procedure is repeated until the solutions converge. In practice, to find the optimal solutions of local problems for subsystems $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$, one may utilize computation techniques such as sum-of-squares optimization (SOS) by suitably parameterizing the CSBCs and corresponding controllers as polynomials, which will be discussed later in Section 3.3.3. Moreover, the global optimization problem may be solved using semi-definite programming (SDP).

Remark 16. *Since objective functions in* (3.41) *are both convex, solutions are guaranteed to converge to optimal ones [84] if the problem is feasible.*

Remark 17. *The ADMM algorithm allows computing CSBCs $\mathbb{B}_i$ that minimize the values of η_i for each subsystem $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$, so that tight probability bounds for the satisfaction of safety specifications may be achieved according to* (3.30)*. Moreover, the computed CSBCs also ensure achieving values of $\underline{\mathrm{X}}_i$ such that the compositionality condition* (3.33) *may be satisfied.*

In the next subsection, we describe how to find CSBCs of subsystems and compute corresponding local controllers by utilizing the SOS programming approach.

3.4.4 Computation of CSBCs and controllers

The ADMM algorithm described in the previous subsection requires the computation of optimal values of $\underline{\mathrm{X}}_i$ and η_i for subsystems $\mathfrak{S}_i$ such that the objective function of the local problem is minimized subject to satisfaction of conditions (3.25)-(3.27). One can reformulate these conditions as a sum-of-squares (SOS) optimization problem to search for suitable CSBC and corresponding local controllers while computing optimal values of $\underline{\mathrm{X}}_i$ and η_i. This can be done by restricting the CSBC to be a non-negative polynomial that can be written as a sum of squares of different polynomials. To do so, we once again utilize an assumption similar to Assumption 3 that restricts the computation of control sub-barrier certificates to stochastic control subsystems with polynomial dynamics.

Assumption 5. *The stochastic control subsystem $\mathfrak{S}_i$ has a continuous state set $X_i \subseteq \mathbb{R}^{n_i}$, and continuous external and internal input sets $U_i \subseteq \mathbb{R}^{m_i}$ and $W_i \subseteq \mathbb{R}^{p_i}$. Its transition map $f_{(i)} : X_i \times U_i \times W_i \times \mathcal{V}_{\varsigma_i} \to X_i$ is a polynomial function of the state x_i, the external input u_i, and the internal input w_i.*

Under Assumption 5, conditions (3.25)-(3.27) can be reformulated as an SOS optimization problem to search for a polynomial CSBC $\mathbb{B}_i$ and a polynomial controller $\varpi_i(\cdot)$ for the subsystem $\mathfrak{S}_i$. The following lemma provides the SOS formulation.

Lemma 4. *Suppose Assumption 5 holds and sets X_i , X_{0_i}, X_{u_i} can be defined by vectors of polynomial inequalities $X_i = \{x_i \in \mathbb{R}^{n_i} \mid b_i(x_i) \geq 0\}$, $X_{0_i} = \{x_i \in \mathbb{R}^{n_i} \mid b_{0_i}(x_i) \geq 0\}$, and $X_{u_i} = \{x_i \in \mathbb{R}^{n_i} \mid b_{u_i}(x_i) \geq 0\}$, where inequalities are provided element-wise. Similarly, let the internal input set W_i be defined by vectors of a polynomial inequality $W_i = \{w_i \in \mathbb{R}^{p_i} \mid b_{w_i}(w_i) \geq 0\}$. Suppose for a given control subsystem $\mathfrak{S}_i$, there exists a sum-of-squares polynomial $\mathbb{B}_i(x_i)$, a constant $\eta_i \in \mathbb{R}_{\geq 0}$, vectors of sum-of-squares polynomials $\lambda_{0_i}(x_i), \lambda_{u_i}(x_i)$, $\lambda_i(x_i)$, $\lambda_{w_i}(w_i)$ and polynomials $\lambda_{v_{ji}}(x_i)$ corresponding to the j^{th} input in $u_i = (u_{1_i}, \ldots, u_{m_{ii}}) \in U_i \subseteq \mathbb{R}^{m_i}$ of appropriate dimensions such that the following expressions are sum-of-squares polynomials:*

$$-\mathbb{B}_i(x_i) - \lambda_{0_i}^T(x_i) b_{0_i}(x_i) + \eta_i, \tag{3.45}$$

$$\mathbb{B}_i(x_i) - \lambda_{u_i}^T(x_i) b_{u_i}(x_i) - 1, \tag{3.46}$$

$$-\mathbb{E}\Big[\mathbb{B}_i(f_{(i)}(x_i, u_i, w_i, \varsigma_i)) \mid x_i, u_i, w_i\Big] + \mathbb{B}_i(x_i) + \begin{bmatrix} w \\ h(x) \end{bmatrix}^T \begin{bmatrix} \underline{X}^{11} & \underline{X}^{12} \\ \underline{X}^{21} & \underline{X}^{22} \end{bmatrix} \begin{bmatrix} w \\ h(x) \end{bmatrix}$$
$$+ \sum_{j=1}^{m_i} (u_{ji} - \lambda_{u_{ji}}(x_i)) - \lambda_i^T(x_i) b_i(x_i) - \lambda_{w_i}^T(w_i) b_{w_i}(w_i). \tag{3.47}$$

Then $\mathbb{B}_i(x_i)$ is a CSBC satisfying conditions (3.25)- (3.27) *and $\varpi_i(x_i) = [\lambda_{u_{1i}}(x_i); \ldots; \lambda_{u_{m_{ii}}}(x_i)], \forall i \in \{1, \ldots, N\}$, is the corresponding controller for the subsystem $\mathfrak{S}_i$.*

Proof. Since $\lambda_{0_i}(x_i)$ in (3.45) is sum-of-squares, we consequently have that the $\lambda_{0_i}^T(x_i) b_{0_i}(x_i) \geq 0$ in the region described by $X_{0_i} = \{x_i \in \mathbb{R}^{n_i} \mid b_{0_i}(x_i) \geq 0\}$. Since $\mathbb{B}_i(x_i)$ is also sum-of-squares and thus non-negative, condition (3.45) directly implies the satisfaction of condition condition (3.25). Similarly, we can show that (3.46) implies condition (3.26). Now, consider and (3.47). If we choose the control input $u_{ji} = \lambda_{u_{ji}}(x_i)$, then since the terms $\lambda_i^T(x_i) b_i(x_i)$ and $\lambda_{i_{w_i}}^T b_{w_i}(w_i)$ are non-negative over X and W, we can prove that it implies (3.27). This completes the proof. □

Remark 18. *Note that one can compute the expected value in* (3.47) *by utilizing the moments of the distribution of ς_i when the distribution of ς_i is known.*

Remark 19. *Our proposed computational method is based on SOS optimization in combination with ADMM algorithm and it relies on the assumption that sub-barrier certificates are polynomial. However, there are other methods for the computation of barrier certificates such as the counter-example guided inductive synthesis (CEGIS) [63] where such an assumption is not required, but at the cost of paying more computational complexity. One may also utilize neural network representations of barrier certificates [140], but at the cost of lacking formal guarantees.*

Remark 20. *The computational complexity of the CBC construction is obtained by analyzing the complexity in the iteration steps of the ADMM algorithm presented in Section 3.4.3. In general, the complexity of searching for a CSBC satisfying* (3.45)-(3.47) *for each subsystem is polynomial with respect to the number of state and input variables [131]. This corresponds to the complexity of solving the local optimization problem* (3.42)*. On the other hand, the global optimization problem* (3.43) *involves solving the LMI* (3.33)*, whose complexity is cubic with respect to the number of subsystems N (or the size of the interconnection matrix M). However, under certain sparsity patterns in the interconnection topology (i.e., sparsity of the matrix M), one can achieve a linear complexity with respect to N [138].*

Remark 21. *The extent of coupling between subsystems can affect the computational complexity in our setting. In particular, if the interconnection topology is too dense (e.g., fully interconnected network), the computational complexity of the ADMM algorithm with LMI and SOS optimization problem potentially increases. More precisely, if each subsystem is affected by more (neighboring) subsystems in the interconnection topology, finding sub-barrier certificates satisfying condition* (3.47) *as well as solving LMI* (3.33) *via ADMM algorithm becomes more complex. This follows directly due to Remark 20.*

3.4.5 Comparison with Small-Gain Approach

In this subsection, we briefly compare the dissipativity-based compositional framework presented in this section with the small-gain one presented in Section 3.3. First and foremost, it is observed that the dissipativity-based approach presented in this section is potentially less conservative than the small-gain approach since the dissipativity-type compositional reasoning can enjoy the structure of the interconnection topology and may not require any constraints on the number or gains of the subsystems. For instance, for large-scale stochastic systems with a *skew-symmetric* interconnection matrix M, one can observe that the subsystems' gains are large which prevents satisfying the small-gain compositionality condition [84]. However, dissipativity-type compositionality conditions can be readily fulfilled irrespective of gains of subsystems or the size of interconnection matrix M (see Remark 15). Second, the small-gain approach requires the satisfaction of a circular compositionality condition (3.15). Unfortunately, there is no *systematic* way to check the satisfaction of this compositionality condition *unless* we restrict the gain of each subsystem to be strictly less than identity. In this case, the condition could be very conservative since the main point of this circularity condition is to allow some subsystems to compensate the undesirable effects of others in the interconnected network which is not the case if all of them are less than identity. In addition, finding the omega-path ϱ_i introduced in the circular condition is very challenging especially if the setting is stochastic. The main reason is that, in the stochastic case, $\max_i \varrho^{-1}$ must be *concave*, Since $\max_i s_i$ itself is not concave, the results in Section 3.3 assumes all subsystems to be the same (*i.e.,* homogeneous) with an identity omega-path $\varrho_i(s) = s$, $\forall i \in \{1, \ldots, N\}$, which gives us the following concave function (identity function)

$$\max_i \varrho_i^{-1}(s) = \max_i(s) = s.$$

As it can be observed, this condition is very restrictive, whereas in the dissipativity approach proposed in this section, the compositionality condition is a simple linear matrix inequality (3.33). As stated in Remark 15, this is well-defined in the relevant literature and can be readily checked via semi-definite programming (SDP).

Third, the small-gain approach proposed in Section 3.3 requires the satisfaction of an additional condition (3.3), which is not required for providing results with the dissipativity-based approach. This makes finding CSBCs much easier. Fourth, the small-gain approach presented in Section 3.3 requires one to first compute CSBCs for each subsystem and then check the compositionality condition a-posteriori. In the case that the compositionality condition is not fulfilled, one requires to re-design the CSBCs from scratch. Whereas in the dissipativity-based approach, by proposing the ADMM algorithm, we are able to combine the computation of CSBCs in (3.25)-(3.27) with the satisfaction of compositionality conditions (3.32)-(3.33). Such optimization is especially useful when the structural properties of the interconnected system are not apparent [84]. Moreover, the algorithm also allows minimizing the values of η_i in (3.25) for each subsystem $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$ so that tight probabilistic bounds for the satisfaction of safety specifications may be achieved according to (3.30) (see Remark 17).

However, it should be noted that our proposed dissipativity approach also has a drawback compared to the small-gain one in Section 3.3. In particular, condition (3.27) is more restrictive than the one in (3.6). The effects of other subsystems in (3.27) is captured with a quadratic supply rate rather than utilizing a more general $\mathcal{K}_\infty$ one in (3.6). This quadratic restriction in our setting is required to pose the compositionality condition as an LMI in (3.33).

3.4.6 Case Study

To demonstrate the effectiveness of our results, we apply our approach to a room temperature regulation problem in a circular building. We consider system dynamics similar to the one considered in Section 3.3.4, but with different parameters. Moreover, we include stochasticity as multiplicative noise. The evolution of the temperature $T(\cdot)$ in the interconnected system is governed by the following dynamics:

$$\mathfrak{S} : T(t+1) = AT(t) + \mu T_H \nu(t) + \theta T_E + 0.01\varsigma(t)T(t),$$

where $A \in \mathbb{R}^{N\times N}$ is a matrix with diagonal elements given by $\bar{a}_{ii} = (1 - 2\alpha - \theta - \mu\nu_i(t))$, off-diagonal elements $\bar{a}_{i,i+1} = \bar{a}_{i+1,i} = \bar{a}_{1,N} = \bar{a}_{N,1} = \alpha$, $i \in \{1, \ldots, N-1\}$, and all other elements are identically zero. The parameters $\alpha = 0.005$, $\theta = 0.06$ and $\mu = 0.145$ are conduction factors between rooms i and $i \pm 1$, external environment and room i, heater and room i, respectively. The heater temperature is maintained at $T_H = 40\,^\circ\mathsf{C}$ and the outside temperature $T_{ei} = -5\,^\circ\mathsf{C}$ for all rooms $i \in \{1, \ldots, N\}$. We also have $T(t) = [T_1(t); \ldots; T_N(t)]$, $T_E = [T_{e1}; \ldots; T_{eN}]$, $\nu(t) = [\nu_1(t); \ldots; \nu_N(t)]$ and $\varsigma(t) = \mathsf{diag}(\varsigma_1(t), \ldots, \varsigma_N(t))$. The state, initial, and unsafe sets are given by $X = [0, 20]$, $X_0 = [17, 18]$, $X_u = [0, 15]$, respectively. The requirement of our case study is to synthesize a controller $\nu : \mathbb{N} \to [0, 0.6]^N$ satisfying the safety specification with respect to X_0 and X_u.

To do this, we consider our network $\mathfrak{S}$ as an interconnection of $N = 300$ subsystems, each of which constitutes a room. The state evolution of these individual subsystems is given by

$$\mathfrak{S}_i : T_i(t+1) = \bar{a}T_i(t) + \mu T_H \nu_i(t) + \alpha w_i(t) + \theta T_{ei} + 0.01\varsigma_i(t)T_i(t).$$

It can be easily verified that $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ with coupling matrix M such that $m_{i,i+1} = m_{i+1,i} = m_{1,N} = m_{N,1} = 1, i \in \{1, \ldots, N-1\}$ and all other elements are identically zero. We now utilize the ADMM algorithm in conjunction with SOS formulation with the help of YALMIP tool [78, 79] to compute CSBCs for subsystems $\mathfrak{S}_i$, $i \in \{1, \ldots, N\}$. We then obtain CSBC as a 4^{th} order polynomial given by $\mathbb{B}_i(T_i) = 9.6445 - 0.6911T_i + 0.1396T_i^2 - 0.0163T_i^3 + 0.0005T_i^4$ and the corresponding controller is computed to be $\varpi_i(T_i) = 0.59 - 0.011T_i$. Parameters satisfying conditions (3.25)-(3.27) are obtained as $\eta_i = 0.0594$ and $\underline{\mathrm{X}}_i = 10^{-3} \times \begin{bmatrix} 0.1348 & 0.0001 \\ 0.0001 & -0.5591 \end{bmatrix}$. One can readily verify that compositionality conditions of Theorem 7 are satisfied with $\eta = 59.4$, $\beta = 1000$ and $\underline{\mathrm{X}}^{comp}$ obtained from $\underline{\mathrm{X}}_i$, $i \in \{1, \ldots, 300\}$, via equation (3.35). Therefore, the overall CBC of the interconnected system is obtained to be $\mathbb{B}(T) = \sum_{i=1}^{300}(9.6445 - 0.6911T_i + 0.1396T_i^2 - 0.0163T_i^3 + 0.0005T_i^4)$, while the suitable controller for the trajectories in $X^0 = L^{-1}(p_0)$ is obtained as $\varpi(T) = [0.59 - 0.011T_1; \ldots; 0.59 - 0.011T_{300}]$. Then, by utilizing the results of Corollary 2, we obtain the probability lower bound on the satisfaction of the safety specification as 0.94.

Figure 3.5 shows the simulation for state trajectories of a representative room in the network for 10 different noise realizations when starting from region X^2. The computation of CSBC and corresponding local controller take up to 240 seconds on a machine with Linux Ubuntu 18.04 OS (Intel i7-8665U CPU with 32GB RAM).

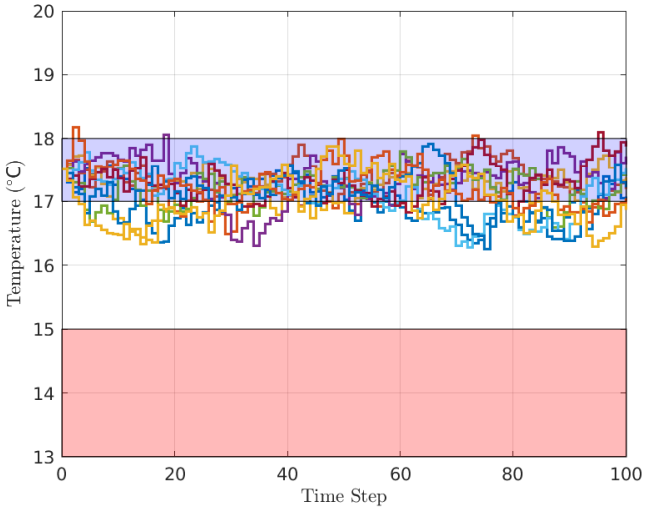


Figure 3.5: Closed-loop state trajectories of a representative room for 10 noise realizations in a network of 300 rooms, starting from a state in X_0. The region X_0 is shown in purple and X_u is shown in pink.

3.5 Conclusion

In this chapter, we proposed a compositional framework for the modular construction of control barrier certificates for large-scale stochastic control systems. In particular, we considered the large-scale stochastic control system as an interconnected one composed of smaller subsystems and proposed the so-called control sub-barrier certificates for subsystems, which are computed along with local controllers. Then, we provided two different compositional approaches, based on small-gain and dissipativity theories, respectively, for utilizing the control sub-barrier certificates to construct control barrier certificates and controllers for the large interconnected system. Using these control barrier certificates, we were then able to obtain probabilistic guarantees for the satisfaction of safety specifications. We obtained probabilistic lower bounds on safety satisfaction over both finite and infinite time horizons.

We also provided a comparison between the two compositional approaches based on the small-gain theorem as well as the dissipativity theory. We reasoned that the results based on dissipativity theory are less conservative than the small-gain-based approach in the sense that the compositionality conditions can be solved irrespective of the number or gains of the subsystem. Moreover, the dissipativity-based compositionality condition is easier to check due to it being a simple LMI, and as a result, one is able to optimize the construction of control sub-barrier certificates with respect to the satisfaction of the compositionality condition. In this chapter, we utilized an ADMM-based optimization problem in order to do the same. As a result, the construction of control barrier certificates was also optimized to achieve tight probabilistic lower bounds on safety satisfaction.

Finally, we illustrated the computation of control sub-barrier certificates for subsystems via SOS programming, in the case of both small-gain and dissipativity-based approaches. Moreover, we presented various case studies to demonstrate the effectiveness of the proposed approaches.

Chapter 4

Formal Verification using k-Inductive Barrier Certificates

4.1 Introduction

Safety verification of digital hardware and software systems has been traditionally performed by utilizing inductive invariants, which are properties of the system that can be shown to hold universally at all the reachable states of the system [42]. On the other hand, barrier certificates utilized for the verification of continuous-state systems can be viewed as continuous counterparts of inductive invariants. As described in the previous chapters, barrier certificates for discrete-time non-stochastic dynamical systems are constructed such that, at every time step of the system evolution, the value of the barrier certificate remains within a prescribed level set. As a result, this level set of the barrier certificate can be considered to be the inductive invariant of the continuous time system. This is also true in the context of stochastic dynamical systems, where barrier certificates take the role of inductive expectation invariants, as they require the expected value of the barrier certificates to always be within some level set. However, it must be mentioned that the traditional barrier certificate conditions presented in the previous chapters for (stochastic) dynamical systems can be quite restrictive, as the (expected) value of the barrier certificates is required to be decaying at each time step. Therefore, in the cases that such conditions are not met, one is unable to find suitable barrier certificates for a system even when it satisfies the desired safety or reachability specification.

The k-induction principle, introduced in [112], utilizes more information available about the system in order to provide easier verification conditions, as encountered in software verification [40, 42]. In this chapter, we extend this principle to barrier certificates and propose several notions of k-inductive barrier certificates that generalize the traditional barrier certificates and provide less conservative conditions so that they may be easier to satisfy. As a result, a larger class of functions can behave as k-inductive barrier certificates, while still ensuring the satisfaction of safety or reachability specifications. In particular, we present two different notions of k-inductive barrier certificates for discrete-time dynamical systems which provide qualitative guarantees for the satisfaction of safety specifications. In the context of stochastic dynamical

systems, we present different notions of k-inductive barrier certificates for safety and reachability specifications, respectively, and utilize them to obtain probabilistic guarantees for the satisfaction of those specifications over infinite time horizons.

4.1.1 Related Literature

Safety and Reachability Verification

There are various techniques in the literature to perform verification of safety and reachability specifications for discrete state systems. The widely recognized methods include model checking [15] as well as deductive and inductive verification techniques [45]. The former usually relies on graph reachability computation of finite-state models. Model-checking-based approaches may also be extended to continuous-state systems by constructing finite state abstractions[122, 114, 71, 1]. Unfortunately, these methods suffer from the curse of dimensionality, and as a result, the computational complexity grows exponentially with the number of states in the system. On the other hand, the latter relies on building mathematical proof rules and using logical inferences to obtain safety or reachability guarantees in software systems. This includes contract-based verification [100] and inductive invariants [125] among other approaches. Contract-based approaches have also been extended to continuous-state systems via assume-guarantee contracts [111, 41]. On the other hand, barrier certificates [96] are continuous-state analogues to inductive invariants.

Barrier Certificate-based Approaches

Barrier certificate-based approaches are inherently discretization-free and have gained considerable attention in the past few years. Barrier certificates were first proposed for the verification of safety specifications in the context of non-stochastic dynamical systems in [96, 95] and were later extended to reachability specifications in [99, 67]. These approaches have also been extended to stochastic (hybrid) dynamical systems [97, 61, 129] where supermartingale conditions are imposed to provide probabilistic guarantees for infinite time horizons. These conditions have been relaxed by utilizing c-martingales in [117, 62, 132], but at the cost of providing guarantees only over finite time horizons.

k-induction principle was first used in the context of continuous-time non-stochastic dynamical systems in [16]. In particular, k-induction was combined with time-bounded backward reachability analysis for safety verification. Moreover, the implementation technique used in the paper above assumes a barrier certificate candidate given a priori and utilizes this to verify whether the conditions are satisfied. In contrast, the results presented in this chapter are focused on discrete-time systems, both non-stochastic and stochastic, and do not require any reachability analysis to provide safety guarantees. Moreover, this chapter does not assume any information on the existence of barrier certificates and provides a computational approach to search for suitable barrier certificates.

4.1.2 Contributions

In Chapter 2, we introduced the barrier certificate-based approach for the verification of safety and reachability specifications in the context of discrete-time (stochastic) dynamical systems. These conditions can be restrictive, and as a result, one may not be able to find suitable barrier certificates even when the system trivially satisfies the concerned properties. In this chapter, we propose and investigate several notions of k-inductive barrier certificates that provide less conservative conditions to verify safety and reachability properties. As a result, larger classes of functions may act as barrier certificates, making them easier to find. Moreover, we motivate the use of k-inductive barrier certificates over traditional ones with the help of several simple examples as well as case studies.

This chapter is organized as follows. Section 4.2 introduces the preliminary concepts of induction as well as k-induction. Then, Section 4.3 focuses on the safety verification of discrete-time dynamical systems via k-inductive barrier certificates. In particular, we present two different notions of k-inductive barrier certificates which generalize the barrier certificate conditions by utilizing the k-inductive barrier certificates. We illustrate via a simple example of a finite state transition system that even when one cannot compute suitable barrier certificates satisfying the traditional conditions, it is possible to compute k-inductive barrier certificates and guarantee safety satisfaction (see Example 1). We also illustrate that the second notion of k-inductive barrier certificates for safety is more expressive than the first (see Example 2). Moreover, under some mild assumptions on the dynamics of the system and the regions of interest, we provide two methods to compute k-inductive barrier certificates based on SOS optimization and SMT solvers. We also demonstrate the effectiveness of our approach over a numerical case study.

Section 4.4 is concerned with the probabilistic safety verification of discrete-time stochastic dynamical systems. Similar to Section 4.3, we highlight the restrictiveness of the traditional barrier certificate conditions and motivate the need for k-induction via a simple example (finite Markov chain considered in Example 3). In particular, we show that, due to the supermartingale condition requirement, traditional barrier certificates may not always exist even when the system is known to be safe with probability 1, and as a result, one obtains a trivial probability lower bound of 0 for safety satisfaction. We then propose a new notion of k-inductive barrier certificates which does not impose any supermartingale requirement on the barrier certificate conditions, while still providing probability lower bounds over infinite time horizons. Then, under assumptions on the underlying dynamics of the systems, we provide an SOS-based approach for the computation of k-inductive barrier certificates. We also demonstrate the efficacy of our results via a case study.

In Section 4.5, we are concerned with the probabilistic verification of discrete-time stochastic dynamical systems against reachability specifications. In this context, we propose two different notions of k-inductive barrier certificates that provide probabilistic guarantees of reachability satisfaction over unbounded time horizons. While the first notion allows computing a probability lower bound of satisfying reachability, the second one (if existing) can be utilized to achieve almost sure reachability guarantees (*i.e.*, with probability 1). We demonstrate the utility of our proposed notions using illustrative examples (see Example 4) and provide a computational approach to obtain k-inductive barrier certificates via SOS programming. Finally, we demonstrate the effectiveness of our approach with the help of a case study.

We must mention that the results presented in this chapter appear in publications [8, 9]. In particular, contents presented in Section 4.3 appear in [8], which are results published at the 60th Conference on Decision and Control. On the other hand [9] covers the contents of Section 4.4 and Section 4.5 and has been published at the 25th ACM Conference on Hybrid Systems: Computation and Control. The contents of this chapter are a result of collaboration with Vishnu Murali, Ashutosh Trivedi, and Majid Zamani. The author of the thesis is credited with the majority of the technical results, implementation, as well as preparation of the manuscript. Vishnu Murali is credited particularly with the formulation of the examples, computation of k-inductive barrier certificates via δ-complete decision procedures in Section 4.3.2, partial implementation of the results of Section 4.3.3, Definition 22 and Theorem 11 of Section 4.5, as well as presenting the figures and revising the manuscript. Ashutosh Trivedi and Majid Zamani provided the necessary support and supervision.

4.2 The k-Induction Principle

In this section, we introduce the preliminary concept of k-induction, which will be utilized throughout this chapter. In general, an inductive proof for a property P consists of a base case, an inductive hypothesis and an inductive step. In a standard inductive proof, also called weak induction, the inductive hypothesis consists of an assumption that the property P holds at any given step, which is used to imply that the property also holds in the next step. Formally, the induction for property P is formulated as follows:

$$\Big(P(0) \wedge \underset{t\in\mathbb{N}}{\forall} \big(P(t) \Longrightarrow P(t{+}1)\big)\Big) \Longrightarrow \underset{t\in\mathbb{N}}{\forall} P(t).$$

On the other hand, the inductive hypothesis of k-induction assumes that the property P holds at all steps until the k^{th} step. The stronger inductive hypothesis weakens the need to enforce the consequent due to the availability of additional information. Mathematically, a k-inductive proof for property P is described as:

$$\left(\bigwedge_{0\le i<k} P(i) \wedge \underset{t\in\mathbb{N}}{\forall}\left(\bigwedge_{0\le i<k} \big(P(t{+}i)\big) \Longrightarrow P(t{+}k)\right)\right) \Longrightarrow \underset{t\in\mathbb{N}}{\forall} P(t).$$

In k-induction, the base case requires the property to be shown to hold true in the first k steps. The inductive step then allows us to show that if the property P holds true in k consecutive steps, then consequently it holds true in the $(k+1)$th step as well and so it must hold true for any time step.

4.3 Safety Verification of Dynamical Systems

In this section, we consider non-stochastic discrete-time dynamical systems (dt-DS) defined in Definition 1 in the absence of control inputs:

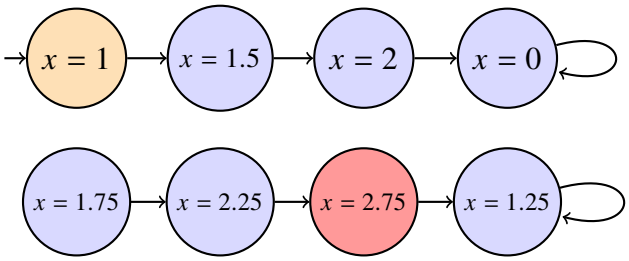


Figure 4.1: A finite state system $\mathfrak{S}$ with the unsafe state shaded in red.

$$\mathfrak{S} : \mathbf{x}(t+1) = f(\mathbf{x}(t)), \tag{4.1}$$

where, similar to Definition 1, $\mathbf{x} : \mathbb{N} \to X$ is the state sequence of the system, and $\mathbf{x}_{x_0}$ refers to the state sequence starting from the initial condition $\mathbf{x}_{x_0}(0) = x_0$. We are interested in verifying safety specifications for $\mathfrak{S}$ such that the state sequences $\mathbf{x}_{x_0}$ do not visit any unsafe regions. In other words, we would like to provide a solution to Problem 1 described in Chapter 2.

One potential solution to solve Problem 1 is to utilize a barrier certificate-based approach, as described in Section 2.5. By considering barrier certificates for safety as in Definition 5, one is able to provide sufficient conditions for the satisfaction of safety specifications. In particular, to prove that a system is safe, it suffices to discover a barrier certificate. The search for a barrier certificate can be performed in a principled fashion by restricting the search space to a given template (*e.g.*, polynomial functions of a specified degree) and then employing appropriate search techniques such as sum-of-squares (SOS) programming [93] or satisfiability modulo theory (SMT) solvers [39] to compute barrier certificates satisfying conditions (2.8)-(2.10). However, condition (2.10) can be quite conservative as it requires the desired barrier certificate to decay at every time step/transition. Therefore, in many cases, no suitable barrier certificate with a given template can be found even when the system is trivially safe. This is illustrated with the help of a simple finite system, as described below.

Example 1. *Consider a finite system $\mathfrak{S}$ as shown in Figure 4.1 (©2021 IEEE) with $x \in X = \{0, 1, 1.25, 1.5, 1.75, 2, 2.25, 2.75\}$ as the states of the system with $x = 1$ as the initial state and $x = 2.75$ as the unsafe state. We want to verify that the unsafe state $x = 2.75$ is never visited by any state sequence of the system. One can immediately see that this property trivially holds, as there is no way to reach the state $x = 2.75$ from the initial state $x = 1$. Unfortunately, we cannot utilize barrier certificates provided in Definition 5 to obtain safety guarantees for the system, when, for instance, the template of barrier certificates is fixed to be linear, i.e., $\mathbb{B}(x) = ax + b$. This can be shown as follows. From condition* (2.8) *for the initial state $x = 1$, we have $a + b <= 0$. Similarly, from condition* (2.9) *for the unsafe state $x = 2.75$, we have that $2.75a + b > 0$. Now, utilizing condition* (2.10) *for the transition from the state $x = 2$ to $x = 0$, we get the inequality $a \geq 0$. Similarly, for the transition from the state $x = 1.5$ to $x = 2$, we obtain $a \leq 0$. The only possible solution is $a = 0$, from which we obtain the barrier certificate as $\mathbb{B}(x) = b$. But this results in a contradiction between conditions* (2.8) *and* (2.9)*. Therefore, no linear barrier certificate exists for this system, even though the system satisfies the safety property.*

One can observe that the standard notion of the barrier certificates as in Definition 5 are comparable to inductive proofs. To clarify this, condition (2.8) presents as the base case, where the values of the barrier certificate at initial states of the system are such that the state sequence always begins from the safe set. Then, condition (2.10) is analogous to the inductive step which ensures that the value of the barrier certificate is non-increasing at each time step so that the state sequence never reaches the unsafe set due to condition (2.9). However, due to the weaker construction of the inductive hypothesis, condition (2.10) is difficult to satisfy.

In the remainder of this section, we seek to alleviate the conservatism of the traditional barrier certificate conditions (2.8)-(2.10) of Definition 5. In particular, we leverage the k-induction principle, often utilized in the context of software verification [42, 22], and propose a modified notion of barrier certificates, that we call *k-inductive barrier certificates*. We show that the constraints of these barrier certificates are more relaxed than the original ones, while still guaranteeing safety satisfaction. This makes the discovery of barrier certificates easier, as more functions will satisfy the k-inductive barrier certificate conditions. For instance, we show that Example 1 admits a k-inductive linear barrier certificate even though it does not admit a standard one. In what follows, we introduce two different notions of k-inductive barrier certificates that utilize this principle to provide sufficient conditions for ensuring safety.

4.3.1 k-Inductive Barrier Certificates for Safety

In this subsection, we introduce two different notions of k-inductive barrier certificates and illustrate their merits with examples. First, we present the following notion.

Definition 19. *We say that a function $\mathbb{B} : X \to \mathbb{R}$ is a k-*inductive barrier certificate *for the dt-DS $\mathfrak{S}$ with respect to a set of initial states $X_0 \subseteq X$ and a set of unsafe states $X_u \subseteq X$, if there exist $k \in \mathbb{N}_{\geq 1}$, $\epsilon \in \mathbb{R}_{\geq 0}$, and $d > k\epsilon$ such that following conditions hold:*

$$\mathbb{B}(x) \leq 0 \qquad \text{for all } x \in X_0, \tag{4.2}$$
$$\mathbb{B}(x) \geq d \qquad \text{for all } x \in X_u, \tag{4.3}$$
$$\mathbb{B}(f(x)) - \mathbb{B}(x) \leq \epsilon \qquad \text{for all } x \in X, \tag{4.4}$$
$$\mathbb{B}(f_k(x)) - \mathbb{B}(x) \leq 0 \qquad \text{for all } x \in X. \tag{4.5}$$

Now, we present the first result of this chapter and utilize the above definition of k-inductive barrier certificates to provide safety guarantees.

Theorem 8. *Consider a discrete-time dynamical system $\mathfrak{S}$. If there exists a k-inductive barrier certificate $\mathbb{B} : X \to \mathbb{R}$ for $\mathfrak{S}$ such that it is a k-inductive barrier certificate as in Definition 19 with respect to initial set $X_0 \subseteq X$ and unsafe set $X_u \subseteq X$, then state sequences $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ starting from $x_0 \in X_0$ never reach the unsafe region X_u.*

Proof. We begin the proof by assuming that there exists a function $\mathbb{B} : X \to \mathbb{R}$ such that conditions (4.2)-(4.5) hold but the system is not safe, *i.e*, there exists some time $T \in \mathbb{N}$ such that $\mathbf{x}(T) \in X_u$. Let $T = ik + k'$, for some $i \in \mathbb{N}$ and $k' < k$. For a state sequence $\mathbf{x}_{x_0}$ starting from

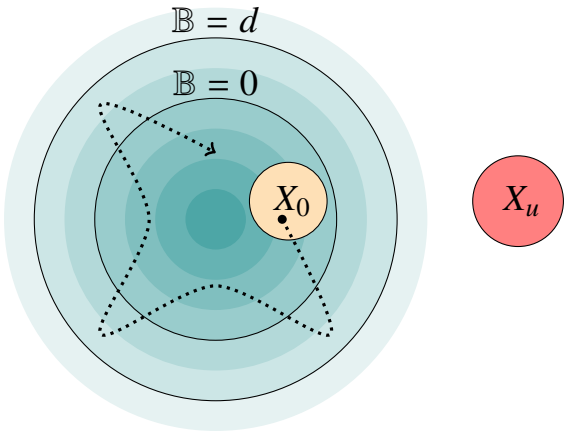


Figure 4.2: Safety verification using k-inductive barrier certificates presented in Definition 19.

$\mathbf{x}_{x_0}(0) = x_0$, from conditions (4.2) and (4.3), we have $\mathbb{B}(x_0) \leq 0$ and $\mathbb{B}(\mathbf{x}_{x_0}(ik + k')) \geq d$. From condition (4.4) and induction, we have that

$$\mathbb{B}(\mathbf{x}_{x_0}(ik + k')) \leq \mathbb{B}(\mathbf{x}_{x_0}(ik)) + k'\epsilon \leq \mathbb{B}(\mathbf{x}_{x_0}(ik)) + k\epsilon.$$

From (4.5) and induction, we get $\mathbb{B}(\mathbf{x}_{x_0}(ik)) \leq B(x_0)$. From the two inequalities and condition (4.2), we obtain

$$\mathbb{B}(\mathbf{x}_{x_0}(ik + k')) - \mathbb{B}(x_0) \leq k\epsilon < d.$$

This is a contradiction to (4.3), implying that $\mathbf{x}_{x_0}(T) \notin X_u$. Therefore, the state sequences $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ that begin from $x_0 \in X_0$ remain in the safe regions. □

Remark 22. *Note that the first definition of k-inductive barrier certificates presented in this paper (see Definition 19) is similar to the Type 2 barrier functions studied in [52]. However, our formulation is in the context of discrete-time systems rather than continuous-time ones as in [52]. Moreover, conditions in [52] ask for strict contraction of the barrier certificates from the initial set after a certain time period, whereas our conditions are more relaxed.*

Safety verification utilizing k-inductive barrier certificates presented in Definition 19 is demonstrated in Figure 4.2. In condition (4.5), the value of the barrier certificate needs to be non-increasing only after every k time steps rather than at each time step. However, the additional condition (4.4) is required to ensure that state sequences do not reach unsafe regions within k time steps. Note that when $k = 1$, $\epsilon = 0$, and $d \in \mathbb{R}_{>0}$ is a small positive number, conditions (4.2)-(4.5) reduce to standard barrier certificate conditions (2.8)-(2.10). We now illustrate k-inductive barrier certificates as in Definition 19 by utilizing the finite system considered in Example 1.

Example 1 (Continued)**.** *Consider the finite system shown in Figure 4.1 (©2021 IEEE). In the previous section, we proved that no linear barrier certificate satisfying* (2.8)-(2.10) *exists for the system. Now, we show that we can instead utilize k-inductive barrier certificates to guarantee that the system indeed satisfies safety specifications. Consider a linear k-inductive barrier certificate*

as in Definition 19 given by $\mathbb{B}(x) = x-1$ *with* $k = 3$. *Then, by assigning* $\epsilon = 0.5$ *and* $d = 1.6$, *it can be immediately observed that condition* (4.2) *is satisfied for initial state* $x = 1$ *and similarly,* (4.3) *is satisfied for unsafe state* $x = 2.75$. *Moreover, for all* 1*-step transitions possible in the system, we can verify that condition* (4.4) *also holds. Similarly, for all the possible* 3*-step transitions from the initial state to the unsafe state, condition* (4.5) *is valid. Therefore, it can be inferred that* $\mathbb{B}(x) = x - 1$ *is indeed a* 3*-inductive barrier certificate as in Definition 19 that verifies the safety of the system.*

As required by the k-induction principle, k-inductive barrier certificates in Definition 19 ensure the safety of the system for k consecutive steps by ensuring only a bounded increase at every step via condition (4.4). The non-increase of $\mathbb{B}(x)$ after every k steps via condition (4.4) is required to ensure the safety of the system at the $(k+1)^{th}$ step, thus capturing the inductive step of the proof. Note that the system remains safe as long as the barrier certificate remains in the set $\mathbb{B}(x) < d$ for all time. However, due to condition (4.5), the barrier certificate cannot stay in the set $\mathbb{B}(x) < d$ forever and must eventually return to the set $\mathbb{B}(x) \leq 0$, leading to some conservatism in the approach.

To better capture the requirements of k-induction, we present a different notion of k-inductive barrier certificates which provides us with less conservative conditions.

Definition 20. *We say that a function* $\mathbb{B} : X \to \mathbb{R}$ *is a* k*-inductive barrier certificate for the dt-DS* $\mathfrak{S}$ *with respect to a set of initial states* $X_0 \subseteq X$ *and a set of unsafe states* $X_u \subseteq X$, *if there exists* $k \in \mathbb{N}_{\geq 1}$ *such that following holds:*

$$\bigwedge_{0\leq i<k} \mathbb{B}(f_i(x)) \leq 0 \qquad \text{for all } x \in X_0, \tag{4.6}$$

$$\mathbb{B}(x) > 0 \qquad \text{for all } x \in X_u, \tag{4.7}$$

$$\bigwedge_{0\leq i<k} (\mathbb{B}(f_i(x)) \leq 0) \implies \mathbb{B}(f_k(x)) \leq 0 \qquad \text{for all } x \in X. \tag{4.8}$$

Now we present the following theorem to utilize k-inductive barrier certificates as in Definition 20 to obtain safety guarantees for the dt-DS $\mathfrak{S}$

Theorem 9. *Consider a dt-DS* $\mathfrak{S}$. *If there exists a function* $\mathbb{B} : X \to \mathbb{R}$ *for* $\mathfrak{S}$ *such that it is a* k*-inductive barrier certificate as in Definition 20 with respect to initial set* $X_0 \subseteq X$ *and unsafe set* $X_u \subseteq X$, *then the state sequences* $\mathbf{x}_{x_0}$ *of* $\mathfrak{S}$ *starting from* $x_0 \in X_0$ *never reach the unsafe region* X_u.

Proof. Assume that k-inductive barrier certificate as in Definition 20 exists for the dt-DS $\mathfrak{S}$ but it is not safe, *i.e.*, for some state sequence $\mathbf{x}_{x_0}$ starting from x_0, there exists some $T \in \mathbb{N}$ such that $\mathbb{B}(\mathbf{x}_{x_0}(T)) \in X_u$. Then, from condition (4.7), we must have $\mathbb{B}(\mathbf{x}_{x_0}(T)) > 0$. Condition (4.6) implies that $\mathbb{B}(\mathbf{x}_{x_0}(0)) \leq 0$, $\mathbb{B}(\mathbf{x}_{x_0}(1)) \leq 0, \ldots, \mathbb{B}(\mathbf{x}_{x_0}(k-1)) \leq 0$, which means that the state sequences starting from the safe set will definitely stay in the safe set for the next $k-1$ consecutive time steps. From condition (4.8), we have that for any given k consecutive time steps, if the system is safe, then the system will remain safe in the $(k+1)^{\text{th}}$ time step. Then, by applying

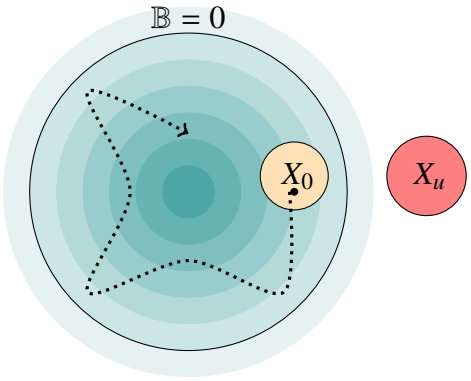


Figure 4.3: Safety verification using k-inductive barrier certificates presented in Definition 20.

the k-induction proof rule with (4.6) as the base case and (4.8) as the inductive hypothesis, we have that $\mathbb{B}(\mathbf{x}_{x_0}(t)) \leq 0$ for all $t \in \mathbb{N}$. This is contradictory to condition (4.7). Therefore, the state sequences $\mathbf{x}_{x_0}$ of system $\mathfrak{S}$ starting from $x_0 \in X_0$ remain safe for all time. □

Note that Definition 20 is similar to t-barrier certificates presented in [16] for continuous-time systems, except the latter uses the contrapositive equivalent of the logical implication in the inductive step in combination with backwards reachability analysis. We observe that for the value of $k = 1$, any k-inductive barrier certificate satisfying (4.2)-(4.5) also satisfies (4.6)-(4.8). Furthermore, unlike conditions (2.13) or (4.4), condition (4.8) does not impose a non-increasing or bounded increase requirement between $\mathbb{B}(f(x))$ and $\mathbb{B}(x)$. The graphical demonstration of k-inductive barrier certificates as in Definition 20 is presented in 4.3. We now illustrate k-inductive barrier certificates as in Definition 20 by once again considering the finite system in Example 1.

Example 1 (Continued)**.** *Consider the finite system shown in Figure 4.1. We now utilize k-inductive barrier certificates as in Definition 20 to prove that the system satisfies the required safety specification. Let $\mathbb{B}$ be a linear function defined as $\mathbb{B}(x) = x - 2$. We show that $\mathbb{B}$ is a k-inductive barrier certificate as in Definition 20 with $k = 2$ as follows. Condition* (4.6) *is shown to be satisfied because $\mathbb{B}(x) \leq 0$ for the state $x = 1$ and its consecutive states $x = 1.5$ and $x = 2$. Similarly, condition* (4.7) *is true as $\mathbb{B}(x) > 0$ at the unsafe state $x = 2.75$. In order to show condition* (4.8)*, we first consider all states x where $\mathbb{B}(x) \leq 0$. The set of states which satisfy this condition is $\{0, 1, 1.25, 1.5, 1.75, 2\}$. For any two-step transitions from the states in the set $\{0, 1, 1.25, 1.5, 2\}$, we see that the antecedent of* (4.8) *holds, and so does the consequent. Hence, the logical implication in condition* (4.8) *holds. For the state $x = 1.75$, even though $\mathbb{B}(x) \leq 0$, after one transition we have $\mathbb{B}(f(x)) > 0$. Therefore, the antecedent automatically fails to hold and condition* (4.8) *is satisfied. Finally for the remaining states in $\mathfrak{S}$, since $\mathbb{B}(x) > 0$, the antecedent is false and so condition* (4.8) *is true. The logical implication is valid in all the cases, which proves that the function $\mathbb{B}(x) = x - 2$ is indeed a 2-inductive barrier certificate that guarantees the satisfaction of safety.*

We now illustrate the merits of k-inductive barrier certificates as in Definition 20 over those of Definition 19 by showing that there exist systems that do not admit linear k-inductive barrier

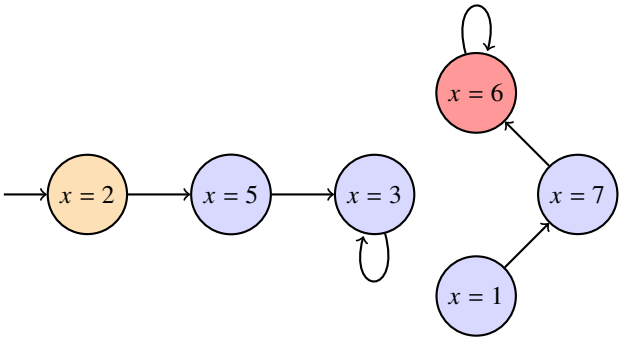


Figure 4.4: A finite state system $\mathfrak{S}'$ with the unsafe state shaded in red.

certificates satisfying conditions (4.2)-(4.5) for any $k \in \mathbb{N}_{\geq 1}$, but admit linear k-inductive barrier certificates satisfying conditions (4.6)-(4.8) for some $k \in \mathbb{N}_{\geq 1}$.

Example 2. *Let us consider the finite system $\mathfrak{S}'$ shown in Figure 4.4 (©2021 IEEE) with $x \in X = \{1, 2, 3, 5, 6, 7\}$ as the states of the system, $x = 2$ as the initial state and $x = 6$ as the unsafe state. Similar to Example 1, we see that the system is trivially safe. Suppose that a k-inductive barrier certificate as in Definition 19 of the form $\mathbb{B}(x) = ax + b$ exists for the system. When $k = 1$, there exists no such k-inductive barrier certificate based on the discussion in Example 1. Therefore, we assume that $k \geq 2$. We let d and ϵ take any value in $\mathbb{R}_{>0}$ such that $d > k\epsilon$. Due to the self-loop at the state $x = 3$, we have $f^k(3) = 3$ for any k. By applying condition* (4.2) *at the initial state $x = 2$, we get $\mathbb{B}(x = 2) = 2a + b \leq 0$. Similarly, by applying condition* (4.3) *at the unsafe state $x = 6$, we have $6a + b \geq d > 0$. From the above inequalities, we get $a \geq 0$. Now, from condition* (4.5) *for a k-step transition from $x = 2$ to $x = 3$ for any $k \geq 2$, we have $\mathbb{B}(3) - \mathbb{B}(2) = a \leq 0$. This results in a contradiction, concluding that there exists no linear k-inductive barrier certificate as in Definition 19 for the system $\mathfrak{S}'$ and therefore, safety cannot be verified. Now, we show that formulating k-inductive barrier certificates via Definition 20 allows us to provide safety guarantees. Let $\mathbb{B}(x) = x - 5$ and $k = 2$. One can see that conditions* (4.6) *and* (4.7) *are trivially satisfied. For $x \in \{2, 3, 5\}$, we have $\mathbb{B}(x) \leq 0$ and, hence,* (4.8) *is satisfied. For $x \in \{6, 7\}$, the antecedent of the implication is false which validates* (4.8)*. Similarly, for the state $x = 1$, we have $\mathbb{B}(x) \leq 0$, but since $\mathbb{B}(f(x)) = \mathbb{B}(7) > 0$, the antecedent of the implication is once again false due to which* (4.8) *is true. Thus, the above system is shown to be safe using linear k-inductive barrier certificates according to Definition 20.*

4.3.2 Computation of k-Inductive Barrier Certificates

In this section, we provide suitable computational approaches for synthesizing k-inductive barrier certificates. We propose two different systematic methods for computing k-inductive barrier certificates as presented in Definition 19 and Definition 20, respectively. The first one is based on sum-of-squares (SOS) optimization, while the second one utilizes the δ-complete procedures over the reals [51].

Sum-of-Squares Optimization

For the synthesis of suitable k-inductive barrier certificates based on Definition 19, one can reformulate conditions (4.2)-(4.5) as an SOS optimization problem [93], similar to the approach presented in the previous chapters. This means that when the underlying dynamics of the system $\mathfrak{S}$ is polynomial and the initial set X_0 and unsafe set X_u are semi-algebraic [20], we cast conditions (4.2)-(4.5) as a collection of SOS constraints in order to compute a suitable polynomial k-inductive barrier certificate of a predefined degree. We now state the following assumption, which is similar to Assumptions 3 and 5 presented in the previous chapters:

Assumption 6. *The dt-DS $\mathfrak{S}$ has a continuous state set $X \subseteq \mathbb{R}^n$, and its transition function $f : X \to X$ is a polynomial function of the state x.*

Under Assumption 6, conditions (4.2)-(4.5) can be formulated as a set of SOS constraints, which is given by the following lemma.

Lemma 5. *Suppose Assumption 6 holds for the dt-DS $\mathfrak{S}$ and sets X, X_0 and X_u are semi-algebraic and can be described as vectors of polynomial inequalities: $X = \{x \in \mathbb{R}^n \mid g(x) \geq 0\}$, $X_0 = \{x \in \mathbb{R}^n \mid g_0(x) \geq 0\}$, and $X_u = \{x \in \mathbb{R}^n \mid g_u(x) \geq 0\}$, respectively, where the inequalities are provided element-wise. Suppose there exists a polynomial $\mathbb{B}(x)$, constants $k \in \mathbb{N}_{\geq 1}$, $\epsilon \geq 0$, and $d > k\epsilon$ and sum-of-squares polynomials $\lambda(x), \hat{\lambda}(x), \lambda_0(x)$, and $\lambda_u(x)$ of appropriate dimensions such that the following expressions are sum-of-squares polynomials:*

$$-\mathbb{B}(x) - \lambda_0^T(x)g_0(x), \tag{4.9}$$

$$\mathbb{B}(x) - \lambda_u^T(x)g_u(x) - d, \tag{4.10}$$

$$-\mathbb{B}(f(x)) + \mathbb{B}(x) - \lambda^T(x)g(x) + \epsilon, \tag{4.11}$$

$$-\mathbb{B}(f^k(x)) + \mathbb{B}(x) - \hat{\lambda}^T(x)g(x). \tag{4.12}$$

Then, function $\mathbb{B}(x)$ is a k-inductive barrier certificate as in Definition 19 satisfying conditions (4.2)-(4.5).

The proof of Lemma 5 is similar to those of Lemmas 1 and 4, and is omitted from the thesis.

δ-Complete Decision Procedures over Reals (dReal)

The SOS optimization problem described in the previous subsection requires the set of constraints to be in conjunctive form. In other words, SOS handles optimization problems when the constraints are written as conjunctions (logical AND) of one another. However, k-inductive barrier certificates as defined in Definition 20 require the satisfaction of a logical implication (condition (4.8)), which cannot be checked using the SOS approach. Therefore, in order to synthesize suitable k-inductive barrier certificates as in Definition 20, we reformulate conditions (4.6)-(4.8) as a feasibility expression with an existential and universal quantifier and make use of the satisfiability modulo theory (SMT) solver dReal [51, 53] and the *universal clause pruning* approach used in [69] to handle this quantifier alternation. We now state the following assumption that is required to synthesize k-inductive barrier certificates using this approach.

Assumption 7. *The dt-DS $\mathfrak{S}$ has a compact set $X \subset \mathbb{R}^n$, and the initial and unsafe sets X_0 and X_u, respectively, are bounded semi-algebraic sets.*

Under Assumption 7, one can reformulate conditions (4.6)-(4.8) as a feasibility formula whose satisfaction by an SMT solver returns a suitable parametric k-inductive barrier certificate. We define a parametric k-inductive barrier certificate with unknown coefficients $c_i \in \mathbb{R}$ and basis functions $b_i(x)$ as $\mathbb{B}(c,x) = \sum_{i=1}^{m} c_i b_i(x)$. For a polynomial k-inductive barrier certificate, the basis functions $b_i(x)$ are monomials over x. We encode conditions (4.6)-(4.8) with the help of the following formulae:

$$\psi_{ante}(c,x) = \Big(\bigwedge_{0 \le i < k} \big(\mathbb{B}(c, f^i(x)) + \delta \le 0\big)\Big), \tag{4.13}$$

$$\psi_{unsafe}(c,x) = \mathbb{B}(c,x) - \delta > 0, \tag{4.14}$$

$$\psi_{cons}(c,x) = \mathbb{B}(c, f^k(x)) + \delta \le 0, \tag{4.15}$$

$$\psi_{c1}(c,x) = (x \in X_0 \implies \psi_{ante}(c,x)), \tag{4.16}$$

$$\psi_{c2}(c,x) = (x \in X_u \implies \psi_{unsafe}(c,x)), \tag{4.17}$$

$$\psi_{c3}(c,x) = (\psi_{ante}(c,x) \implies \psi_{cons}(c,x)), \tag{4.18}$$

$$\psi_{cond1}(c) = \forall x \in X(\psi_{c1}(c,x)), \tag{4.19}$$

$$\psi_{cond2}(c) = \forall x \in X(\psi_{c2}(c,x)), \tag{4.20}$$

$$\psi_{cond3}(c) = \forall x \in X(\psi_{c3}(c,x)), \tag{4.21}$$

$$\psi_{bar}(c) = (\psi_{cond1}(c) \wedge \psi_{cond2}(c) \wedge \psi_{cond3}(c)), \tag{4.22}$$

where $\delta \in \mathbb{R}_{>0}$ is a tolerance parameter to ensure the satisfaction of conditions (4.6)-(4.8) via δ-complete decision procedures. Then, a function $\mathbb{B}(x)$ is a k-inductive barrier certificate if the formula $\phi = \exists c \psi_{bar}(c)$ is satisfiable. In other words, there must exist coefficients $c_i \in \mathbb{R}, i \in \{1, \ldots, m\}$, of the k-inductive barrier certificate such that the formula $\psi_{bar}(c)$ holds over the bounded state set. We note that ϕ is a formula with one quantifier alternation where a universal quantifier over state variables x follows the existential quantifier over coefficients c, and the state variables are in a bounded domain. To ensure all variables are in a bounded domain, we also bound the coefficients c to lie in a fixed interval. To determine the satisfiability of ϕ, we make use of the branch-and-prune algorithm of dReal [53] in conjunction with the universal clause pruning approach presented in [69] to handle the quantifier alternation. dReal handles bounded logical formulae over nonlinear functions using Interval Constraint Propagation (ICP) [18] as the underlying theory solver. Given a formula ϕ, the algorithm either returns δ-sat if there exist intervals of size at most δ such that ϕ is satisfiable in those intervals, or it returns unsat indicating ϕ is unsatisfiable. Since all the variables are bounded, they can instead be considered as intervals, *i.e.*, each variable takes any value in its associated interval. We define a box B to be a product of these intervals. We use B_c to specify the intervals of the box B that correspond to the coefficient variables c, and similarly, B_x specifies the intervals that correspond to state variables x. ICP takes the box B_c and the formula ϕ as input and computes those points in the interval that do not satisfy the constraint via a fixed point algorithm. It then *prunes* the box by removing these points from the interval. If the maximum width of the box is larger than δ, it then *branches* by

selecting a coefficient variable c_i and divides its associated interval I_{c_i} into two halves, resulting in two new boxes. These boxes are pushed onto a stack S, and the algorithm then iterates over the boxes present in S, by pruning and branching until S is empty or the size of a box is small enough, *i.e.*, within δ. For the sake of completeness, the *branch-and-prune* approach described above is presented in Algorithm 1.

Now, we utilize the universal clause pruning algorithm to handle the universal quantifier over state variables x. To do this, we rewrite (4.21) as

$$\psi_{cond3}(c) = \forall x \in X\Big(\bigvee_{0\leq i\leq k} h_i(c,x) > 0\Big),$$

where $h_i(c,x) = \mathbb{B}(c, f^i(x)) + \delta$ for all $0 \leq i < k$ and $h_k(c,x) = -\mathbb{B}(c, f^k(x)) - \delta$. We then prune on this constraint using Algorithm 2. We consider $\delta', \varepsilon \in \mathbb{R}$ such that $0 < \delta' < \varepsilon < \delta$. The calls to the Prune and Solve function are for the universal quantifier-free formulae which rely on the techniques in [53] for pruning and solving. We repeat the same procedure for conditions (4.19) and (4.20) and note that the universal quantifier commutes over conjunction.

Algorithm 1 Algorithm for ICP adapted from [51] for barrier certificates

```
function ICP(B_c, ψ_cond1(c), ψ_cond2(c), ψ_cond3(c), S)
    S.push(B_c)
    while S ≠ ∅ do
        B ← S.pop()
        B^1 ← PruneB(B, B_x, ψ_cond1(c), δ', ε, δ)
        B^2 ← PruneB(B^1, B_x, ψ_cond2(c), δ', ε, δ)
        B^3 ← PruneB(B^2, B_x, ψ_cond3(c), δ', ε, δ)
        if B^3 ≠ ∅ then
            if |I_{c_i}| ≥ δ then
                {B', B''} ← Branch(B, i)
                S.push(B', B'')
            else
                return sat
    return unsat
```

Remark 23. *The proposed approach above can also be used to find suitable k-inductive barrier certificates as in Definition 19. Also, the use of SMT solvers allows one here to propose a more sophisticated form of Definition 19 where instead of requiring that for all states $x \in X$ every sequence of k-transitions result in a net decrease in the value of the barrier certificate, one may require it only for those states with the value of the barrier certificate bounded from above by d.*

4.3.3 Case Study

For our case study, we consider the discrete-time model of a source-free series RLC circuit with state variables i, v denoting the inductor current and capacitor voltage. The dynamics can be given by the following difference equations:

Algorithm 2 Algorithm for universal clause pruning adapted from [69] for $\psi_{cond3}(c)$

function PRUNEB($B_c, B_x, \psi_{cond3}(c), \delta', \varepsilon, \delta$)
  repeat
    $B_c^{prev} \leftarrow B_c$
    $\psi_{count} \leftarrow \bigwedge_{0 \le i \le k} (h_i(c,x) < -\varepsilon)$
    $x_{count} \leftarrow Solve(\psi_{count}, \delta')$
    if $x_{count} = \varnothing$ **then**
      return B_c
    for $i \in \{0, \dots k\}$ **do**
      $B^i \leftarrow B_c \cap \text{Prune}(B_c, h_i(c, x_{count}) \ge 0)$
    $B_c \leftarrow \bigsqcup_{i=0}^{k} B^i$
  until
  $B_c \neq B_c^{prev}$
  return B_c

$$\mathfrak{S} : \begin{cases} i(t+1) = i(t) + \tau_s(-\frac{R}{L}i(t) + -\frac{1}{L}v(t)), \\ v(t+1) = v(t) + \tau_s \frac{1}{C} i(t), \end{cases} \tag{4.23}$$

where $\tau_s = 0.5s$ is the sampling time, $R = 3\Omega$ is the series resistance, $L = 8$H is the series inductance, and $C = 0.5$F is the capacitance of the circuit. Let the state set be $X = [-1, 5] \times [-4, 4]$. The initial set and the unsafe set are given by $X_0 = [0, 0.5] \times [0, 1]$ and $X_u = [1, 5] \times [-4, 4]$, respectively. We consider a function of the parametric form $\mathbb{B}(i, v) = c_1 i^2 + c_2 v^2 + c_3$ and attempt to compute suitable coefficients $c_1, c_2, c_3 \in \mathbb{R}$ such that $\mathbb{B}(i, v)$ is a standard barrier certificate as defined in Definition 5. To do so, we utilize SOSTools [98] in conjunction with SeDuMi [119] on MATLAB to reformulate conditions (2.8)-(2.10) as an SOS optimization problem. However, we see that there exist no values of c_1, c_2, c_3 such that $\mathbb{B}(i, v)$ conditions (2.8)-(2.10) are satisfied. Therefore, using the standard barrier certificate approach, one cannot verify the safety of dt-DS $\mathfrak{S}$.

Using Sum-of-Squares Optimization

We now compute coefficients c_1, c_2, c_3 such that $\mathbb{B}(i, v)$ is a k-inductive barrier certificate as in Definition 19. Once again, we utilize SOSTools and SeDuMi to reformulate conditions (4.2)-(4.5) as an SOS problem via Lemma 5. By considering $k = 6$, $\epsilon = 0.06$ and $d = 0.361$, we obtain $\mathbb{B}(i, v) = 0.7127i^2 + 0.04319v^2 - 0.2957$ as the k-inductive barrier certificate as described in Definition 19. Therefore, by using Theorem 8, one can conclude that system $\mathfrak{S}$ indeed satisfies the safety objective with respect to the initial set X_0 and unsafe set X_u. We shall mention that the computation time for this approach using the mentioned tools is about 20 seconds on a machine running with Linux Ubuntu OS (Intel i7-8665U CPU with 32 GB of RAM).

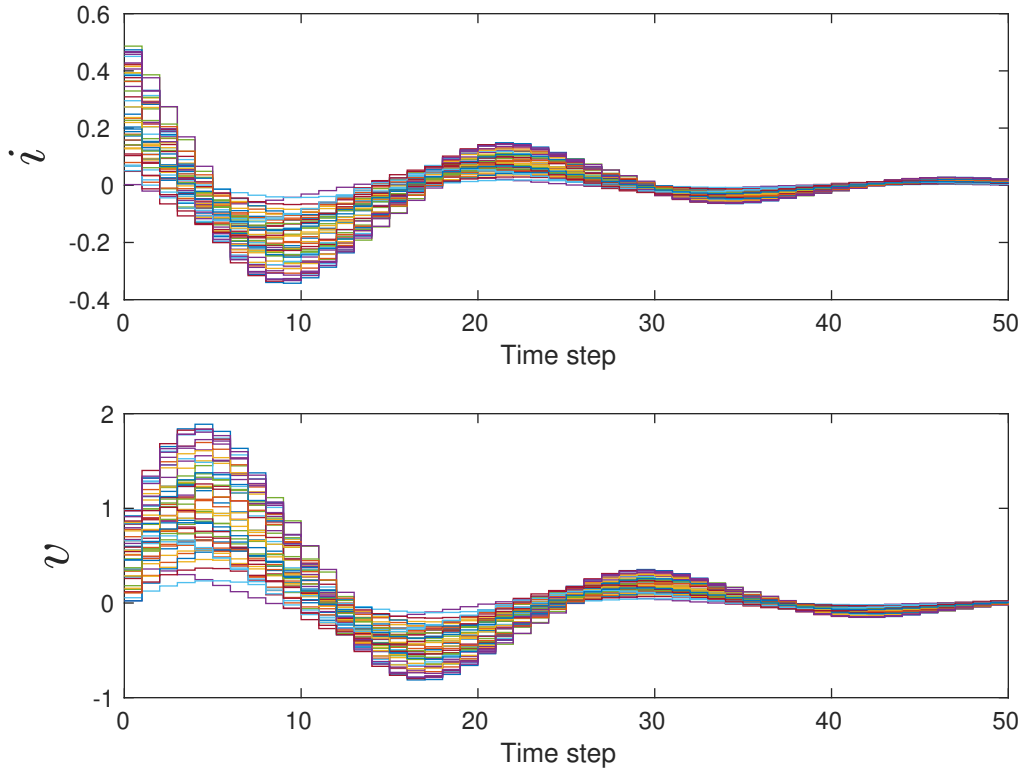


Figure 4.5: State sequences with respect to current i and voltage v starting from different initial states inside X_0.

Using δ-complete Decision Procedures over Reals

We now demonstrate the computation of k-inductive barrier certificates as in Definition 20 for the same dynamics and regions of interest considered above. We formulate a parametric k-inductive barrier certificate $\mathbb{B}(i, v) = c_1 i^2 + c_2 v^2 + c_3$ and compute the coefficients $c_1, c_2, c_3 \in \mathbb{R}$ such that $\mathbb{B}(i, v)$ is a k-inductive barrier certificate as defined in Definition 20 by utilizing dReal in conjunction with universal clause pruning. To do so, we first bound the coefficients $c_1, c_2 \in [0.1, 6]$ and $c_3 \in [-6, 6]$. We consider $\delta + 0.2$ as the constant in equation (4.14) to ensure the k-inductive barrier certificate is strictly positive in the unsafe region. Similarly, we consider $\delta + 0.1$ instead of δ in equations (4.13) and (4.15) to ensure that the k-inductive barrier certificate is strictly negative in the safe regions. By setting $k = 3$ and $\delta = 0.05$, we obtain $\mathbb{B}(i, v) = 1.403317i^2 + 0.104829v^2 - 1.128599$ as the k-inductive barrier certificate as in Definition 20. By utilizing Theorem 9, one can conclude that the system $\mathfrak{S}$ satisfies the safety objective with respect to the initial set X_0 and unsafe set X_u. The computation time is around 15 seconds on a machine running MacOS 11.2 (Intel i9-9980HK with 64 GB of RAM).

Figure 4.5 (©2021 IEEE) shows the state sequences of the system starting from different initial conditions inside X_0. As it can be observed, the state sequences always stay away from the

unsafe region X_u.

4.4 Safety Verification of Stochastic Dynamical Systems

In this section, we consider discrete-time stochastic dynamical systems (dt-SS) defined in Definition 2 in the absence of control inputs:

$$\mathfrak{S} : \mathbf{x}(t+1) = f(x(t), \varsigma(t)), \tag{4.24}$$

where, similar to Definition 2, $\mathbf{x} : \Omega \times \mathbb{N} \to X$ is the solution process of the system, and $\mathbf{x}_{x_0}$ refers to the solution process starting from the initial condition $\mathbf{x}_{x_0}(0) = x_0$. Our aim is to provide a probability upper bound for the satisfaction of safety specifications. In other words, given an initial set of states X_0, and unsafe set X_u, we would like to solve Problem 3.

To do this, one can utilize barrier certificates as defined in Definition 7, which requires the barrier certificate to be a *supermartingale* function, such that the conditional expectation at the next value of the barrier certificate is smaller than the present value irrespective of the prior values. This property is essential in providing probability guarantees for the satisfaction of safety properties via Theorem 1. As has been already described in the previous chapters, the search for a suitable barrier certificate satisfying conditions (2.14)-(2.16) may be performed by restricting the function space to a certain parametric form and then utilizing SOS optimization or SMT solvers. However, the supermartingale condition imposed on barrier certificates via Definition 7 can be quite restrictive. Due to this, one may have to replace the probability $1 - \varepsilon$ in equation (2.17) with a trivial value of 0, and as a result, the approach fails to give a non-trivial probability of safety. We now utilize the following example to show instances where the barrier certificate approach fails to provide non-trivial probabilistic guarantees even when the system is safe with a high probability when considering a fixed parametric form for the barrier certificates.

Example 3. *Consider a Markov chain shown in Figure 4.6 as a finite state stochastic system* $\mathfrak{S}$ *with* $x \in X = \{0, 0.1, 0.2, 0.3, 0.5, 6, 10\}$ *as states of the system,* $x = 0.2$ *as the initial state and* $x = 10$ *as the unsafe state. By utilizing barrier certificates, we want to provide a tight lower bound on the probability that the solution processes do not reach unsafe regions. As it can be seen from the figure, the probability with which the system remains safe is* 0.99*. However, by choosing a linear barrier certificate according to Definition 7, we cannot provide a non-trivial probabilistic lower bound on the satisfaction of safety.*

Consider $\mathbb{B}(x) = ax + b$*. According to condition* (2.14)*, since* $x = 0.2$ *is the initial state, we have* $0.2a + b \leq \varepsilon$*, for some* $0 \leq \varepsilon \leq 1$*. Moreover, by applying the supermartingale condition* (2.16) *at* $x = 0.2$*, we get* $\mathbb{E}[\mathbb{B}(f(x)) \mid x = 0.2] - \mathbb{B}(x) = 0.2a \leq 0$*, implying that* $a \leq 0$*. However, due to condition* (2.15) *and the fact that* $x = 10$ *is the unsafe state, we have* $10a + b \geq 1$*. Then,* $a \leq 0$ *would lead to contradiction between conditions conditions* (2.14) *and* (2.15)*. Therefore, there does not exist a linear barrier certificate for any value of* ε*.*

A practical approach to tackle this issue is to relax condition (2.16) by utilizing a *c-martingale* instead of a supermartingale (see Corollary 1). In this case, the expected value of the barrier

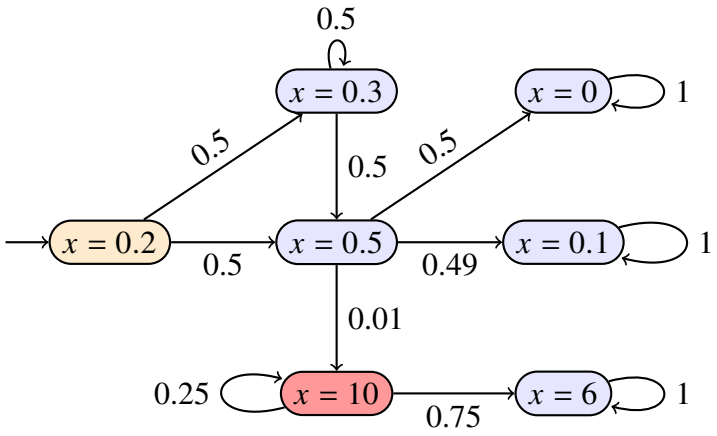


Figure 4.6: Finite Markov chain $\mathfrak{S}$ for Example 3. The initial state is denoted in yellow and the unsafe state is in red.

certificate is not required to be non-increasing at every time step. Instead, the expected value can increase at every time step as long as it is bounded by a constant c. This condition then allows the barrier certificate to increase slowly in expectation such that it takes a *long* time to reach the unsafe regions of the state set. It is apparent from equation (2.22) that the probability of satisfaction depends on the time horizon T. Due to this dependency, one may only obtain a high probability of satisfaction for short time horizons. However, for reactive systems, such as medical devices and power grids, it is vital to provide long-term or even unbounded-time safety guarantees. Therefore, it becomes necessary to be able to relax the supermartingale requirement of barrier certificates while still providing probabilistic safety guarantees over infinite time horizons. Note that the standard notion of barrier certificates as in Definition 7 are analogous to standard inductive proofs discussed in 4.2. The definition of barrier certificates is similar to inductive proofs that yield expectation invariants [24]. Particularly, via conditions (2.14) and (2.16), the expectation of barrier certificate at any time instant is bounded by ε. This allows one to view condition (2.14) as the base case, while the supermartingale condition (2.16) is the inductive step.

We show that we can effectively weaken the supermartingale conditions for safety by leveraging the k-induction principle, which results in less conservative conditions for barrier certificates that are easier to satisfy. These barrier certificates, which we dub as *k-inductive barrier certificates*, can still provide probabilistic guarantees for the satisfaction of safety over infinite time horizons. Therefore, a dt-SS $\mathfrak{S}$ that does not admit the standard notion of barrier certificates for safety may admit a k-inductive barrier certificate, while still providing infinite time horizon guarantees.

4.4.1 k-Inductive Barrier Certificates for Probabilistic Safety

This section presents the main results concerning probabilistic safety verification via k-inductive barrier certificates. Our approach relies on looking at the behavior of the dt-SS in future time instances, such as after i time steps rather than at every time step. We obtain such behavior by

simply utilizing the recursive application of the function f defined in (4.24). In particular, the value of the solution process after the i^{th} time step, $i \geq 1$ is obtained as

$$\mathbf{x}(t+i) = f_i(\mathbf{x}(t), \varsigma_i(t)), \tag{4.25}$$

where $\varsigma_i(t) = [\varsigma(t); \ldots; \varsigma(t+i-1)]$ is the vector containing all the noise terms from time t to time $t+i-1$, and we define f_i recursively, where $f_i(\mathbf{x}(t), \varsigma_i(t)) = f(\mathbf{x}(t), \varsigma(t))$, if $i = 1$, and $f_i(\mathbf{x}(t), \varsigma_i(t)) = f(f_{i-1}(\mathbf{x}(t), \varsigma_{i-1}(t)), \varsigma(t+i-1))$ for all $i > 1$.

To provide probabilistic safety guarantees over infinite time horizons, one can simply extend the notion of c-martingale barrier certificates and leverage the k-induction principle. We define k-inductive barrier certificates for safety as follows.

Definition 21. *Consider a dt-SS $\mathfrak{S}$. We say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a k-inductive barrier certificate for $\mathfrak{S}$ with respect to a set of initial states X_0 and an unsafe set X_u if there exist constants $k \in \mathbb{N}_{\geq 1}$, $0 \leq \varepsilon \leq 1$ and $c \geq 0$ such that the following holds:*

$$\mathbb{B}(x) \leq \varepsilon, \qquad \text{for all } x \in X_0, \tag{4.26}$$
$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in X_u, \tag{4.27}$$
$$\mathbb{E}[\mathbb{B}(f(x, \varsigma)) \mid x] - \mathbb{B}(x) \leq c, \qquad \text{for all } x \in X, \tag{4.28}$$
$$\mathbb{E}[\mathbb{B}(f_k(x, \varsigma_k)) \mid x] - \mathbb{B}(x) \leq 0, \qquad \text{for all } x \in X. \tag{4.29}$$

Note that condition (4.28) requires the barrier certificate to be a c-martingale at every time step and condition (4.29) requires the barrier certificate sampled after every k^{th} step to be a supermartingale. We now present the key result for safety verification based on this definition of k-inductive barrier certificates.

Theorem 10. *Consider a dt-SS $\mathfrak{S}$. Let $\mathbb{B}$ be a barrier certificate for $\mathfrak{S}$ satisfying conditions* (4.26)-(4.29) *with some $0 \leq \varepsilon \leq 1$, $c \geq 0$, and $k \in \mathbb{N}_{\geq 1}$. Then the probability that the solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X_0$ does not reach the unsafe region X_u is bounded by*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin X_u \text{ for all } t \in \mathbb{N} \mid x_0\} \geq 1 - k\varepsilon - \frac{k(k-1)c}{2}. \tag{4.30}$$

Proof. According to condition (4.27), $X_u \subseteq \{x \in X \mid \mathbb{B}(x) \geq 1\}$. Therefore, it follows that

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_u \text{ for some } t \in \mathbb{N} \mid x_0\} \leq \mathbb{P}\{\sup_{t \in \mathbb{N}} \mathbb{B}(\mathbf{x}_{x_0}(t)) \geq 1 \mid x_0\}. \tag{4.31}$$

Now, for the dt-SS $\mathfrak{S}$, consider k systems sampled after every k steps, each starting from initial conditions $x_0, \mathbf{x}_{x_0}(1), \ldots, \mathbf{x}_{x_0}(k-1)$, respectively. The dynamics of these systems are obtained as

$$\begin{aligned}
&\mathbf{x}_{x_0}(t+k) = f_k(\mathbf{x}_{x_0}(t), \varsigma_k(t)),\\
&\mathbf{x}_{x_0}(t+k+1) = f_k(\mathbf{x}_{x_0}(t+1), \varsigma_k(t+1)),\\
&\vdots\\
&\mathbf{x}_{x_0}(t+2k-1) = f_k(\mathbf{x}_{x_0}(t+k-1), \varsigma_k(t+k-1)).
\end{aligned}$$

Due to condition (4.29), the barrier certificate $\mathbb{B}$ satisfies the supermartingale condition (2.16) for each of these systems. Now, by means of Boole's inequality and Theorem 1, we obtain

$$\begin{aligned}\mathbb{P}\{\sup_{t\in\mathbb{N}} \mathbb{B}(\mathbf{x}_{x_0}(t)) \geq 1 \mid x_0\} &\leq \sum_{i=0}^{k-1} \mathbb{P}\{\sup_{t=jk, j\in\mathbb{N}} \mathbb{B}(\mathbf{x}_{x_0}(i+t)) \geq 1 \mid \mathbf{x}_{x_0}(i)\} \\ &\leq \sum_{i=0}^{k-1} \mathbb{E}(\mathbb{B}(\mathbf{x}_{x_0}(i)).\end{aligned}$$

Now, from condition (4.26), we have that $\mathbb{E}(\mathbb{B}(\mathbf{x}_{x_0}(0))) = B(x_0) \leq \varepsilon$. Moreover, by applying law of total expectation and condition (4.28) recursively for each term in the right-hand side of the above inequality, we get

$$\begin{aligned}\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_u \text{ for some } t \in \mathbb{N} \mid x_0\} &\leq \varepsilon + \sum_{i=1}^{k-1} (\varepsilon + ic) \\ &= k\varepsilon + \frac{k(k-1)c}{2}.\end{aligned}$$

By complementing the above, we obtain the lower bound (4.30) on the probability such that the system remains in the safe regions. □

Remark 24. *Note that, in order to obtain meaningful probabilities, the value of k in inequality* (4.30) *is bounded by*

$$1 \leq k \leq \frac{(c-2\varepsilon) + \sqrt{4\varepsilon^2 + c^2 - 4c(2+\varepsilon)}}{2c}.$$

One can readily observe that k-inductive barrier certificates as in Definition 21 also lead to expectation invariants, as the expected value of the barrier certificate remains bounded in the set $\mathbb{E}[\mathbb{B}(\mathbf{x}_{x_0}(t)) \mid x_0] \leq \varepsilon + (k-1)c$ for all $t \in \mathbb{N}$ due to the bounded increase of $\mathbb{E}[\mathbb{B}(\mathbf{x}_{x_0}(t)) \mid x_0]$ at every time step via conditions (4.28) and (4.29). Note that, when $k = 1$ and $c = 0$, conditions (4.26)-(4.29) reduce to standard barrier certificate conditions (2.14)-(2.16). Moreover, one immediately observes that the probability bounds in (4.30) also converge to those in (2.17) under the same conditions. Therefore, any barrier certificate satisfying conditions (2.14)-(2.16) is also a 1-inductive barrier certificate as in Definition 7. However, the converse may not hold true since conditions (4.26)-(4.29) are more relaxed. We now illustrate k-inductive barrier certificates as in Definition 21 with the Markov chain considered in Example 3.

Example 3 (Continued)**.** *Let us consider the finite Markov chain $\mathfrak{S}$ presented in Figure 4.6. For this system, we already showed that there exists no linear barrier certificate satisfying conditions* (2.14)-(2.16) *for any $0 \leq \varepsilon < 1$ which leads to trivial probabilistic bounds for the satisfaction of safety. Now, we show that by using k-inductive barrier certificates as in Definition 21, we get more reliable probabilistic bounds for the satisfaction of safety.*

Consider $\mathbb{B}(x) = 0.1x + 0.01$, constants $\varepsilon = 0.05$ and $c = 0.02$, and $k = 3$. The enumerated values of $\mathbb{B}(x), \mathbb{E}[\mathbb{B}(f(x,\varsigma) \mid x]$, $\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2)) \mid x]$, and $\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x]$ for all states

x	$\mathbb{B}(x)$	$\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x]$	$\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2)) \mid x]$	$\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x]$
0.2	0.03	0.05	0.03745	0.029675
0.3	0.04	0.05	0.03745	0.029675
0.5	0.06	0.0249	0.0219	0.02115
0	0.01	0.01	0.01	0.01
0.1	0.02	0.02	0.02	0.02
6	0.61	0.61	0.61	0.61
10	1.01	0.71	0.635	0.61625

Table 4.1: The values of $\mathbb{E}[\mathbb{B}(f_i(x,\varsigma_i)) \mid x]$ for all $i \in \{1,2,3\}$ and all $x \in X$ for Example 3. Note that $\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x] - \mathbb{B}(x) \leq 0$ for all $x \in X$.

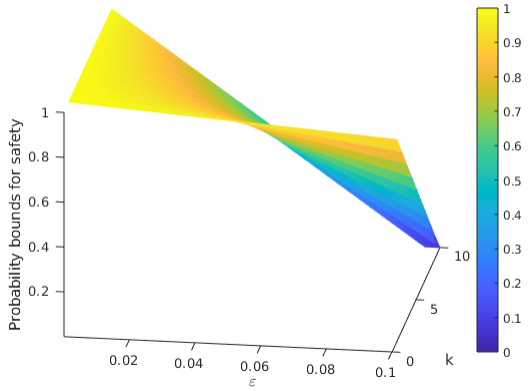


Figure 4.7: Variation of probability bounds for safety with respect to k and ε values.

$x \in X$ are provided in Table 4.1. We immediately see that condition (4.26) *is satisfied for the initial state $x = 0.2$. Similarly, condition* (4.27) *holds for the unsafe state $x = 10$. Moreover, conditions* (4.28) *and* (4.29) *also hold for all $x \in X$. Therefore, $\mathbb{B}(x) = 0.1x + 0.01$ is indeed a linear 3-inductive barrier certificate for $\mathfrak{S}$. We now apply Theorem 10 to obtain a lower bound on the probability of safety as $\mathbb{P}\{x_{x_0}(t) \notin X_u = \{10\}$ for all $t \in \mathbb{N} \mid x_0 = \{0.2\}\} \geq 0.79$.*

Figure 4.7 shows how the probability bounds for safety in (4.30) is affected for different values of $k \leq 10$ and $\varepsilon \leq 0.1$, for a fixed value of c for $k > 1$ (for $k = 0$, we have $c = 0$). Ideally, to obtain a high probability bound for safety, one requires $k = 1$ and ε to be as small as possible. However, due to the restrictive nature of barrier certificate conditions when $k = 1$, the minimal obtained value of ε, even if exists, may be high. In such cases, by considering $k > 1$, one is still able to relax the barrier certificate conditions, allowing to further reduce the value of ε such that a higher and a more reliable, less conservative probability is obtained.

4.4.2 Computation of k-Inductive Barrier Certificates

We now complete this section by providing an SOS-based optimization problem for the computation of k-inductive barrier certificates as in Definition 21. To do so, we once again require the

assumption on the underlying dynamics of the dt-SS $\mathfrak{S}$, as well as the sets X_0, X_u and X.

Assumption 8. *The dt-SS $\mathfrak{S}$ has a continuous state set X and the function $f : X \times V_\varsigma \to X$ is polynomial in the state variable x and noise variable ς. Moreover, the sets X, X_0, X_u and X_R are semi-algebraic.*

Then, we formulate the following lemma to compute suitable k-inductive barrier certificates for safety verification.

Lemma 6. *Consider a dt-SS $\mathfrak{S}$. Suppose Assumption 8 holds and there exists a sum-of-squares polynomial $\mathbb{B}(x)$, constants $k \in \mathbb{N}_{\geq 1}$, $0 \leq \varepsilon \leq 1$ and $c \geq 0$, and vectors of sum-of-squares polynomials $\lambda(x), \hat{\lambda}(x), \lambda_0(x)$, and $\lambda_u(x)$ of appropriate dimensions such that the following expressions are sum-of-squares polynomials:*

$$-\mathbb{B}(x) - \lambda_0^T(x) g_0(x) + \varepsilon, \tag{4.32}$$

$$\mathbb{B}(x) - \lambda_u^T(x) g_u(x) - 1, \tag{4.33}$$

$$-\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x] + \mathbb{B}(x) - \lambda^T(x) g(x) + c, \tag{4.34}$$

$$-\mathbb{E}[\mathbb{B}(f_k(x,\varsigma_k)) \mid x] + \mathbb{B}(x) - \hat{\lambda}^T(x) g(x). \tag{4.35}$$

Then the function $\mathbb{B}(x)$ is a k-inductive barrier certificate as in Definition 21 satisfying conditions (4.26)-(4.29).

4.4.3 Case Study

In this case study, we study the safety property of a series RLC circuit. The dynamics of the dt-SS $\mathfrak{S}$ are given as

$$\mathfrak{S} : \begin{cases} i(t+1) = i(t) + \tau_s(-\frac{R}{L} i(t) + -\frac{1}{L} v(t)) + G\varsigma(t), \\ v(t+1) = v(t) + \tau_s \frac{1}{C} i(t), \end{cases} \tag{4.36}$$

where $i(t)$ denotes the current at time t, $v(t)$ is the voltage, $\tau_s = 0.5s$ is the sampling time, $R = 2\Omega$ is the series resistance, $L = 9\text{H}$ is the series inductance, $C = 0.5\text{F}$ is the capacitance of the circuit, and $G = 0.004$ is the noise coefficient. The state space of the system is given as $X = [-2, 2] \times [-4, 4]$, where the initial set $X_0 = [0, 0.5] \times [0, 1]$ and the unsafe set $X_u = [1, 2] \times [-4, 4]$.

We aim to utilize barrier certificates for safety as in Definition 7 to find the probability bound with which $\mathfrak{S}$ satisfies the safety property. To do so, we first consider the barrier certificate to be a polynomial of degree 6, and use the SOS programming toolbox YALMIP [78] version R20200930 along with SeDuMi [119] version 1.3 on MATLAB R2019b to search for a suitable barrier certificate satisfying conditions (2.14)-(2.16). However, we fail to find a supermartingale that achieves any meaningful probability of satisfaction.

We now compute a suitable polynomial k-inductive barrier certificate of degree 6 as in Definition 21 by reformulating conditions (4.26)-(4.29) as an SOS problem via Lemma 6. By

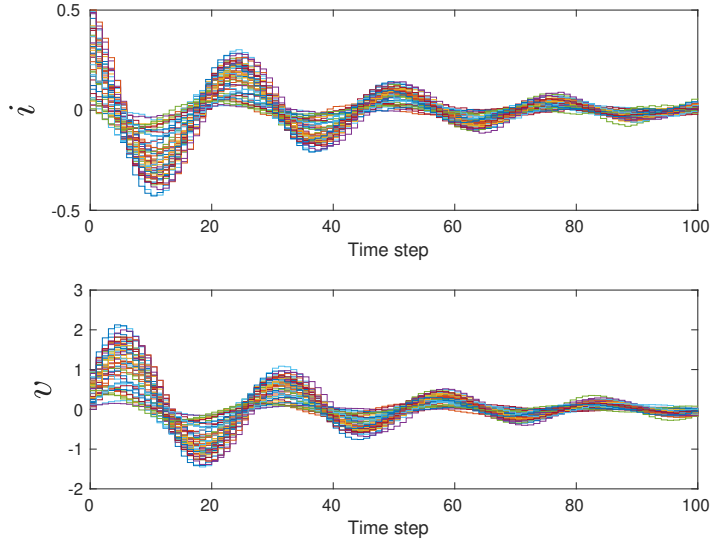


Figure 4.8: Solution processes of $\mathfrak{S}$ with respect to current i and voltage v from different initial states.

considering $k = 2$, $\varepsilon = 0.029$ and $c = 10^{-4}$, we get a barrier certificate of degree 6 satisfying conditions (4.26)-(4.29). By Theorem 10, we can infer that the system $\mathfrak{S}$ satisfies the safety specification with a probability of at least 0.9419 for an infinite time horizon. In comparison, by utilizing c-martingale barrier certificates via Corollary 1 for the same value of ε and c, one would obtain the probability of 0.9419 for a bounded time of 4564.5 seconds. Figures 4.8 shows the current and voltage for 50 representative solution processes starting from different initial conditions inside X_0. The computation time for this approach using the tools above is about 40 seconds on a machine running with Linux Ubuntu OS (Intel i7 – 8665U CPU with 32GB of RAM).

4.5 Reachability Verification of Stochastic Dynamical Systems

In this section, we are interested in verifying reachability properties for stochastic dynamical systems with dynamics (4.24), considered in the previous section. Particularly, we would like to provide probabilistic guarantees for the solution processes $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ starting from the initial set X_0 to reach some desired set of states X_R. In other words, we aim to solve Problem 6.

Similar to the safety verification procedure, one may also utilize barrier certificates to provide probabilistic guarantees over reachability specifications. This can be done via Definition 10 or via Definition 11, under the satisfaction of Assumption 1. While Definition 11 provides a mechanism to obtain a lower bound on the probability of satisfaction reach-and-avoid like specification, Definition 11 enables us to obtain *almost-sure* guarantees for reachability, where the probability of satisfaction is 1. These barrier certificates may be computed by restricting them to a certain parametric form and utilizing computational techniques like SOS or SMT solvers,

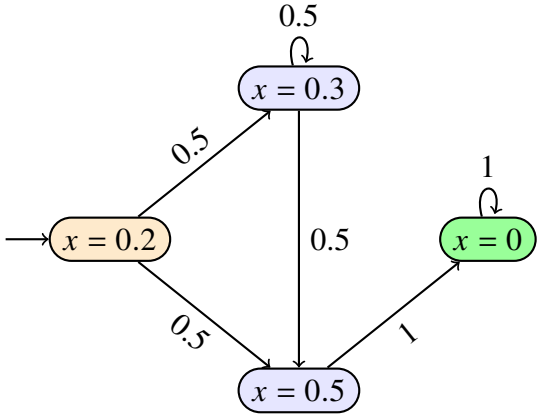


Figure 4.9: Finite Markov chain $\mathfrak{S}'$ for Example 4. The initial state is denoted in yellow and the target state in green.

similar to the case of safety. However, computing barrier certificates can be hard due to the restrictiveness of the barrier certificate conditions. This is due to the fact that conditions (2.29) and (2.33) require the satisfaction of a strict supermartingale condition, due to which their values need to be strictly decreasing at every time step. In the following example, we demonstrate that the standard barrier certificate approach presented in Chapter 2 fails to provide probabilistic reachability guarantees for a fixed template of barrier certificates even when the system satisfies the reachability specification almost surely.

Example 4. *Consider a Markov chain shown in Figure 4.9 as a finite state stochastic system* $\mathfrak{S}'$ *with* $x \in X = \{0, 0.2, 0.3, 0.5\}$ *as states of the system,* $x = 0.2$ *as the initial state and* $x = 0$ *as the target state. It can be immediately seen that the solution processes of* $\mathfrak{S}'$ *reach the target state with probability* 1*. However, we want to provide a barrier certificate of a fixed template to guarantee the satisfaction of the reachability specification with a non-trivial probability bound via Definition 10.*

Consider a linear barrier certificate $\mathbb{B}(x) = ax + b$*. Note that in the context of finite systems, if there are no states to avoid, we do not need to ensure condition* (2.28) *for any state in* X*. Now, according to condition* (2.27)*, since* $x = 0.2$ *is the initial state, we get* $0.2a+b \leq \varepsilon$*. By applying the supermartingale condition* (2.29) *at* $x = 0.2$*, we get* $\mathbb{E}[\mathbb{B}(f(x)) \mid x = 0.2] - \mathbb{B}(x) = 0.2a < -\delta$*, implying that* $a < 0$*. Similarly, applying condition* (2.29) *at* $x = 0.5$*, we get* $\mathbb{E}[\mathbb{B}(f(x)) \mid x = 0.5] - \mathbb{B}(x) = -0.5a < -\delta$ *implying that* $a > 0$*, which results in a contradiction. Therefore there exists no linear barrier certificate satisfying conditions* (2.27)-(2.29) *for any value of* ε *and we cannot give a non-trivial probability of reachability with a linear barrier certificate as in Definition 10.*

Similarly, consider a linear barrier certificate as in Definition 11. Note that condition (2.33) *is the same as condition* (2.29) *for finite systems. So it also follows that there exists no linear barrier certificate satisfying condition* (2.33) *and we cannot ensure reachability with a linear barrier certificate as in Definition 11 as well.*

The barrier certificates for reachability via Definitions 10 and 11 are also analogous to standard induction where conditions (2.27) and (2.29) as well as conditions (2.31) and (2.33) act as base

cases and inductive steps for Definitions 10 and 11, respectively. Now, we consider k-inductive barrier certificates for reachability and show that they still provide unbounded-time guarantees while also relaxing the standard barrier certificate conditions, so that a larger class of functions may act as barrier certificates.

4.5.1 k-Inductive Barrier Certificates for Probabilistic Reachability

In this section, we leverage k-inductive barrier certificates to obtain probabilistic guarantees for reachability specifications over infinite time horizons. To do this, one can relax the supermartingale condition imposed at every time step to a supermartingale requirement after k time steps while necessitating a c-martingale condition at every time step. We first consider k-inductive barrier certificates based on Definition 10, provided under the satisfaction of Assumption 1, which requires the dt-SS $\mathfrak{S}$ to be forward invariant in the state set X.

Definition 22. *Consider a dt-SS $\mathfrak{S}$ that satisfies Assumption 1. We say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a k-inductive barrier certificate for dt-SS $\mathfrak{S}$ with respect to a set of initial states X_0 and a set of target states X_R if there exists constants $k \in \mathbb{N}_{\geq 1}$, $0 \leq \varepsilon \leq 1$, $c \geq 0$ and $\delta > 0$ such that the following conditions hold:*

$$\mathbb{B}(x) \leq \varepsilon, \qquad \text{for all } x \in X_0, \tag{4.37}$$

$$\mathbb{B}(x) \geq 1, \qquad \text{for all } x \in \partial X \setminus \partial X_R, \tag{4.38}$$

$$\mathbb{E}[\mathbb{B}(f(x, \varsigma)) \mid x] - \mathbb{B}(x) \leq c, \qquad \text{for all } x \in \overline{X \setminus X_R}, \tag{4.39}$$

$$\mathbb{E}[\mathbb{B}(f_k(x, \varsigma_k)) \mid x] - \mathbb{B}(x) \leq -\delta, \qquad \text{for all } x \in \overline{X \setminus X_R}. \tag{4.40}$$

Note that condition (4.39) requires the barrier certificate to be a c-martingale at every time step and condition (4.40) requires the barrier certificate sampled after every k^{th} step to be decreasing in expectation for all states not in the set of target states.

Now, we present the first main result of this section and provide probabilistic guarantees for reachability specifications via the above definition of k-inductive barrier certificates.

Theorem 11. *Consider a dt-SS $\mathfrak{S}$ with dynamics* (4.24) *satisfying Assumption 1. Let $\mathbb{B}$ be a barrier certificate for $\mathfrak{S}$ satisfying conditions* (4.37)-(4.40) *with some $0 \leq \varepsilon \leq 1$, $c \geq 0$, $\delta > 0$ and $k \in \mathbb{N}_{\geq 1}$. Then the probability the the solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X_0$ reaches the target region X_R is bounded by*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_R \text{ for some } t \in \mathbb{N} \mid x_0\} \geq 1 - k\varepsilon - \frac{k(k-1)c}{2}. \tag{4.41}$$

Proof. The proof for this theorem can be obtained by utilizing Theorem 10 from Section 4.4 and Theorem 3. From Theorem 10 with $X_u = \partial X \setminus \partial X_R$, one has the probability that a solution process $\mathbf{x}_{x_0}$ of $\mathfrak{S}$ starting from $x_0 \in X$ does not enter the boundary set $\partial X \setminus \partial X_R$ is

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \notin \partial X \setminus \partial X_R \text{ for all } t \in \mathbb{N} \mid x_0\} \geq 1 - k\varepsilon - \frac{k(k-1)c}{2}. \tag{4.42}$$

Now, for $\mathfrak{S}$, consider k-systems sampled after every k steps, each starting from initial conditions $x_0, \mathbf{x}_{x_0}(1), \ldots, \mathbf{x}_{x_0}(k-1)$. From condition (4.40), we have that each of these k systems satisfies the supermartingale requirement, and therefore, from Doob's martingale convergence, it must be the case that the value of barrier certificate must converge to its minimum. Now, by utilizing a similar argument to that of Theorem 3, under the condition that the solution process does not enter the set $\partial X \setminus \partial X_R$, we have that the solution process must almost surely enter the target set X_R. Therefore, solution process $\mathbf{x}_{x_0}$ starting from $x_0 \in X_0$ reaches the target set X_R over unbounded-time horizons with a probability as obtained in (4.41). □

Note again that, when $k = 1$ and $c = 0$, conditions (4.37)-(4.40) converge to standard barrier certificate conditions (2.27)-(2.29). One also observes that the probability bounds in (4.41) converge to those in (2.30) under the same conditions. Therefore, any barrier certificate satisfying conditions (2.27)-(2.29) is also a 1-inductive barrier certificate as in Definition 22.

Remark 25. *The probability bounds for reachability obtained in* (4.41) *are the same as the ones obtained for safety in* (4.30)*. This is due to the fact that we leverage the reach-while-avoid nature of conditions* (4.37)-(4.40)*, which ensure that the system avoids the set $\partial X \setminus \partial X_R$, and then utilizes Doob's martingale convergence [43] to ensure that the system reaches the target set X_R with a probability lower bound in* (4.41)*.*

We now illustrate k-inductive barrier certificates as in Definition 22 with the Markov Chain considered in Example 4.

Example 4 (Continued)**.** *Consider the finite Markov chain $\mathfrak{S}'$ of Figure 4.9. We already showed that there exists no linear barrier certificate as in Definition 10 for any $0 \leq \varepsilon < 1$. Now, we show that by using k-inductive barrier certificates as in Definition 22, we get more reliable probability bounds for the satisfaction of reachability.*

Consider $\mathbb{B}(x) = 0.1x + 0.01$, constants $\varepsilon = 0.03$, $c = 0.02$, and $k = 3$. The enumerated values of $\mathbb{B}(x), \mathbb{E}[\mathbb{B}(f(x,\varsigma) \mid x]$, $\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2)) \mid x]$ and $\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x]$ for all states $x \in X$ are provided in Table 4.2. We immediately see that condition (4.37) *is satisfied for the initial state $x = 0.2$. As we deal with a finite state system, and there are no states to avoid, there is no need to ensure the satisfaction of condition* (4.38)*. It can be seen that conditions* (4.39) *and* (4.40) *also hold for all $x \in X$. Therefore, $\mathbb{B}(x) = 0.1x + 0.01$ is indeed a linear 3-inductive barrier certificate for $\mathfrak{S}$. We apply Theorem 11 to obtain a lower bound on reachability probability as:*

$$\mathbb{P}\{x_{x_0}(t) \in X_R = \{0\} \text{ for some } t \in \mathbb{N} \mid x_0 = \{0.2\}\} \geq 0.85,$$

which provides better guarantees than the linear barrier certificate.

We now extend barrier certificates for reachability as in Definition 11 to k-inductive barrier certificates presented below.

Definition 23. *Consider a dt-SS $\mathfrak{S}$ that satisfies Assumption 1. We say that a function $\mathbb{B} : X \to \mathbb{R}_{\geq 0}$ is a k-inductive barrier certificate for $\mathfrak{S}$ with respect to a set of initial states $X \setminus X_R$ and a*

x	$\mathbb{B}(x)$	$\mathbb{E}[\mathbb{B}(f(x,\varsigma))\|x]$	$\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2))\|x]$	$\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3))\|x]$
0.2	0.03	0.05	0.03	0.02
0.3	0.04	0.05	0.03	0.02
0.5	0.06	0.01	0.01	0.01
0	0.01	0.01	0.01	0.01

Table 4.2: The values of $\mathbb{E}[\mathbb{B}(f_i(x,\varsigma_i)) \mid x]$ for all $i \in \{1,2,3\}$ and all $x \in X$ for Example 4. Note that $\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x] < \mathbb{B}(x)$ for all $x \in \overline{X \setminus X_R}$

.

set of target states X_R if there exist constants $k \in \mathbb{N}_{\geq 1}$, $\varepsilon \geq 0$, $c \geq 0$, and $\delta > 0$ such that the following conditions hold:

$$\mathbb{B}(x) \geq \varepsilon, \qquad \text{for all } x \in X \setminus X_R \tag{4.43}$$
$$\mathbb{B}(x) < \varepsilon \qquad \text{for all } x \in X_R, \tag{4.44}$$
$$\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x] - \mathbb{B}(x) \leq c \qquad \text{for all } x \in X \setminus X_R, \tag{4.45}$$
$$\mathbb{E}[\mathbb{B}(f_k(x,\varsigma_k)) \mid x] - \mathbb{B}(x) \leq -\delta \qquad \text{for all } x \in X \setminus X_R. \tag{4.46}$$

Similar to Definition 11, condition (4.45) requires the barrier certificate to be a c-martingale at every time step and condition (4.46) requires the barrier certificate sampled after every kth step to be decreasing in expectation for those states not in the set of target states. We now present the third result of our paper based on this definition of k-inductive barrier certificates.

Theorem 12. *Consider a dt-SS $\mathfrak{S} = (X, \varsigma, f)$ satisfying Assumption 1. Let $\mathbb{B}$ be a barrier certificate for $\mathfrak{S}$ satisfying conditions* (4.43)-(4.46) *with some $\varepsilon, c \geq 0$, $\delta > 0$, and $k \in \mathbb{N}_{\geq 1}$. Then a solution process $\mathbf{x}_{x_0}$ starting from an initial condition $x_0 \in X \setminus X_R$ reaches the target region X_R with probability* 1*, i.e.,*

$$\mathbb{P}\{\mathbf{x}_{x_0}(t) \in X_R \text{ for some } t \in \mathbb{N} \mid x_0\} = 1. \tag{4.47}$$

Proof. For the dt-SS $\mathfrak{S}$, consider k systems sampled after every k steps starting from initial conditions $x_0, x(1), \ldots, x(k-1)$ respectively. The dynamics of these systems are obtained as

$$\begin{aligned} x(t+k) &= f_k(x(t), \varsigma_k(t)), \\ x(t+k+1) &= f_k(x(t+1), \varsigma_k(t+1)), \\ &\vdots \\ x(t+2k-1) &= f_k(x(t+k-1), \varsigma_k(t+k-1)). \end{aligned}$$

Due to condition (4.46), the barrier certificate $\mathbb{B}$ satisfies the supermartingale condition for each of these systems. Therefore the probability of each of these systems eventually reaching some state in X_R is 1 by Theorem 4, *i.e.*, each of these systems eventually reach some state in X_R with probability 1. This implies that $\mathfrak{S}$ must satisfy the reachability specification with probability 1, as obtained in (4.47). □

x	$\mathbb{B}(x)$	$\mathbb{E}[\mathbb{B}(f(x,\varsigma))\|x]$	$\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2))\|x]$	$\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3))\|x]$
0.2	0.29	0.49	0.29	0.19
0.3	0.39	0.49	0.29	0.19
0.5	0.59	0.09	0.09	0.09
0	0.09	0.09	0.09	0.09

Table 4.3: The values of $\mathbb{E}[\mathbb{B}(f_i(x,\varsigma_i)) \mid x]$ for all $i \in \{1,2,3\}$ and all $x \in X$ for Example 4. Note that $\mathbb{E}[\mathbb{B}(f_3(x,\varsigma_3)) \mid x] < \mathbb{B}(x)$ for all $x \in X \setminus X_R$.

Remark 26. *The existence of barrier certificates as in Definition 23 gives a probability of* 1 *for reachability. This bound is independent of the values of the constants k, ε, c and δ. Therefore these constants can be set to any value that is greater than* 0 *and still give an almost sure guarantee of reachability to the target set.*

Note that when $c < 0$ and $k = 1$, conditions (4.43)-(4.46) reduce to standard barrier conditions (2.31)-(2.33). Therefore any barrier certificate satisfying conditions (2.31)-(2.33) is also a 1-inductive barrier certificate as in Definition 11. However, the converse may not hold true, as conditions (4.43)-(4.46) are more relaxed. We now illustrate k-inductive barrier certificates as in Definition 23 with the Markov chain considered in Example 4.

Example 4 (Continued)**.** *We once again consider the finite Markov chain $\mathfrak{S}'$ presented in Figure 4.9. We show that by using k-inductive barrier certificates as in Definition 23, we get that $\mathfrak{S}'$ satisfies the reachability specification with probability* 1*. Consider $\mathbb{B}(x) = x + 0.09$, constants $\varepsilon = 0.1$, and $c = 0.2$, and $k = 3$. The enumerated values of $\mathbb{B}(x), \mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x]$ and $\mathbb{E}[\mathbb{B}(f_2(x,\varsigma_2)) \mid x]$ are provided in Table 4.3. We immediately observe that condition* (4.43) *is satisfied for all states except $x = 0$ and similarly, condition* (4.44) *holds for the target state $x = 0$. Lastly conditions* (4.45) *and* (4.46) *also hold for all $x \in X \setminus X_R$. Therefore $\mathbb{B}(x) = x + 0.09$ is indeed a linear* 3*-inductive barrier certificate for $\mathfrak{S}'$. This allows one to conclude that the system reaches the state $x = 0$ with probability* 1*.*

4.5.2 Computation of k-Inductive Barrier Certificates

We now utilize Assumption 8 stated in Section 4.4 to formulate k-inductive barrier certificates for reachability as in Definition 22 as a collection of sum-of-squares constraints corresponding to conditions (4.37)-(4.40), which can be obtained by employing the following lemma.

Lemma 7. *Consider a dt-SS $\mathfrak{S}$. Suppose Assumption 8 holds and there exists a sum-of-squares polynomial $\mathbb{B}(x)$, constants $k \in \mathbb{N}_{\geq 1}$, $0 \leq \varepsilon \leq 1$, $\delta > 0$, and $c \geq 0$, and vectors of sum-of-squares polynomials $\lambda_0(x), \lambda_b(x)$, $\lambda_c(x)$ and $\hat{\lambda}_c(x)$ of appropriate dimensions such that the following*

expressions are sum-of-squares polynomials:

$$-\mathbb{B}(x) - \lambda_0^T(x)g_0(x) + \varepsilon, \tag{4.48}$$
$$\mathbb{B}(x) - \lambda_b^T(x)g_b(x) - 1, \tag{4.49}$$
$$-\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x] + \mathbb{B}(x) - \lambda_c^T(x)g_c(x) + c, \tag{4.50}$$
$$-\mathbb{E}[\mathbb{B}(f_k(x,\varsigma_k)) \mid x] + \mathbb{B}(x) - \hat{\lambda}_c^T(x)g_c(x) - \delta. \tag{4.51}$$

Then function $\mathbb{B}(x)$ is a k-inductive barrier certificate as in Definition 22 satisfying conditions (4.37)-(4.40)*.*

Similarly one may use the following lemma to find k-inductive barrier certificates for reachability according to Definition 23.

Lemma 8. *Consider a dt-SS $\mathfrak{S}$. Suppose Assumption 8 holds and there exists a sum-of-squares polynomial $\mathbb{B}(x)$, constants $k \in \mathbb{N}_{\geq 1}$, $0 \leq \varepsilon \leq 1$, $\delta > 0$, and $c \geq 0$, and vectors of sum-of-squares polynomials $\lambda_r(x)$, $\lambda_c(x)$, $\hat{\lambda}_c(x)$ and $\bar{\lambda}_c(x)$ of appropriate dimensions such that the following expressions are sum-of-squares polynomials:*

$$\mathbb{B}(x) - \bar{\lambda}_c^T(x)g_z(x) - \varepsilon, \tag{4.52}$$
$$-\mathbb{B}(x) - \lambda_r^T(x)g_r(x) + \varepsilon - \epsilon, \tag{4.53}$$
$$-\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x] + \mathbb{B}(x) - \lambda_c^T(x)g_z(x) + c, \tag{4.54}$$
$$-\mathbb{E}[\mathbb{B}(f_k(x,\varsigma_k)) \mid x] + \mathbb{B}(x) - \hat{\lambda}_c^T(x)g_z(x) - \delta, \tag{4.55}$$

where ϵ is a small positive constant used to ensure the satisfaction of strict inequality (4.44)*. Then the function $\mathbb{B}(x)$ is a k-inductive barrier certificate as in Definition 23 satisfying conditions* (4.43)-(4.46)*.*

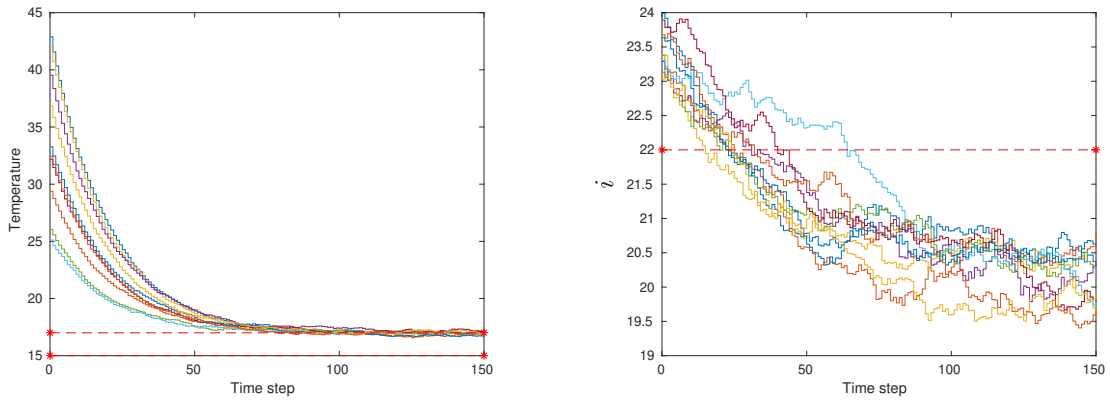
Remark 27. *The expected value $\mathbb{E}[\mathbb{B}(f(x,\varsigma)) \mid x]$ can be evaluated when the probability distribution of the stochastic variable ς is known by considering all the monomials of the polynomial expression $\mathbb{B}(f_k(x,\varsigma_k))$ and utilizing the moments of the distribution of ς. For a Gaussian distribution, this can be done in linear time.*

Remark 28. *The SOS optimization problem is solved by fixing the degree d of the polynomial function $\mathbb{B}(x)$ along with the value of k. In general, if one cannot find a suitable function $\mathbb{B}(x)$ that satisfies the required constraints, one needs to solve the problem with a higher degree polynomial $\mathbb{B}(x)$ or a higher value of k. Note that for a fixed state dimension, the computational complexity grows polynomially with respect to d [93], and only linearly with k. Therefore, it would be more beneficial to use k-inductive barrier certificates with higher values of k than higher values of d.*

4.5.3 Case Study

In this case study, we consider reachability specification for the temperature evolution of a room. The thermal model for the room is adapted from [62]. The dynamics of the dt-SS $\mathfrak{S}$ are given as

$$\mathfrak{S} : x(t+1) = (1 - \tau_s\alpha)x(t) + \tau_s\alpha T_e + G\varsigma(t),$$



(a) (b)

Figure 4.10: (a) Solution processes of $\mathfrak{S}$ from Section 4.5.3, starting from different initial states in $X \setminus X_R$. (b) Solution processes of $\mathfrak{S}'$ from Section 4.5.3. In both figures, the set X_R is highlighted by red dashed lines.

where $\alpha = 0.01$ is the heat exchange coefficient, $T_e = 17$ is the ambient temperature, $\tau_s = 5$ minutes is the sampling time and $G = 0.05$ is the noise coefficient. The state space of the system is given as $X = [15, 35]$, whereas the target set is specified as $X_R = [15, 17]$. We aim to utilize barrier certificates for reachability as in Definition 11 to verify whether $\mathfrak{S}$ satisfies reachability with probability 1. To do so, we first consider the barrier certificate to be a polynomial of degree 2, and search for a suitable barrier certificate satisfying conditions (2.31)-(2.33) by considering $X \setminus X_R = [17 + \vartheta, 35]$, where $\vartheta = 0.001$, and reformulating them into SOS constraints with tolerance parameters $\epsilon, \delta = 0.01$. However, we fail to find a suitable barrier certificate satisfying conditions (2.31)-(2.33). Therefore, using standard barrier certificates for reachability according to Definition 11, one cannot verify whether $\mathfrak{S}$ satisfies reachability with probability 1.

Instead, let us compute a suitable polynomial k-inductive barrier certificate of degree 2 as in Definition 23. We can reformulate conditions (4.43)-(4.46) as an SOS problem via Lemma 8. Setting $k = 11$, $\varepsilon = 1300$, $c = 0.001$, and $\delta = 0.01$, we obtain $\mathbb{B}(x) = 166.5118 + 34.7652x + 1.8769x^2$ as a k-inductive barrier certificate satisfying conditions (4.52)-(4.55) with a tolerance parameter $\epsilon = 0.01$. From Lemma 8 and Theorem 12, it follows that the system $\mathfrak{S}$ indeed satisfies the reachability specification with probability 1. Figure 4.10a shows 10 representative solution processes starting from different initial conditions inside $X \setminus X_R$. The computations take 15 seconds on our machine running with Linux Ubuntu OS (Intel i7 – 8665U CPU with 32GB of RAM). Note that we use MATLAB 2019b to perform our computations.

We now modify the parameters of the dynamics and consider a system $\mathfrak{S}'$ such that we can find standard barrier certificates satisfying Definition 10. Consider $\alpha = 0.004$, $T_e = 20$, $\tau_s = 5$ and $G = 0.08$ as the noise coefficient. The state space of the system is $X = [18, 45]$, the initial set of states $X_0 = [23, 24]$ and the target set $X_R = [18, 22]$. We first consider the barrier certificate to be a polynomial of degree 2 and search for a suitable barrier certificate satisfying conditions (2.24)-(2.26) by considering the sets $\partial X \setminus \partial X_R = [44 + \vartheta, 45]$, where $\vartheta = 0.01$,

and $\overline{X \setminus X_R} = [22, 45]$, and reformulate them into SOS constraints with $\delta = 0.001$. We find a barrier certificate $\mathbb{B}(x) = 0.3658 - 0.05066x + 0.0018x^2$ satisfying conditions (2.24)-(2.26) for $\varepsilon = 0.24$. Thus, by utilizing Theorem 3, we get the lower bound on the probability of satisfying reachability as 0.76. We now consider a k-inductive barrier certificate as in Definition 22. For $k = 2$, $\varepsilon = 0.054$, $c = 0.0001$, and $\delta = 0.001$, we obtain $\mathbb{B}(x) = 1.1837 - 0.1196x + 0.003x^2$ as a k-inductive barrier certificate satisfying conditions (4.48)-(4.51). Then, by utilizing Theorem 11, we can say the system $\mathfrak{S}'$ satisfies the reachability specification with a probability of at least 0.89 which is greater than the lower bound obtained by using standard barrier certificates. This illustrates that even when standard barrier certificates exist, we may obtain more reliable probabilities for satisfaction with k-inductive barrier certificates (see Remark 25). Figure 4.10b shows 10 representative solution processes starting from X_0. The computation time for this approach is about 7 seconds with the mentioned tools and machine.

4.6 Conclusion

In this chapter, we leveraged the k-induction principle to propose several notions of k-inductive barrier certificates for (stochastic) dynamical systems that extend the standard barrier certificates while providing less conservative conditions that are easier to satisfy. By doing so, larger classes of functions may act as barrier certificates, making the synthesis of safety and reachability certificates more likely to be successful. In particular, we first proposed two different notions of k-inductive barrier certificates for the safety of discrete-time dynamical systems and motivated their use through a simple finite state transition system. We also compared the two notions and illustrated via an example that the second notion is more expressive than the first. Secondly, we proposed a notion of k-inductive barrier certificates for the probabilistic safety verification of discrete-time stochastic dynamical systems. We demonstrated that k-inductive barrier certificates do not require to satisfy the restrictive supermartingale condition, and are still able to provide lower bounds on the probability of safety satisfaction over infinite time horizons. Third, we proposed two different definitions of k-inductive barrier certificates for reachability verification of discrete-time stochastic dynamical systems. While the first definition provides lower bounds on the probability that the system satisfies reachability properties, the second definition provides almost sure reachability guarantees. We illustrate via examples that both these definitions are less conservative than their standard barrier certificate counterparts.

Finally, we presented computational techniques for the synthesis of k-inductive barrier certificates via sum-of-squares optimization and δ-complete decision procedures (where applicable). Moreover, we presented various case studies to demonstrate the effectiveness of our proposed approaches.

Chapter 5

Formal Analysis of Complex Logic Specifications

5.1 Introduction

Classical control problems can involve checking complex mathematical models against relatively simple properties like stability, invariance, or reachability. However, in many real-world applications, the systems are required to perform complicated logic tasks. For example, many robotic applications need to satisfy some motion planning objectives. On the other hand, self-driving cars may be required to avoid obstacles (*i.e.*, other cars, pedestrians, etc.) while agreeing to a pre-defined set of traffic rules. Barrier certificate-based approaches developed in the previous chapters can inherently tackle either safety or reachability specifications. Unfortunately, they cannot be directly utilized to tackle arbitrary logic specifications.

In the formal methods community, complicated logic tasks are traditionally expressed using classical temporal logic specifications such as linear temporal logic (LTL), ω-regular properties or automata over (in)finite traces. For example, consider a robot that needs to sequentially visit boxes to pick up an item and drop it at the next box. This specification can be expressed using linear temporal logic or as an automaton. As such, the aforementioned specifications express properties on a set of desirable system executions. While these logics can describe a large number of specifications of interest that consider individual execution traces of systems, many important information-flow properties and planning objectives involve relating multiple execution traces. These properties cannot be expressed by classical temporal logic specifications equipped to express properties of individual traces. Hyperproperties, on the other hand, are properties of collective behaviour relating to multiple traces. For example, suppose that a system requires that some secret information is never revealed, *i.e.*, observations from the outside remain indistinguishable from each other, despite the secret. This specification, known as opacity [82], requires us to relate and quantify two observation traces simultaneously. Similarly, an optimality objective [127] for a robotic system would require the existence of a trace that is more favorable than all the other traces of the system, again quantifying multiple execution traces at a time. Other examples of hyperproperties include noninterference [55] and observational determinism [103].

HyperLTL, developed as an extension to LTL to specify hyperproperties, uses trace variables to denote individual execution traces and utilizes universal and existential quantifiers over atomic propositions to specify on which traces atomic propositions must hold.

In this chapter, we consider the formal analysis (verification and/or synthesis) of (stochastic) control systems for the aforementioned logic specifications. In particular, we propose an automata-theoretic framework for the analysis, wherein we utilize the automata corresponding to the specifications to decompose the high-level specifications into a set of smaller safety problems. Then, by combining the safety guarantees obtained from barrier certificate-based approaches, one is able to provide guarantees for the original specification.

5.1.1 Related Literature

Analysis of Temporal Logic Specifications

There have been several results in the literature for the verification and synthesis of control systems against temporal logic specifications. Many earlier results have utilized abstraction-based techniques which need discretization of the state sets [122]. Examples include abstraction-based framework for linear systems [123], for nonlinear systems [135], synthesizing feedback strategies for piece-wise affine systems [134], and counterexample-guided abstraction refinement (CEGAR) for nonlinear systems [130] to name a few. More recently, barrier certificate approaches have also been utilized for the verification and synthesis of LTL specifications. Examples in this direction include automata-theoretic approaches for verification of non-stochastic nonlinear systems [131], via eventuality-based barrier certificates for hybrid systems [19], using composite control barrier functions for a fragment of LTL for robotic tasks in [116], and automata-theoretic approach for verification and synthesis of stochastic systems [62, 63, 64]. Barrier certificate-based approaches have also been used for other classes of temporal logics defined over individual execution traces, such as signal temporal logic [75, 74, 133].

Analysis of Hyperproperties

Unfortunately, most of the existing results pertaining to hyperproperties are tailored to finite-state transition systems. For example, the results in [50] present a practical verification approach for finite-state systems with respect to alternation-free fragments of HyperLTL formulae. The proposed approaches in [30] present a model-checker for HyperLTL specifications with an alternation depth of at most one. The results in [49] propose a new model-checking algorithm based on model-counting for quantitative hyperproperties. A bounded model checking algorithm for hyperproperties is proposed in [60]. Verification of other types of hyperproperties such as k-safety hyperproperties and hyperliveness properties have also been studied in [46] and [32], respectively. Checking satisfiability of certain fragments of HyperLTL specifications, such as the "$\forall^*\exists^*$" fragment, are undecidable in general [47]. Formal verification of continuous state-space CPS against general hyperproperties remains largely unexplored. Hyperproperties have been studied for continuous-space systems in [90] as well as [128], but in the context of falsification and statistical model checking, respectively.

5.1.2 Contributions

The focus of this chapter is to extend the (control) barrier certificate-based approaches to the formal analysis of complex logic specifications that can be expressed using linear temporal logic over finite traces or as deterministic finite automata (DFA), ω-regular specifications, as well as hyperproperties expressed using HyperLTL. To do so, we use a divide-and-conquer approach to decompose the complex specifications into smaller tasks based on the structure of the automaton associated with the specification, and then solve the smaller verification or synthesis tasks via (control) barrier certificates. Then, we combine the guarantees obtained via (control) barrier certificates to obtain satisfaction guarantees for the original specification.

This chapter is organized as follows. The second section of this chapter (Section 5.2) is concerned with the controller synthesis task for (possibly interconnected) stochastic control systems against specifications that can be represented using deterministic finite automata (DFA). As such, these specifications are a generalization of LTL specifications that are defined over finite traces, *i.e.*, LTL_F. For such specifications, we first compile DFA corresponding to the negation of the specification and decompose the automata into a sequence of smaller safety tasks. Then, for each safety task, we utilize control barrier certificates to synthesize suitable controllers and obtain probability upper bounds on the safety violation over finite time horizons. Then, we combine such guarantees obtained for the safety tasks to provide overall probability upper bounds on the violation of the DFA corresponding to the negation of the specifications. By complementing this probability, we can suitably obtain the required probability lower bounds on the satisfaction of the original specifications. Correspondingly, we also propose a switching controller structure by combining the controllers obtained for the safety tasks that ensure the satisfaction of the specifications with the obtained probability lower bounds. Finally, we demonstrate our approach by extending the safety synthesis of the network of Kuramoto oscillators presented in Section 3.3 of Chapter 3 to a specification represented by DFA.

In the third section (Section 5.3), we consider the controller synthesis for (possibly interconnected) stochastic systems against ω-regular properties. These specifications are defined over infinite time horizons and can be described by the accepting languages of deterministic Streett automata (DSA). We provide a systematic approach to decompose these high-level specifications to simpler safety synthesis tasks by employing the automata corresponding to the specifications. Then, by utilizing control barrier certificates for these safety tasks, we obtain probability guarantees on the violation of these safety tasks, which can be combined to obtain the overall lower bound on the probability that the original ω-regular specification is satisfied. Correspondingly, we also obtain a switching controller to ensure that the concerned stochastic control system satisfies the original specification. Finally, we demonstrate the effectiveness of our approach by extending the safety synthesis obtained for the room temperature network considered in Section 3.4 of Chapter 3 to an ω-regular specification expressed using a suitable DSA.

The fourth section (Section 5.4) aims to propose for the first time a discretization-free, barrier certificate-based verification procedure against hyperproperties. In particular, we consider those specifications that can be expressed by HyperLTL formulae. The verification procedure is achieved by decomposing the given specification into simpler safety tasks, so-called *conditional invariance*, by constructing an implicitly quantified Büchi automaton corresponding to

the complement of the specification. We introduce *augmented barrier certificates* (ABCs), defined over an augmented system obtained by taking the product of the original system with itself (self-composition), which provide us with sufficient conditions ensuring the satisfaction of those conditional invariances. Then, we propose an automata-theoretic approach to extend the applicability of ABCs beyond conditional invariance to HyperLTL specifications by finding barrier certificates ensuring the possibility of avoiding accepting traces of corresponding automata by disallowing certain transitions on different lassos (a simple path followed by a simple accepting cycle). To do so, the "existential" player is required to select a trace before knowing the choices of the "universal" player. This necessitates the need for a common ABC for some transitions of all lassos, which may be hard to ensure in practice. On the other hand, when the HyperLTL property belongs to $\forall^*\exists^*$-fragment [30], we exploit separate ABCs to provide the necessary guarantees by leveraging the structure of the automata corresponding to the negation of specifications. For systems with polynomial-type dynamics, we present a sum-of-squares (SOS) approach to compute polynomial-type ABCs for the individual conditional invariance. Finally, we demonstrate the effectiveness of our proposed approach by verifying two physical case studies with respect to initial-state opacity and initial-state robustness, respectively.

We must mention that the results presented in this chapter appear in our publications [5, 6, 10]. In particular, Section 5.2 is based on [5] which appeared as a technical note in Transactions of Automatic Control. Section 5.3 is based on the journal paper [6] that has been accepted for publication in Nonlinear Analysis: Hybrid Systems. Both these publications were joint works with Abolfazl Lavaei and Majid Zamani. The author of the thesis has established the results and written the drafts. Abolfazl Lavaei contributed to the initial discussions, revision of the draft as well as mentoring. Majid Zamani supervised the work. The results of Section 5.4 is based on [10] which has been conditionally accepted as a full paper in Transactions of Automatic Control. An extended abstract version of these results also appeared in [7]. This was a joint work with Vishnu Murali, Ashutosh Trivedi and Majid Zamani. The author of the thesis and Vishnu Murali have contributed equally towards the technical results and preparation of the manuscript, whereas Ashutosh Trivedi and Majid Zamani provided the necessary support and supervision.

5.2 Controller Synthesis for Stochastic Control Systems against Specifications as DFA

In this section, we deal with a general class of specifications that can be expressed by deterministic finite automata (DFA), as has been defined in Definition 4 in Chapter 2. The primary goal of this section is to synthesize suitable controllers for (possibly large-scale interconnected) stochastic control systems $\mathfrak{S} = (X, U, \varsigma, f)$ against specifications described by DFA $\mathcal{A}^f = (Q, q_0, \Sigma, \delta, F)$, where $\Sigma = 2^{\mathcal{AP}}$. To do so, we provide an automata-theoretic approach to decompose the DFA $\mathcal{A}^f$ into a collection of safety specifications. Then, we utilize c-martingale control barrier certificates, and correspondingly, results from Corollary 1 or Theorem 5, to provide probabilistic safety guarantees over finite time horizons. These probabilistic guarantees are then combined to obtain the overall lower bound on the probability of the satisfaction of the original DFA

specification.

5.2.1 Problem Definition

First, we recollect from Chapter 2 that a finite trace $\boldsymbol{\sigma}_f = (\sigma_0, \ldots, \sigma_{n-1}) \in \Sigma^*$ is said to be accepted by DFA $\mathcal{A}^f$ if there exists a corresponding finite path $\mathbf{q}_f = (q_0, q_1, \ldots, q_n) \in Q^{n+1}$ with $q_{m+1} = \delta(q_m, \sigma_m)$, $\forall m \in \{1, \ldots, n\}$ and $q_n \in F$. In order to determine whether the stochastic control system $\mathfrak{S}$ satisfies the specification represented by DFA $\mathcal{A}^f$, it is sufficient to determine whether the traces corresponding to the solution processes of $\mathfrak{S}$ are accepted by DFA $\mathcal{A}^f$. To do so, one must first relate the solution processes of $\mathfrak{S}$ to the traces of $\mathcal{A}^f$. This can be done as follows.

Definition 24. *For an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$ and a DFA $\mathcal{A}^f = (Q, q_0, \mathcal{AP}, \delta, F)$, consider a labeling function $L : X \to \mathcal{AP}$. For a finite state sequence $\mathbf{x}_M = (\mathbf{x}(0), \mathbf{x}(1), \ldots, \mathbf{x}(M-1)) \in X^M$ of a length $M \in \mathbb{N}$, the corresponding finite trace over $\mathcal{AP}$ is given by $L(\mathbf{x}_M) := (\sigma_0, \sigma_1, \ldots, \sigma_{M-1}) \in \mathcal{AP}^M$, where $\sigma_i = L(x(i))$ for all $i \in \{0, 1, \ldots, M-1\}$.*

Remark 29. *A DFA $\mathcal{A}^f$ is normally constructed over the alphabet $\Sigma = 2^{\mathcal{AP}}$. However, without loss of generality, we work here with the set of atomic propositions $\mathcal{AP}$ as the alphabet rather than its power set $2^{\mathcal{AP}}$, i.e., $\Sigma = \mathcal{AP}$. This is due to the fact that for any two atomic propositions $p_i, p_j \in \mathcal{AP}$, $i, j \leq |\mathcal{AP}|$, we have $p_i \wedge p_j = \emptyset$, and therefore edges with conjunctions between atomic propositions can be removed from the DFA. Moreover, other Boolean combinations like disjunction and negation can be easily resolved by adding parallel edges with simple atomic propositions in each of the edges, respectively.*

Remark 30. *Note that all LTL specifications over finite-time horizons (i.e. LTL_f) can be represented by DFA (see Chapter 2). Note that specifications represented by DFA are more expressive than LTL_f [38]. This is the reason we consider specifications expressed by DFA directly rather than those expressed by LTL_f.*

We now define the probability that the solution processes of the (interconnected) system satisfy a specification over a time horizon M.

Definition 25. *Consider an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$, a specification given by the accepting language of a DFA $\mathcal{A}^f = (Q, q_0, \mathcal{AP}, \delta, F)$ and a labeling function $L : X \to \mathcal{AP}$. Then, the probability with which the solution process $\mathbf{x}_{a,\varpi,M}$ of $\mathfrak{S}$ of length $M \in \mathbb{N}$ starting from an initial condition $x(0) = x_0$ under the controller ϖ, satisfies the specification expressed by $\mathcal{A}$ is denoted by $\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,M}) \models \mathcal{A}^f\}$.*

The synthesis problem considered in this section involves computing a controller in conjunction with a tight lower bound on the probability of satisfaction over the (interconnected) dt-SCS $\mathfrak{S}$. This problem can be formally presented as follows.

Problem 7. *Given an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$, a desired specification admitted by the accepting language of the DFA $\mathcal{A}^f = (Q, q_0, \mathcal{AP}, \delta, F)$ over a set of atomic propositions $\mathcal{AP} = \{p_0, p_1 \ldots, p_R\}$, $R \in \mathbb{N}$, and a labeling function $L : X \to \mathcal{AP}$, compute a controller ϖ and a constant $\varkappa \in [0, 1]$ such that $\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,M}) \models \mathcal{A}\} \geq \varkappa$.*

To tackle this problem, we utilize a DFA representing the complement of the complex specification and decompose it into simpler safety tasks. For each such task, we aim to find a suitable c-martingale CBC (see paragraph below Definition 8, or Definition 14) along with a controller for the (interconnected) dt-SCS that gives the probability upper bound on the violation of safety tasks over finite time horizons. Note that CBCs for safety may be constructed monolithically (*i.e.* for a non-interconnected stochastic control system) as per Corollary 1 or by using a compositional framework (*i.e.*, for an interconnected system) as in Chapter 3. In the case of the latter, consider two sets X_0 and X_u as introduced in Definition 14. Consider that these sets are connected to atomic propositions $\mathcal{AP}$ through some labeling function $L : X \to \mathcal{AP}$. We assume that those sets can be decomposed as $X_0 = \prod_{i=1}^{N} X_{0_i}$ and $X_u = \prod_{i=1}^{N} X_{u_i}$. By doing so, one can simply compute a CSBC for each subsystem separately and utilize the compositional framework to obtain a CBC for the interconnected system. Similarly, it is assumed that atomic propositions in the set $\mathcal{AP}$ can also be decomposed accordingly. This implies that sets X_{0_i} and X_{u_i}, $i \in \{1, \ldots, N\}$, are also connected to the corresponding decomposed structure of $\mathcal{AP}$.

In the following section, we discuss the decomposition procedure and explain in detail the computation of probability bounds on the satisfaction of specifications represented by DFA.

5.2.2 Specification Decomposition

In this subsection, we describe the specification decomposition procedure, in which we divide a complex specification into simpler safety tasks by utilizing the automaton representing the complement of the specification. This was initially proposed in [63] for the synthesis of monolithic systems.

For a DFA $\mathcal{A}^f = (Q, q_0, \mathcal{AP}, \delta, F)$ that describes the property of interest, consider the complement DFA $\bar{\mathcal{A}}^f = (Q, q_0, \mathcal{AP}, \delta, \bar{F})$ with $\bar{F} = Q \setminus F$ whose accepting language consists of all finite words not present in $\mathcal{L}(\mathcal{A}^f)$. Therefore, if a system violates the satisfaction of the acceptance condition of $\bar{\mathcal{A}}^f$, then the system satisfies the acceptance condition of the original specification given by $\mathcal{A}^f$. A sequence $\mathbf{q} = (q_0, q_1, \ldots, q_n \in Q^{n+1})$ is an accepting state run of $\bar{\mathcal{A}}^f$ if $q_n \in \bar{F}$ if there exists a finite trace $\boldsymbol{\sigma} = (\sigma_0, \sigma_1, \ldots, \sigma_{n-1})$ such that $q_{i+1} = \delta(q_i, \sigma_i)$ for all $i \in \{0, 1, \ldots, n-1\}$. The length of the accepting state run is given by $|\mathbf{q}| = n + 1$. Let $Q_z \subseteq Q$ denote the set of states having self-loops. Let $\mathcal{R}_M, M \in \mathbb{N}$, be the set of all finite accepting state runs of at most length $M + 1$ excluding self-loops, where

$$\mathcal{R}_M := \{\mathbf{q} = (q_0, q_1, \ldots, q_m) \in Q^{m+1} \,\big|\, m \leq M, q_m \in \bar{F}, q_i \neq q_{i+1}, \forall i < m\}.$$

$\mathcal{R}_M$ can be computed algorithmically by considering the DFA as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = Q$ are vertices representing states of the DFA and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ are edges such that $(q, q') \in \mathcal{E}$ if and only if $q' \neq q$, and there exists $\sigma \in \mathcal{AP}$ such that $\delta(q, \sigma) = q'$. It can be readily observed that a finite path starting at the vertex q_0 and ending at a vertex $q_n \in \bar{F}$ is an accepting state run $\mathbf{q}$ of $\bar{\mathcal{A}}^f$ without any self-loops, and therefore, it belongs to $\mathcal{R}_M$. Using algorithms provided in the graph theory such as the depth-first search algorithm [105], one can readily obtain $\mathcal{R}_M$. Now, for each $p \in \mathcal{AP}$, we define $\mathcal{R}_M^p$ as

$$\mathcal{R}_M^p := \{\mathbf{q} = (q_0, q_1, \ldots, q_m) \in \mathcal{R}_M \mid q_1 = \delta(q_0, p) \in \mathcal{AP}\}.$$

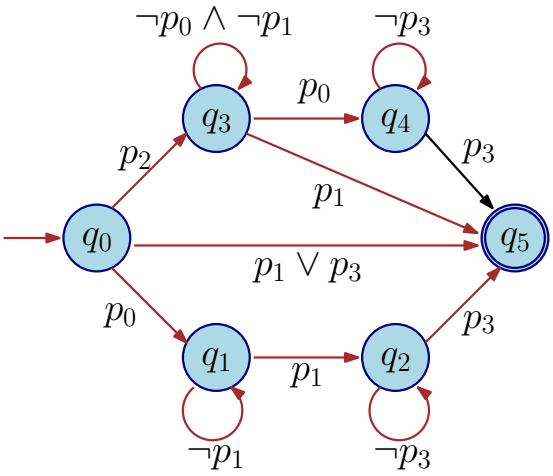


Figure 5.1: DFA $\bar{\mathcal{A}}^f$ employed in Example 5.

We consider any $\mathbf{q} = (q_0, q_1, \ldots, q_m) \in \mathcal{R}_M^p$ and define $\mathcal{P}^p(\mathbf{q})$ as a set of all state runs augmented with a horizon as

$$\mathcal{P}^p(\mathbf{q}) := \{(q_i, q_{i+1}, q_{i+2}), T_h(\mathbf{q}, q_{l+1}) \mid 0 \leq i \leq m-2\}. \tag{5.1}$$

We correspondingly define the set $\mathcal{S}^p(\mathbf{q})$ to be the set of all *consecutive transition pairs* $(\sigma_{X_0}, \sigma_{X_u}) \in \mathcal{AP}$ such that

$$\mathcal{S}^p(\mathbf{q}) = \{(\sigma_{X_0}, \sigma_{X_u}) \mid q_{i+1} = \delta(q_i, \sigma_{X_0}), q_{i+2} = \delta(q_{i+1}, \sigma_{X_u}), (q_i, q_{i+1}, q+i+2) \in \mathcal{P}^p(\mathbf{q})\} \tag{5.2}$$

Each element in $\mathcal{P}^p(\mathbf{q})$ has a length of three and is augmented with a time horizon which is given by $T_h(\mathbf{q}, q_{l+1}) = M + 2 - |\mathbf{q}|$ for $q_{l+1} \in Q_z$, and 1 otherwise. Consecutive transition pairs in $\mathcal{S}^p(\mathbf{q})$ corresponding to elements in $\mathcal{P}^p(\mathbf{q})$ are referred to as safety tasks. Consequently, we define $\mathcal{P}_M(\bar{\mathcal{A}}^f) = \bigcup_{p \in \mathcal{AP}} \bigcup_{\mathbf{q} \in \mathcal{R}_M^p} \mathcal{P}^p(\mathbf{q})$ as the set of all state runs of length 3 arising from different accepting state runs of a length less than or equal to $M+1$.

Remark 31. *Note that $\mathcal{P}^p(\mathbf{q}) = \emptyset$ for those accepting state runs whose length is 2. Any such sequences begin from a subset of the state space that already violates the desired specification and the outcome is accordingly a trivial zero probability for the satisfaction of the specification. Hence, we neglect such accepting state runs.*

Remark 32. *The self-loops play a pivotal role in the computation of the time horizon $T_h(\mathbf{q}, q_{i+1})$ for any safety task $\vartheta = (q_i, q_{i+1}, q_{i+2}) \in \mathcal{P}^p(\mathbf{q})$. This is crucial to account for the number of time steps that the solution process can remain in the self-loop $q_{i+1} \in Q_z$ before reaching q_{i+2} [63].*

We illustrate the decomposition of DFA $\bar{\mathcal{A}}^f$ into sequential safety tasks with the help of a running example.

Example 5. *Consider a DFA $\bar{\mathcal{A}}^f$ as shown in Figure 5.1. According to the definition of DFA, initial state is q_0, set of atomic propositions $\mathcal{AP} = \{p_0, p_1, p_2, p_3\}$ and set of final states $\bar{F} = \{q_5\}$. The set of states with self-loops is given by $Q_z = \{q_1, q_2, q_3, q_4\}$. We only consider accepting state runs with lengths less than or equal to 5, i.e., $M = 4$. The set of such accepting state runs without self-loops is*

$$\mathcal{R}_4 = \{(q_0, q_5), (q_0, q_3, q_5), (q_0, q_1, q_2, q_5), (q_0, q_3, q_4, q_5)\}.$$

The sets $\mathcal{R}_4^p$ for all $p \in \mathcal{AP}$ are given by

$$\mathcal{R}_4^{p_0} = \{(q_0, q_1, q_2, q_5)\}, \;\; \mathcal{R}_4^{p_1} = \{(q_0, q_5)\},$$
$$\mathcal{R}_4^{p_2} = \{(q_0, q_3, q_5), (q_0, q_3, q_4, q_5)\}, \;\; \mathcal{R}_4^{p_3} = \{(q_0, q_5)\}.$$

For all $\mathbf{q} \in \mathcal{R}_4^p$, we define $\mathcal{P}^p(\mathbf{q})$ as

$$\mathcal{P}^{p_0}(q_0, q_1, q_2, q_5) = \{(q_0, q_1, q_2, 2), (q_1, q_2, q_5, 2)\}, \;\; \mathcal{P}^{p_1}(q_0, q_5) = \mathcal{P}^{p_3}(q_0, q_5) = \emptyset,$$
$$\mathcal{P}^{p_2}(q_0, q_3, q_5) = \{(q_0, q_3, q_5, 3)\}, \;\; \mathcal{P}^{p_2}(q_0, q_3, q_4, q_5) = \{(q_0, q_3, q_4, 2), (q_3, q_4, q_5, 2)\}.$$

For each $\mathbf{q} \in \mathcal{R}_4$, the corresponding finite traces $\boldsymbol{\sigma}(\mathbf{q})$ are given by

$$\boldsymbol{\sigma}(q_0, q_5) = \{(p_1 \vee p_3)\}, \;\; \boldsymbol{\sigma}(q_0, q_3, q_5) = \{(p_2, p_1)\},$$
$$\boldsymbol{\sigma}(q_0, q_1, q_2, q_5) = \{(p_0, p_1, p_3)\},$$
$$\boldsymbol{\sigma}(q_0, q_3, q_4, q_5) = \{(p_2, p_0, p_3)\}.$$

Now, for each safety task, we construct an appropriate CBC along with a corresponding controller to obtain an upper bound on the probability that the interconnected system $\mathfrak{S}$ reaches unsafe regions in finite-time horizons. We now raise the following lemma to compute CBCs and the probabilities with which safety tasks are violated.

Lemma 9. *For an accepting state run $\mathbf{q} \in \mathcal{R}_M^p$ for some $M \in \mathbb{N}$ and some $p \in \mathcal{AP}$, consider $\vartheta = (q, q', q'') \in \mathcal{P}^p(\mathbf{q})$ and its corresponding safety task $(\sigma_{X_0}, \sigma_{X_u})$, associated with time horizon T_h. If there exists a c-martingale CBC and a controller ϖ with respect to $X_0 = L^{-1}(\sigma_{X_0})$ and $X_u = L^{-1}(\sigma_{X_u})$, then the upper bound on the probability that a solution process of dt-SCS $\mathfrak{S}$ starts from an initial state $x_0 \in X_0$ under the controller ϖ and reaches X_u within the finite-time horizon $[0, T_h)$ is obtained by utilizing Corollary 1 (or Theorem 5) and is given as*

$$\mathbb{P}\{\mathbf{x}_{x_0,\varpi,M} \in X_u \text{ for some } t \in [0, T_h) \mid x_0\} \leq \varepsilon_{\vartheta,T_h}, \tag{5.3}$$

where $\varepsilon_{\vartheta,T_h}$ is obtained via equation (2.22) *or equation* (3.10).

Once we compute the CBCs and the corresponding upper bound probabilities for all individual safety tasks, we combine them to obtain an upper bound probability of satisfaction of the property expressed by $\bar{\mathcal{A}}^f$, or in other words, an upper bound on the probability of violation of the specification given by the accepting language of $\mathcal{A}^f$. Consequently, we quantify a lower bound on the probability of satisfaction together with a controller that ensures the satisfaction of the desired specification given by $\mathcal{A}^f$. The next section explains the structure of this controller as well as the proposed procedure to compute the lower bound on the probability that the overall complex specification is satisfied by the interconnected system.

5.2.3 Controller and Probability Computation

Controller Structure

Computing CBC and its corresponding controller for the specification described by each individual safety task could be ambiguous when applying the controllers for $\mathfrak{S}$ in a closed-loop fashion. To clarify this, we consider the DFA $\bar{\mathcal{A}}^f$ from Figure 5.1. Moreover, consider the set $\mathcal{P}_M(\bar{\mathcal{A}}^f)$ for accepting state runs of a length of at most $M+1$, as obtained in Example 5. The safety tasks corresponding to $\vartheta_1 = (q_0, q_3, q_4, 2)$ and $\vartheta_2 = (q_0, q_3, q_5, 3)$ constitute two individual problems: one for computing the upper bound of reaching the region $L^{-1}(p_0)$ from $L^{-1}(p_2)$ and the other one for reaching the region $L^{-1}(p_1)$ from the same region $L^{-1}(p_2)$. Ideally, one should find two different CBCs and controllers. But since there are two outgoing transitions from state q_3, namely $\delta(q_3, p_0)$ and $\delta(q_3, p_1)$, computing different controllers means that the region $L^{-1}(p_2)$ employs two different controllers simultaneously and this issue results in ambiguity in the closed-loop system.

One potential solution to tackle this problem is to replace X_u in Lemma 9 with the union of regions and combine the two safety problems into one. This results in a common CBC and controller for different safety tasks. In other words, we partition $\mathcal{P}_M(\bar{\mathcal{A}}^f)$ and combine the safety tasks with the same CBC and controller and place them in a single partition set. Consequently, we obtain a switching controller since multiple locations in the automaton $\bar{\mathcal{A}}^f$ admit different controllers. In order to represent such a switching policy, a DFA $\bar{\mathcal{A}}^f_s$ is constructed. This procedure has been adapted from [63].

Safety tasks admitting a common CBC and controller are combined together in a single partition set. First, we define the partition set

$$\gamma_{(q,q',\Delta(q'))} := \{(q, q', q'', T) \in \mathcal{P}_M(\bar{\mathcal{A}}^f) \,\big|\, q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\},$$

where for any state $q \in Q$, $\Delta(q)$ is the set of states that can be reached from q in one transition. For each partition set $\gamma_{(q,q',\Delta(q'))}$, we denote its corresponding CBC and controller as $\mathbb{B}_{\gamma_{(q,q',\Delta(q'))}}(x)$ and $\varpi_{\gamma_{(q,q',\Delta(q'))}}(x)$, respectively. For all safety tasks corresponding to $\vartheta \in \mathcal{P}_M(\bar{\mathcal{A}}^f)$, we therefore have

$$\mathbb{B}_\vartheta(x) = \mathbb{B}_{\gamma_{(q,q',\Delta(q'))}}(x) \text{ and } \varpi_\vartheta(x) = \varpi_{\gamma_{(q,q',\Delta(q'))}}(x), \text{ if } \vartheta \in \gamma_{(q,q',\Delta(q'))}.$$

This results in a switching controller, where multiple locations on the automaton dictate different controllers. For the DFA $\bar{\mathcal{A}}^f = (Q, q_0, \mathcal{AP}, \delta, \bar{F})$ with $\bar{F} = Q \backslash F$, the corresponding DFA representing switching mechanism is given by DFA $\bar{\mathcal{A}}^f_s = (Q_s, q_{0s}, \mathcal{AP}_s, \delta_s, F_s)$ where $Q_s := q_{0s} \cup \{(q, q', \Delta(q')) \mid q, q' \in Q \backslash \bar{F}\} \cup \bar{F}$ is the set of states, $q_{0s} := (q_0, \Delta(q_0))$ is the initial state, $\mathcal{AP}_s = \mathcal{AP}$ is the set of atomic propositions and $F_s = \bar{F}$ is the set of final states. The transition function δ_s is defined as

- for $q_{0s} = (q_0, \Delta(q_0))$,
  - $\delta_s((q_0, \Delta(q_0)), \sigma_{(q_0,q'_0)}) = (q_0, q'_0, \Delta(q'_0))$ where $q'_0 \in \Delta(q_0)$,
- for all $q_s = (q, q', \Delta(q')) \in Q_s \backslash (q_{0s} \cup \bar{F})$,

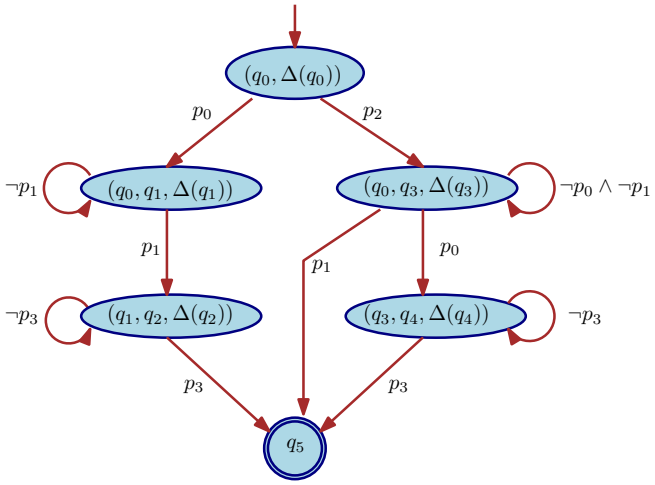


Figure 5.2: DFA $\bar{\mathcal{A}}_s^f$ representing the switching mechanism.

- $\delta_s((q, q', \Delta(q')), \sigma_{(q',q'')}) = (q', q'', \Delta(q''))$, where $q, q', q'' \in Q$, $q'' \in \Delta(q')$ and $q'' \notin \bar{F}$, and
- $\delta_s((q, q', \Delta(q')), \sigma_{(q',q'')}) = q''$ where $q, q', q'' \in Q$, $q'' \in \Delta(q')$ and $q'' \in \bar{F}$.

Now, the controller for Problem 7 is formally given by

$$\tilde{\varpi}(x, q_s) = \tilde{\varpi}_{\gamma_{q_s'}}(x), \quad \forall (q_s, L(x), q_s') \in \delta_s. \tag{5.4}$$

Example 5 (Continued)**.** *The DFA $\bar{\mathcal{A}}_s^f$ representing the switching mechanism between controllers for Example 5 is shown in Figure 5.2.*

Probability Computation

For each individual safety task obtained from $\vartheta = (q, q', q'', T_h) \in \mathcal{P}_M(\bar{\mathcal{A}}^f)$, we first compute upper bounds on the safety violation and then combine them to provide an upper bound on the probability that the specification represented by the language of DFA $\mathcal{A}^f$ is violated, which is provided by the following theorem.

Theorem 13. *For a specification given by the accepting language of DFA $\mathcal{A}^f$, let $\bar{\mathcal{A}}^f$ represent the complement of $\mathcal{A}^f$. For $\bar{\mathcal{A}}^f$, let $\mathcal{R}_M^p$ be the set of all accepting state runs of the length of at most $M+1$ and $\mathcal{P}^p(\mathbf{q})$ be the set of state runs of length 3 augmented with the horizon T_h for $p \in \mathcal{AP}$. Then the probability that the solution processes of dt-SCS $\mathfrak{S}$ starting from any initial state $x_0 \in L^{-1}(p)$ satisfy the specification represented by $\bar{\mathcal{A}}^f$ under the controller in* (5.4) *within the time horizon $[0, M) \subseteq \mathbb{N}$ is upper bounded by*

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,M}) \models \bar{\mathcal{A}}^f\} \leq \sum_{\mathbf{q} \in \mathcal{R}_M^p} \prod_{\vartheta \in \mathcal{P}^p(\mathbf{q})} \{\varepsilon_{\vartheta,T_h} \mid \vartheta = (q, q', q'', T_h) \in \mathcal{P}^p(\mathbf{q})\}, \tag{5.5}$$

where $\varepsilon_{\vartheta T_h}$ is obtained using Lemma 9 and is the upper bound on the probability that solution processes of the system $\mathfrak{S}$ start from $X_0 := L^{-1}(\sigma_{X_0})$ and reach $X_u := L^{-1}(\sigma_{X_u})$ within the time horizon $[0, T_h) \subseteq \mathbb{N}$, where $(\sigma_{X_0}, \sigma_{X_u})$ is the safety task corresponding to ϑ.

Proof. Consider a set of accepting state runs $\mathcal{R}_M^p$ of the length of at most $M+1$ for all $p \in \mathcal{AP}$ and set $\mathcal{P}^p(\mathbf{q})$ as the set of state runs of a length 3, augmented with the horizon T_h. For $\vartheta = (q, q', q'', T_h) \in \mathcal{P}^p(\mathbf{q})$, we can establish from Lemma 9 that the upper bound on the probability that a solution process of dt-SCS $\mathfrak{S}$ starts from $X_0 = L^{-1}(\sigma_{X_0})$ and reaches $X_u = L^{-1}(\sigma_{X_u})$ within the time horizon $[0, T_h) \subseteq \mathbb{N}$ under the influence of the controller ϖ_ϑ is given by $\varepsilon_{\vartheta T_h}$. Then the probability that the solution process reaches the accepting state by following the finite trace corresponding to $\mathbf{q}$ is the product of all probability bounds corresponding to elements $\vartheta = (q, q', q'', T_h) \in \mathcal{P}^p(\mathbf{q})$ and is given by

$$\mathbb{P}(L(\mathbf{x}_{x_0,\varpi,M}) \models \bar{\mathcal{A}}^f) \leq \prod_{\vartheta \in \mathcal{P}^p(\mathbf{q})} \{\varepsilon_{\vartheta T_h} \,\big|\, \vartheta = (q, q', q'', T_h) \in \mathcal{P}^p(\mathbf{q})\}.$$

Using the horizon T_h, we obtain the upper bound on probabilities for accepting state runs $\mathcal{R}_M^p$ with a length of at most $M+1$ by considering all possible self-loop combinations. Given the initial condition $x_0 \in L^{-1}(p)$, the final upper bound for a solution process of $\mathfrak{S}$ to violate the required specification is essentially the summation of probabilities of all possible accepting state runs from the initial state to the final state of $\bar{\mathcal{A}}^f$, and is given by

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,M}) \models \bar{\mathcal{A}}^f\} \leq \sum_{q \in \mathcal{R}_M^p} \prod_{\vartheta \in \mathcal{P}^p(\mathbf{q})} \{\varepsilon_{\vartheta T_h} \,\big|\, \vartheta = (q, q', q'', T_h) \in \mathcal{P}(\mathbf{q})\},$$

which completes the proof. □

In the following corollary, we provide the formula for computing the lower bound on the probability that the interconnected system $\mathfrak{S}$ satisfies the desired specification represented by the DFA $\mathcal{A}$. This is the final result of the section providing a solution to Problem 7.

Corollary 3. *The probability that the solution processes of (interconnected) stochastic control system $\mathfrak{S}$ start from any initial state $x_0 \in L^{-1}(p)$ and satisfy the specification given by the accepting language of DFA $\mathcal{A}^f$ over a finite-time horizon $[0, M) \subseteq \mathbb{N}$ is lower bounded by*

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,M}) \models \mathcal{A}^f\} \geq 1 - \sum_{\mathbf{q} \in \mathcal{R}_M^p} \prod_{\vartheta \in \mathcal{P}^p(\mathbf{q})} \{\varepsilon_{\vartheta T_h} | \vartheta = (q, q', q'', T_h) \in \mathcal{P}^p(\mathbf{q})\}. \tag{5.6}$$

5.2.4 Case Study

For our case study, we apply our results to the large-scale Kuramoto oscillator network presented in Section 3.3.4. In particular, the results in Section 3.3.4 were limited to safety specifications. In this section, we extend the controller synthesis problem for a specification that can be expressed

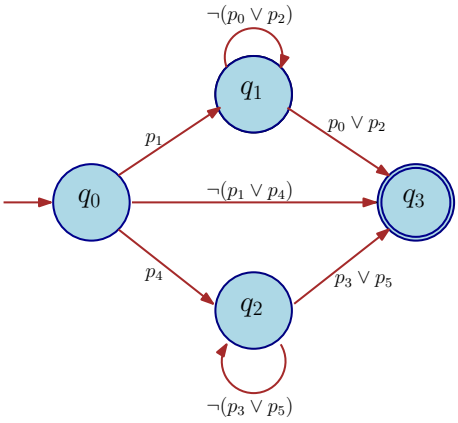


Figure 5.3: DFA $\bar{\mathcal{A}}^f$ representing the complement of specification.

as a deterministic finite automaton. For the dynamics of the system, we refer to Figure 3.3 (©2022 IEEE). Note that $\theta = [\theta_1; \ldots; \theta_N]$ is the phase of oscillators with $\theta_i \in [0, 2\pi]$, $\forall i \in \{1, \ldots, N\}$, $\Omega = [\Omega_1; \ldots; \Omega_N] = [0.01; \ldots; 0.01]$ is the natural frequency of oscillators, $K = 0.0012$ is the coupling strength, $\tau = 0.1$ is the sampling time, $\phi(\theta(t)) = [\phi(\theta_1(t)); \ldots; \phi(\theta_N(t))]$ such that $\phi(\theta_i(t)) = \sum_{\substack{j=1 \\ i \neq j}} sin(\theta_j(t) - \theta_i(t)), \forall i \in \{1, \ldots, N\}$, $\nu(t) = [\nu_1(t); \ldots; \nu_N(t)]$, and $\varsigma(t) = [\varsigma_1(t); \ldots; \varsigma_N(t)]$. Regions of interest are given by $X^0 = [0, \frac{\pi}{15}]^N, X^1 = [\frac{4\pi}{9}, \frac{5\pi}{9}]^N, X^2 = [\frac{14\pi}{15}, \pi]^N, X^3 = [\pi, \frac{16\pi}{15}]^N, X^4 = [\frac{13\pi}{9}, \frac{14\pi}{9}]^N, X^5 = [\frac{29\pi}{15}, 2\pi]^N$ and $X^6 = X \backslash (X^0 \cup X^1 \cup X^2 \cup X^3 \cup X^4 \cup X^5)$. Each region is associated with an element of the atomic proposition given by $\mathcal{AP} = \{p_0, p_1, p_2, p_3, p_4, p_5, p_6\}$ such that the labeling function $L(x_l) = p_l$ for all $x_l \in X^l, l \in \{0, 1, \ldots, 6\}$.

The main goal is to compute a controller such that if the system starts from X^1, it must always stay away from X^0 and X^2, and if it starts from X^4, it must always stay away from X^3 and X^5 within the time horizon $[0, T_d) \subseteq \mathbb{N}$, with time horizon $T_d = 7$. Such a property can be represented as an LTL$_f$ specification given by $(p_1 \wedge \Box\neg(p_0 \vee p_2)) \vee (p_4 \wedge \Box\neg(p_3 \vee p_5))$ over the finite time horizon of $T_d = 7$. It can also be represented by the accepting language of a DFA. Figure 5.3 shows the complement DFA $\bar{\mathcal{A}}^f$. The DFA $\mathcal{A}^f$ representing the original specification can be readily obtained by switching the non-accepting and accepting states in the figure. We first begin by decomposing the complement of the specification into simple safety tasks. We consider accepting state runs without self-loops with $M = 7$. The DFA $\bar{\mathcal{A}}^f$ has three such accepting state runs and $\mathcal{R}_M = \{(q_0, q_3), (q_0, q_1, q_3), (q_0, q_2, q_3)\}$. For all $p \in \mathcal{AP}$, we have $\mathcal{R}_M^{p_0} = \mathcal{R}_M^{p_2} = \mathcal{R}_M^{p_3} = \mathcal{R}_M^{p_5} = \mathcal{R}_M^{p_6} = \{(q_0, q_3)\}$, $\mathcal{R}_M^{p_1} = \{(q_0, q_1, q_3)\}$, and $\mathcal{R}_M^{p_4} = \{(q_0, q_2, q_3)\}$. Sets $\mathcal{P}^p(\mathbf{q})$ can be obtained for each of these accepting state runs as $\mathcal{P}^{p_1}(q_0, q_1, q_3) = \{(q_0, q_1, q_3, 6)\}$ and $\mathcal{P}^{p_2}(q_0, q_2, q_3) = \{(q_0, q_2, q_3, 6)\}$. Note that since $\mathbf{q} = (q_0, q_3)$ is a state run of a length 2, it admits a trivial probability as mentioned in Remark 31, and therefore, it can be neglected. Consequently, we need to find control barrier certificates and corresponding controllers for the remaining two safety tasks $\vartheta_1 = (q_0, q_1, q_3, 6)$ and $\vartheta_2 = (q_0, q_2, q_3, 6)$, namely $(p_1, (p_0 \vee p_2))$ and $(p_4, (p_0 \vee p_5))$, respectively.

Note that since the considered system is a large-scale one, computing control barrier certifi-

Table 5.1: CSBC, controller, and parameters obtained for safety tasks ϑ for all $1 \leq i \leq N$ subsystems.

ϑ	$\mathbb{B}_i(\theta_i)$	$\varpi_{i\vartheta}(\theta_i)$	η_i	β_i	c_i	$\alpha_i(s)$	$\kappa_i(s)$	$\rho_i(s)$	$\varrho_i(s)$
$(q_0, q_1, q_3, 6)$	$0.001361\theta_i^8$ – $0.0001877\theta_i^7$ + $0.0004904\theta_i^6$ – $0.03395\theta_i^5$ + $0.00107\theta_i^4 - 0.1927\theta_i^3 + 1.71\theta_i^2 - 3.205\theta_i + 1.827$	$-0.532\theta_i^2 + 1.69$	0.02	1.2	0.0083	$4.7 \times 10^{-7}s$	$0.997s$	$4.49 \times 10^{-7}s$	s
$(q_0, q_2, q_3, 6)$	$0.5396\theta_i{}^2 - 5.086\theta_i + 11.86$	$-0.21\theta_i^2 + 4.6591$	0.017	1	0.0162	$4.5 \times 10^{-8}s$	$0.998s$	$4.49 \times 10^{-8}s$	s

cates monolithically and providing probability guarantees for the safety tasks via Corollary 1 is not scalable. Therefore, we utilize the compositional framework based on the small gain theorem, as provided in Section 3.3 in Chapter 3 to compute control barrier certificates. To do so, we consider the network of N nonlinear oscillators as an interconnection of N subsystems, *i.e.*, $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_N)$ where each subsystem $\mathfrak{S}_i, i \in \{1, \ldots, N\}$, can be described by dynamics as shown in Figure 3.3 (©2022. To compute control sub-barrier certificates and the corresponding local controllers, we utilize the SOS algorithm in Section 3.3.3 and in particular, we use SOSTOOLS and SDP solver SeDuMi. Since dynamics of $\mathfrak{S}$ are not polynomial and SOS algorithm is only equipped to provide solutions for polynomial dynamics, we make an approximation to our dynamics, as already explained in the case study in Section 3.3.4.

The CSBC, local controller, and other parameters for the safety tasks corresponding to ϑ_1 and ϑ_2 are shown in Table 5.1 (©2022 IEEE). For both tasks, we utilize results from Section 3.3 to show that CBC can be constructed compositionally. Then, by Lemma 9, we correspondingly obtain upper bounds for reaching states corresponding to $p_0 \vee p_2$ and $p_3 \vee p_5$ from p_1 and p_4, respectively. These values are reported in Table 5.2 (©2022 IEEE). The switching mechanism for controllers is obtained as described in Subsection 5.2.3. Now, by employing Theorem 13 and Corollary 3, we obtain the lower bound on the probability that the solution processes of the interconnected system $\mathfrak{S}$ start from an initial state $x_0 \in X^1$ and satisfy the specification represented by the language of DFA $\mathcal{A}$ within the time horizon $T_d = 7$ as

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,7}) \models \mathcal{A}\} \geq 0.94.$$

Similarly, for the solution processes of the interconnected system $\mathfrak{S}$ starting from $x_0 \in X^4$, we acquire

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi,7} \models \mathcal{A}\} \geq 0.9.$$

Figure 5.4 (©2022 IEEE) shows the evolution of solution processes within the time horizon $T_d = 7$ when starting from initial regions of X_1 and X_4. The CSBC computation for ϑ_1 takes 1 minute with a memory usage of 30 MB and for ϑ_2, it takes 20 seconds and 1 MB memory on a Microsoft Windows machine (Intel i7-8665U CPU with 32 GB of RAM).

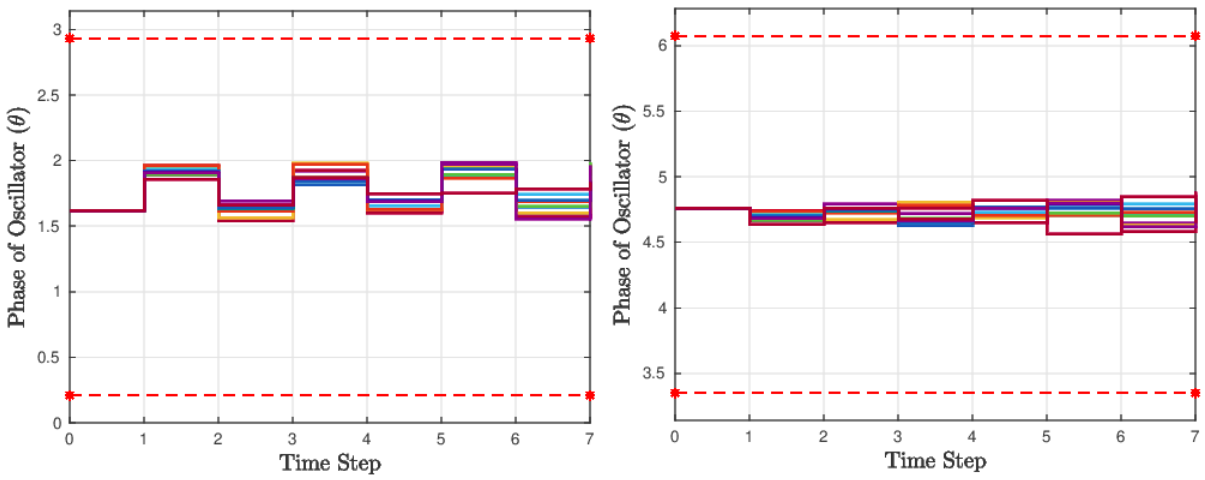


Figure 5.4: Closed-loop state trajectories of a representative oscillator in a network of 100 oscillators with 10 noise realizations with an initial state starting from (left) X_1, and (right) X_4.

Table 5.2: CBC, controller, and probabilistic guarantees obtained for safety tasks ϑ for the interconnected system.

ϑ	$\mathbb{B}(\theta)$	$\varpi_\vartheta(\theta)$	η	β	c	$\kappa(s)$	$\varkappa_{\vartheta T_h}$
$(q_0, q_1, q_3, 6)$	$\max_i\{0.001361\theta_i^8 - 0.0001877\theta_i^7 + 0.0004904\theta_i^6 - 0.03395\theta_i^5 + 0.00107\theta_i^4 - 0.1927\theta_i^3 + 1.71\theta_i^2 - 3.205\theta_i + 1.827\}$	$[-0.532\theta_1^2+1.69;\ldots;-0.532\theta_{100}^2+1.69\theta_{100}]$	0.02	1.2	0.0083	0.997^s	0.0568
$(q_0, q_2, q_3, 6)$	$\max_i\{0.5396{\theta_i}^2 - 5.086\theta_i + 11.86\}$	$[-0.21\theta_1^2+4.6591;\ldots;-0.21\theta_{100}^2+4.6591\theta_{100}]$	0.017	1	0.0162	0.998^s	0.109

5.3 Controller Synthesis for Stochastic Control Systems against ω-Regular Properties

In the last section, we considered the controller synthesis procedure for (interconnected) stochastic control systems against specifications that can be expressed by deterministic finite automata. It must be noted that such specifications are inherently defined over finite time horizons. However, for reactive systems such as medical devices and power grids, it is essential to provide long-term guarantees over infinite time horizons. Therefore, it becomes necessary to express specifications using ω-regular languages. Such specifications can be expressed by ω-automata that can recognize infinite traces, such as non-deterministic Büchi automata [23], deterministic Rabin automata [101], deterministic Streett automata [118], parity automata or Muller automata [86]. While the above-mentioned automata have different acceptance conditions, they have the same expressive power and all of them recognize ω-regular languages. Here, we use deterministic Streett automata (DSA) to describe ω-regular properties and provide a controller synthesis procedure for (interconnected) stochastic control systems against specifications represented as DSA. Similar to Section 5.3, in this section, we provide a controller synthesis procedure for such specifications by

decomposing the automata into a set of safety tasks and utilizing CBCs to provide probabilistic guarantees of these tasks. Then, we combine these guarantees to obtain lower bounds on the probability that the system satisfies the original DSA specification.

5.3.1 Probem Definition

First, recollect from Section 2.4 that DSA $\mathcal{A}^s = (Q, q_0, \Sigma, \delta, Acc)$ is a tuple consisting of a set of states, an initial state, the alphabet Σ, deterministic transition function δ and an accepting condition that is given by a pair of states, *i.e.*, $\{< E_1, F_1 >, < E_2, F_2 >, \ldots, < E_z, F_z >\}$, where $< E_i, F_i >$ with $E_i, F_i \subseteq Q, \forall i \in \{1, \ldots, z\}$. Moreover, we define we $E = \{E_1, E_2, \ldots, E_z\}$ and $F = \{F_1, F_2, \ldots, F_z\}$ where $< E_i, F_i > \in Acc$, $\forall i \in \{1, \ldots, z\}$. Finally, an infinite run $\mathbf{q} = (q_0, q_1, \ldots)$ is said to be an accepting run for $\mathcal{A}^s$ if for all $E_i \in E$ and $F_i \in F$, $i \in \{1, \ldots, z\}$, we have $inf(\mathbf{q}) \cap E_i = \varnothing$ or $inf(\mathbf{q}) \cap F_i \neq \varnothing$. The corresponding trace $\sigma(\mathbf{q})$ is said to be accepted by the DSA $\mathcal{A}$, denoted by $\sigma(\mathbf{q}) \models \mathcal{A}^s$. The language of $\mathcal{A}$, denoted by $\mathcal{L}(\mathcal{A}^s)$, comprises all the traces accepted by $\mathcal{A}^s$. We consider specifications expressed by accepting languages of DSA $\mathcal{A}^s$ when input symbols are defined over a set of atomic propositions $\mathcal{AP}$ as the alphabet, *i.e.*, $\Sigma = 2^{\mathcal{AP}}$. However, as seen in Remark 34, without any loss of generality, we can consider that $\Sigma = \mathcal{AP}$. Moreover, specifications expressed as linear temporal logic (LTL) formulae can be represented by DSA with the help of existing tools like `ltl2dstar` [68].

Remark 33. *A DSA $\mathcal{A}$ with the set $E = \varnothing$ accepts any infinite run. Therefore, without loss of generality, we assume that the set E is non-empty.*

It should be noted that while deterministic Büchi automata are also a class of ω-automata and are used for representing languages over infinite words, their expressive power is strictly weaker than other classes of ω-automata such as non-deterministic Büchi, deterministic Streett or Rabin automata. It is worth mentioning that if the verification is the main objective, one can utilize both deterministic and non-deterministic Büchi automata. The latter is preferred because it has higher expressive power, and accordingly, it can represent ω-regular properties. However, in our work, we deal with the controller synthesis problem where the determinism of automata is crucial, and one cannot directly work with non-deterministic Büchi automata (NBA). In this case, one needs to determinize the automata without losing their expressiveness. Although the idea of this work can be applied to less expressive automata including deterministic Büchi automata, we prefer to deal with the full class of LTL properties. Hence, we work with deterministic Streett automata since they have the same expressive power as NBA.

Similar to Section 5.2, our synthesis procedure relies on the decomposition of the automata into simple safety tasks and utilizing the probabilistic guarantees for safety and combining them to provide guarantees for the original specification. Note that we obtain the controller synthesis for safety tasks using supermartingale barrier certificates as in Definition 8 or Definition 17 since they suitably provide probability guarantees over infinite time horizons. To do so, we first define how ω-regular properties are connected to solution processes of the (interconnected) dt-SCS $\mathfrak{S}$ via the labeling function $L : X \to \mathcal{AP}$.

Definition 26. *For an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$ and a DSA $\mathcal{A}^s = (Q, q_0, \mathcal{AP}, \delta,$ Acc), consider a labeling function $L : X \to \mathcal{AP}$. For an infinite-state sequence $\mathbf{x} = (\mathbf{x}(0), \mathbf{x}(1), \ldots) \in X^\omega$, the corresponding trace over $\mathcal{AP}$ is given by $L(\mathbf{x}) := (\sigma_0, \sigma_1, \ldots) \in \mathcal{AP}^\omega$, where $\sigma_i = L(\mathbf{x}(i))$ for all $i \in \mathbb{N}$.*

Similar to the previous results of Section 5.2, we consider that the set of atomic propositions $\mathcal{AP} = \{p_0, p_1, \ldots, p_R\}, R \in \mathbb{N}$ provides a measurable partition of the state space $X = \bigcup_{i=1}^{R} X^i$ via the labeling function $L : X \to \mathcal{AP}$ such that $L(x \in X^i) = p_i$. Without loss of generality, it can be assumed that $X^i \neq \emptyset$ for any $i \in \{1, \ldots, R\}$.

Remark 34. *Since $\mathcal{AP}$ provides a measurable partition of the state set X, for any two atomic propositions $p_i, p_j \in \mathcal{AP}$, $i \neq j$, $i, j \in \{1, \ldots, R\}$, we have that $p_i \wedge p_j = \emptyset$. Therefore, while constructing the DSA $\mathcal{A}$ corresponding to the required specification, one can remove the edges with $p_i \wedge p_j$ as they are infeasible. Moreover, other Boolean combinations of atomic propositions may also be resolved. For example, an edge with $p_i \vee p_j$ can be resolved by adding two new edges with p_i and p_j, respectively. The negation $\neg p_i$ can also be handled in a similar fashion. Therefore, we assume in the remainder of the paper that the alphabet Σ is defined directly over the set of atomic propositions rather than its power set, i.e., $\Sigma = \mathcal{AP}$.*

We now define the probability with which solution processes of interconnected dt-SCS $\mathfrak{S}$ satisfy an ω-regular specification represented by DSA $\mathcal{A}^s$.

Definition 27. *Consider an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$, a specification given by the accepting language of the DSA $\mathcal{A}^s = (Q, q_0, \mathcal{AP}, \delta, Acc)$ and a labeling function $L : X \to \mathcal{AP}$. Then the probability with which the solution process $\mathbf{x}_{x_0,\varpi}$ under the controller ϖ with an initial condition $x(0) = x_0$ satisfies the specification expressed by $\mathcal{A}^s$ is given by $\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi}) \models \mathcal{A}^s\}$.*

To tackle the controller synthesis problem, we must obtain a suitable controller in conjunction with a lower bound on the probability that the (interconnected) dt-SCS satisfies a specification described by DSA. We now formally state the problem considered in this section as follows.

Problem 8. *Given an (interconnected) dt-SCS $\mathfrak{S} = (X, U, \varsigma, f)$, a specification represented by the accepting language of a DSA $\mathcal{A}^s = (Q, q_0, \mathcal{AP}, \delta, Acc)$ over a set of atomic propositions $\mathcal{AP} = \{p_0, p_1 \ldots, p_R\}$, $R \in \mathbb{N}$, and a labeling function $L : X \to \mathcal{AP}$, compute a controller ϖ (if existing) and a constant $\varkappa \in [0, 1]$ such that $\mathbb{P}\{L(\boldsymbol{x}_{x_0,\varpi}) \models \mathcal{A}\} \geq \varkappa$.*

To tackle this problem for a given specification represented by the DSA, we decompose the DSA into a set of sequential safety tasks such that the satisfaction of these smaller tasks leads to the satisfaction of the original DSA. Then, our problem is reduced to finding CBCs and appropriate controllers for each of these safety tasks. Note that, similar to the results in Section 5.2, one may provide similar results for both smaller stochastic control systems by utilizing control barrier certificates as in Definition 8, or for interconnected stochastic control systems by compositionally constructing control barrier certificates as in Definition 17. However, to apply the results on the latter, one requires some mild assumptions on the considered sets corresponding to the atomic propositions. In particular, let sets X_0 and X_u, as introduced in Definition 17, be connected to

the atomic propositions in $\mathcal{AP}$ via some labeling function $L : X \to \mathcal{AP}$. We assume that these sets can be decomposed as $X_0 = \prod_{i=1}^{N} X_{0_i}$ and $X_u = \prod_{i=1}^{N} X_{u_i}$. We also assume that the atomic propositions in $\mathcal{AP}$ can also be decomposed similarly. This allows connecting the sets X_{0_i} and X_{u_i}, $i \in \{1, \ldots, N\}$, to the corresponding decomposed structure of $\mathcal{AP}$, and consequently, one can apply the results of Section 3.4 by computing CSBCs for each subsystem separately and then combining them to obtain CBC for the interconnected system compositionally.

In the following section, we describe the automata decomposition procedure to decompose the DSA into smaller safety tasks.

5.3.2 Specification Decomposition

In order to facilitate controller synthesis for general ω-regular specifications represented by accepting languages of DSA, we use a divide-and-conquer method and reduce the automaton to a set of simple sequential safety tasks. This method is slightly similar to the one proposed in Section 5.2 for decomposing specifications represented by DFA by considering the complement of the specifications. However, in our case, we directly deal with the specifications represented by DSA without requiring any complementation. Moreover, as mentioned earlier, we are dealing here with infinite time horizons rather than finite time horizons. We now briefly describe the decomposition procedure.

A specification represented by DSA $\mathcal{A}^s = (Q, q_0, \mathcal{AP}, \delta, Acc)$ is said to be satisfied by a dt-SCS $\mathfrak{S}$ if the traces $L(\mathbf{x}_{x_0,\varpi})$ corresponding to the solution processes $\mathbf{x}_{x_0,\varpi}$ of $\mathfrak{S}$ are accepted by $\mathcal{A}^s$. Note that this is the case when the corresponding runs of the form $\mathbf{q} = (q_0, q_1, \ldots) \in Q^\omega$ satisfy the following condition: for all $E_i \in E$ and $F_i \in F$, $i \in \{1, \ldots, z\}$, $inf(\mathbf{q}) \cap E_i = \varnothing$ or $inf(\mathbf{q}) \cap F_i \neq \varnothing$. Note that satisfying $inf(\mathbf{q}) \cap E_i = \varnothing$, for all $E_i \in E$, automatically implies the satisfaction of the original acceptance condition of the DSA $\mathcal{A}^s$. We refer to this as the partial acceptance condition of the DSA. Moreover, we call an infinite run $\bar{\mathbf{q}} = (q_0, q_1, \ldots)$ a partially accepting state run iff for all $E_i \in E$, we have $inf(\bar{\mathbf{q}}) \cap E_i = \varnothing$. Now, to decompose the DSA $\mathcal{A}^s$ into consecutive safety tasks, we first obtain all *partially accepting lasso runs* (or simply, *lassos*) of $\mathcal{A}^s$. Such a lasso consists of a simple (*i.e.* without self-loops) finite path from the initial state $q_0 \in Q$ to a state in E, concatenated with a simple finite cycle from the state in E to itself. Then, formally, a lasso is a pair $\tilde{\mathbf{q}} = (\tilde{\mathbf{q}}_f, \tilde{\mathbf{q}}_l)$ such that $\tilde{\mathbf{q}}_f = (q_0^f, q_1^f, \ldots, q_{a_f}^f, q_0^l)$ represents the finite path and $\tilde{\mathbf{q}}_l = (q_0^l, q_1^l, \ldots, q_{a_l}^l, q_0^l)$ represents the finite cycle, where $a_f, a_l \in \mathbb{N}$, $q_0^f = q_0$ and $q_0^l \in E$. Note that the number of such lassos for the DSA $\mathcal{A}^s$ is finite since $\mathcal{A}^s$ consists of finite numbers of states and edges. Let $\mathcal{R}$ be the set of all such lassos, and $\mathcal{R}_f$ and $\mathcal{R}_l$ be sets containing only finite paths $\tilde{\mathbf{q}}_f$ and $\tilde{\mathbf{q}}_l$, respectively. Now, for each $p \in \mathcal{AP}$, we define a set $\mathcal{R}^p$ as

$$\mathcal{R}^p := \Big\{\tilde{\mathbf{q}} = (\underbrace{q_0^f, q_1^f, \ldots, q_{a_f}^f}_{\tilde{\mathbf{q}}_f}, \overbrace{q_0^l}^{\tilde{\mathbf{q}}_l}, q_1^l, \ldots, q_{a_l}^l, q_0^l) \in \mathcal{R} \,\Big|\, q_1^f = \delta(q_0^f, p), p \in \mathcal{AP}\Big\}. \tag{5.7}$$

Similarly, sets $\mathcal{R}_f^p$ and $\mathcal{R}_l^p$ are defined for simple finite paths and simple finite cycles, respectively. Now, in order to perform decomposition into safety tasks, we define a set $\mathcal{P}^p(\tilde{\mathbf{q}})$ for any $\tilde{\mathbf{q}} =$

$(q_0, q_1, \ldots, q_{a_f+a_l+3}) \in \mathcal{R}^p$ as

$$\mathcal{P}^p(\tilde{\mathbf{q}}) = \big\{(q_i, q_{i+1}, q_{i+2}) \,\big|\, 0 \leq i \leq a_f + a_l + 1\big\}. \tag{5.8}$$

We correspondingly define the set $\mathcal{S}^p(\tilde{\mathbf{q}})$ to be the set of all *consecutive transition pairs* $(\sigma_{X_0}, \sigma_{X_u}) \in \mathcal{AP}$ such that

$$\mathcal{S}^p(\tilde{\mathbf{q}}) = \{(\sigma_{X_0}, \sigma_{X_u}) \mid q_{i+1} = \delta(q_i, \sigma_{X_0}), q_{i+2} = \delta(q_{i+1}, \sigma_{X_u}), (q_i, q_{i+1}, q_{i+2}) \in \mathcal{P}^p(\tilde{\mathbf{q}})\}. \tag{5.9}$$

Consecutive transition pairs in $\mathcal{S}^p(\mathbf{q})$ corresponding to elements $\mathcal{P}^p(\mathbf{q})$ are referred to as safety tasks. Consequently, we define $\mathcal{P}^p(\tilde{\mathbf{q}}_f)$ and $\mathcal{P}^p(\tilde{\mathbf{q}}_l)$ to comprise the elements from simple finite paths and cycles $\tilde{\mathbf{q}}_f$ and $\tilde{\mathbf{q}}_l$, respectively, for each $p \in \mathcal{AP}$. Finally, we define $\mathcal{P}(\mathcal{A})^s = \bigcup_{p\in\mathcal{AP}} \bigcup_{\tilde{\mathbf{q}}\in\mathcal{R}^p} \mathcal{P}^p(\tilde{\mathbf{q}})$ as the set of all such elements obtained from the DSA $\mathcal{A}^s$.

Remark 35. *Note that even though self-loops are ignored while decomposing the DSA $\mathcal{A}^s$ into safety tasks $\vartheta = (q_i, q_{i+1}, q_{i+2}) \in \mathcal{P}^p(\tilde{\mathbf{q}})$, it is crucial to account for the time spent in the self-loops before reaching the state q_{i+2} from q_i. This is automatically accounted for via the construction of control barrier certificates.*

We employ the following example for the sake of better illustration.

Example 6. *We perform safety decomposition for the DSA $\mathcal{A}^s$ shown in Figure 5.5. The figure indicates with an arrow $\rightarrow$ the initial state of the system, while $\bullet$ and $\blacksquare$ indicate the states that can be visited finitely and infinitely many times, respectively. In other words, we have q_0 as initial state, the set of the atomic proposition $\mathcal{AP} = \{p_0, p_1, p_2\}$ and* Acc $=< q_4, q_2 >$ *as the acceptance condition. Therefore, an infinite run $\mathbf{q}$ is accepted if it visits q_4 only finitely often or q_2 infinitely often. In order to decompose the problem into safety tasks, we consider the partially accepting lasso runs of the DSA $\mathcal{A}^s$ and obtain the set $\mathcal{R}$ consisting of all such lassos. This is given by*

$$\begin{aligned}\mathcal{R} = \big\{&(q_0, q_4, q_5, q_4), (q_0, q_3, q_4, q_5, q_4), (q_0, q_1, q_4, q_5, q_4),\\ &(q_0, q_1, q_2, q_4, q_5, q_4)\big\}.\end{aligned}$$

The sets $\mathcal{R}^p$ for $p \in \mathcal{AP}$, are obtained as

$$\begin{aligned}\mathcal{R}^{p_0} &= \big\{(q_0, q_1, q_4, q_5, q_4), (q_0, q_1, q_2, q_4, q_5, q_4)\big\},\\ \mathcal{R}^{p_1} &= \big\{(q_0, q_4, q_5, q_4)\big\}, \quad \mathcal{R}^{p_2} = \big\{(q_0, q_3, q_4, q_5, q_4)\big\}.\end{aligned}$$

Now for each $\tilde{\mathbf{q}} \in \mathcal{R}^p$, we define $\mathcal{P}^p(\tilde{\mathbf{q}})$ as follows:

$$\begin{aligned}&\mathcal{P}^{p_0}(q_0, q_1, q_2, q_4, q_5, q_4) = \big\{(q_0, q_1, q_2), (q_1, q_2, q_4), (q_2, q_4, q_5), (q_4, q_5, q_4)\big\},\\ &\mathcal{P}^{p_0}(q_0, q_1, q_4, q_5, q_4) = \big\{(q_0, q_1, q_4), (q_1, q_4, q_5), (q_4, q_5, q_4)\big\},\\ &\mathcal{P}^{p_1}(q_0, q_4, q_5, q_4) = \big\{(q_0, q_4, q_5), (q_4, q_5, q_4)\big\},\\ &\mathcal{P}^{p_2}(q_0, q_3, q_4, q_5, q_4) = \big\{(q_0, q_3, q_4), (q_3, q_4, q_5), (q_4, q_5, q_4)\big\}.\end{aligned}$$

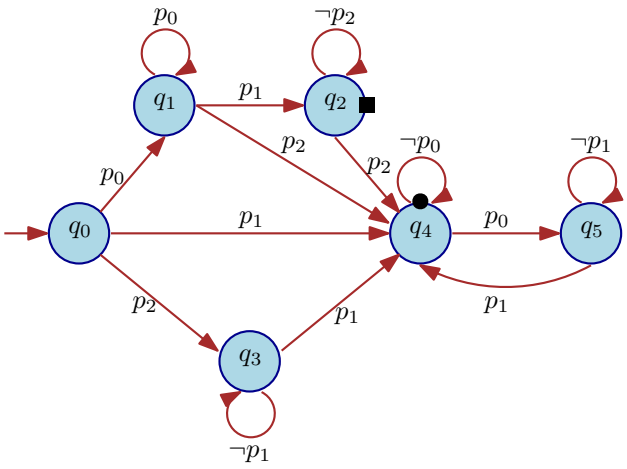


Figure 5.5: DSA $\mathcal{A}^s$ employed in Example 6.

Finite words $\boldsymbol{\sigma}(\tilde{\mathbf{q}})$ *corresponding to* $\tilde{\mathbf{q}} \in \mathcal{R}$ *are obtained as*

$$\boldsymbol{\sigma}(q_0, q_1, q_2, q_4, q_5, q_4) = (p_0, p_1, p_2, p_0, p_1),$$
$$\boldsymbol{\sigma}(q_0, q_1, q_4, q_5, q_4) = (p_0, p_2, p_0, p_1), \quad \boldsymbol{\sigma}(q_0, q_4, q_5, q_4) = (p_1, p_0, p_1),$$
$$\boldsymbol{\sigma}(q_0, q_3, q_4, q_5, q_4) = (p_2, p_1, p_0, p_1).$$

We now propose a systematic procedure utilizing CBC to obtain a suitable controller while computing (preferably maximizing) the lower bound on the probability that interconnected dt-SCS $\mathfrak{S}$ satisfies the specification expressed by the DSA $\mathcal{A}$. To do this, we first consider all the elements in the set $\mathcal{P}(\mathcal{A}^s)$, each of which corresponds to a safety task $(\sigma_{X_0}, \sigma_{X_u})$. We then compute the upper bound on the probability that these safety tasks are violated and then combine them to obtain an overall lower bound on the probability of the satisfaction of the specification given by $\mathcal{A}^s$.

Lemma 10. *For a lasso* $\tilde{\mathbf{q}} \in \mathcal{R}^p$ *with* $p \in \mathcal{AP}$*, consider a safety task* $(\sigma_{X_0}, \sigma_{X_u})$ *corresponding to* $\vartheta = (q, q', q'') \in \mathcal{P}^p(\tilde{\mathbf{q}})$*. If there exists a CBC and a suitable controller* ϖ *with respect to* $X_0 = L^{-1}(\sigma_{X_0})$*,* $X_u = L^{-1}(\sigma_{X_u})$*, then the probability that the solution process of dt-SCS* $\mathfrak{S}$ *with initial condition* $x_0 \in X_0$ *reaches the region* X_u *under* ϖ *is upper bounded by*

$$\mathbb{P}\{\mathbf{x}_{x_0,\varpi} \in X_u \text{ for some } t \in \mathbb{N} \mid a\} \leq \varepsilon_\vartheta, \tag{5.10}$$

where ε_ϑ *is obtained from equation* (2.21) *or equation 3.30.*

Remark 36. *The satisfaction of the specification represented by the DSA* $\mathcal{A}^s$ *requires the disjunction of two different occurrences, i.e., states in E should be visited finitely often or states in F should be visited infinitely often. However, since the disjunction is already satisfied when one of the occurrences holds, the probability of satisfaction of the specification represented by* $\mathcal{A}^s$ *can be ultimately lower bounded by the probability of the states in E being visited only finitely often.*

Therefore, we can ignore states in F and proceed with sequential decomposition only by taking into account states in E. This is tailored to the nature of CBCs which provide safety guarantees and results in some conservatism in our approach.

Remark 37. *For any $\vartheta = (q, q', q'')$, if we have $L^{-1}(\sigma_{X_0}) \cap L^{-1}(\sigma_{X_u}) \neq \emptyset$, then the safety task does not admit any control barrier certificate, and correspondingly the probability of violating the safety task is* 1. *This is due to the nature of CBCs that require the separation of the initial set and the unsafe set (see Definition 8 or Definition 17).*

5.3.3 Controller and Probability Computation

Generally, every safety task obtained $\mathcal{P}(\mathcal{A}^s)$ admits a single CBC and its corresponding controller. However, in a scenario where there is more than one edge emanating from a single state in the automaton, this can result in ambiguities. For this reason, we combine multiple safety tasks into a single partition set and adopt a switching controller dependent on the location in the automaton. The next subsection explains the switching controller in detail. We also discuss the computation of the overall lower bound on the probability that solution processes of the interconnected system satisfy the original specification.

Controller Structure

Consider the DSA $\mathcal{A}^s$ shown in Figure 5.5. Consider two safety tasks corresponding to two elements $\vartheta_1 = (q_0, q_1, q_2)$ and $\vartheta_2 = (q_0, q_1, q_4)$. Ideally, we must compute two different CBCs and controllers for each of these tasks, one for avoiding the region $L^{-1}(p_1)$ and the other for avoiding $L^{-1}(p_2)$ from a *common* initial region $L^{-1}(p_0)$. Since one cannot employ two different controllers simultaneously in the same region of the state space, this issue results in ambiguity while deploying controllers for the closed-loop system. We resolve this issue by combining the two safety tasks into one by simply replacing the set X_u in Lemma 10 with the union of regions corresponding to the alphabet present in all outgoing edges from the common state. To do so, we combine all safety tasks corresponding $\mathcal{P}(\mathcal{A}^s)$ with a common CBC and put them together in a single partition set. Such sets are defined as

$$\gamma_{(q,q',\Delta(q'))} := \left\{(q, q', q'') \in \mathcal{P}(\mathcal{A}^s) \,\middle|\, q, q', q'' \in Q \text{ and } q'' \in \Delta(q')\right\},$$

where $\Delta(q)$ is the set of states that can be reached from a state $q \in Q$. For the partition set $\gamma_{(q,q',\Delta(q'))}$, the corresponding CBC and controller are denoted as $\mathbb{B}_{\gamma_{(q,q',\Delta(q'))}}(x)$ and $\varpi_{\gamma_{(q,q',\Delta(q'))}}(x)$, respectively. For all safety tasks $\vartheta \in \mathcal{P}(\mathcal{A}^s)$, we therefore have

$$\mathbb{B}_\vartheta(x) = \mathbb{B}_{\gamma_{(q,q',\Delta(q'))}}(x) \text{ and } \varpi_\vartheta(x) = \varpi_{\gamma_{(q,q',\Delta(q'))}}(x), \text{ if } \vartheta \in \gamma_{(q,q',\Delta(q'))}.$$

The system admits a switching controller as the control input depending on the state of the automaton. To represent such a switching controller, a new switching automaton $\mathcal{A}^s_s$ is constructed. For the DSA $\mathcal{A}^s = (Q, q_0, \mathcal{AP}, \delta, Acc)$, we represent the corresponding switching mechanism as $\mathcal{A}^s_s = (Q_s, q_{0s}, \mathcal{AP}_s, \delta_s)$ where $Q_s := q_{0s} \cup \{(q, q', \Delta(q')) \mid q, q' \in Q\}$ is the set of states, $q_{0s} := (q_0, \Delta(q_0))$ is the initial state and $\mathcal{AP}_s = \mathcal{AP}$ is the set of atomic propositions. The transition function δ_s is defined as

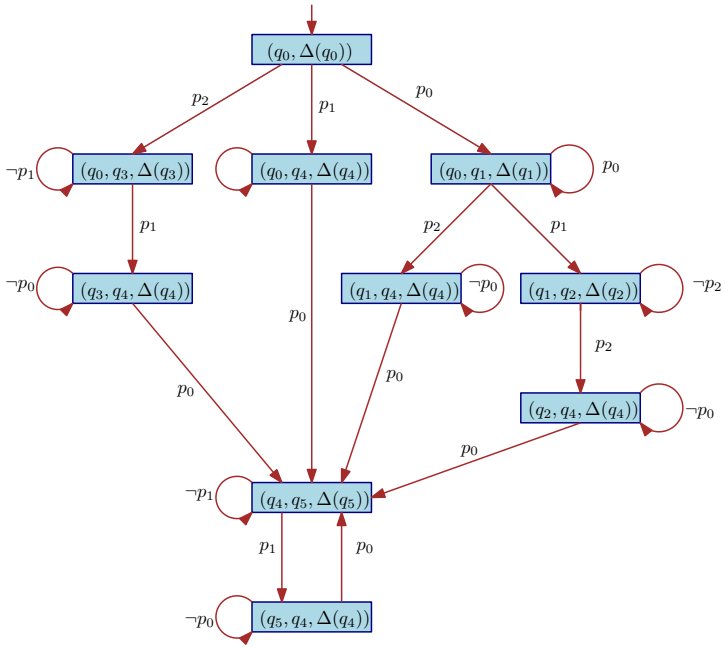


Figure 5.6: Automaton $\mathcal{A}_s^s$ representing switching mechanism.

- for $q_{0s} = (q_0, \Delta(q_0))$, we have $\delta_s((q_0, \Delta(q_0)), \sigma_{(q_0, q_0')}) = (q_0, q_0', \Delta(q_0'))$ such that $q_0' \in \Delta(q_0)$;
- for all $q_s = (q, q', \Delta(q')) \in Q_s \backslash q_{0s}$, we have $\delta_s((q, q', \Delta(q'), \sigma_{(q', q'')}) = (q', q'', \Delta(q''))$ such that $q, q', q'' \in Q$, $q'' \in \Delta(q')$.

Finally, one can obtain the controller for Problem 8 as

$$\tilde{\varpi}(x, q_s) = \tilde{\varpi}_{\gamma_{q_s'}}(x), \quad \forall (q_s, L(x), q_s') \in \delta_s. \tag{5.11}$$

Example 6 (Continued)**.** *The automaton $\mathcal{A}_s^s$ representing the switching controller for Example 6 is shown in Figure 5.6.*

Probability Computation

We now compute the lower bound on the probability that the (interconnected) dt-SCS $\mathfrak{S}$ satisfies the desired specification expressed by the DSA $\mathcal{A}^s$. This is done by first computing the upper bounds on the probability of violating the safety tasks corresponding to $\vartheta = (q, q', q'') \in \mathcal{P}(\mathcal{A}^s)$ using Lemma 10 and combining them to obtain the probability upper bound on visiting the states in E infinitely often. This is then used to compute the probability lower bound on visiting the states in E finitely often, thereby providing the lower bound on the probability with which the (interconnected) dt-SCS $\mathfrak{S}$ satisfies the specification. This is formally explained in the following theorem.

Theorem 14. *For a specification expressed by a DSA $\mathcal{A}^s$, let $\mathcal{R}^p, \mathcal{R}^p_f$, and $\mathcal{R}^p_l$ be all lassos, simple finite paths and simple finite cycles for $p \in \mathcal{AP}$, respectively. Moreover, let $\mathcal{P}^p(\tilde{\boldsymbol{q}}), \mathcal{P}^p(\tilde{\boldsymbol{q}}_f)$, and $\mathcal{P}^p(\tilde{\boldsymbol{q}}_l)$ be derived from $\mathcal{R}^p, \mathcal{R}^p_f$, and $\mathcal{R}^p_l$, respectively. The lower bound on the probability that the solution processes of the dt-SCS $\mathfrak{S}$ start from an initial state $x_0 \in L^{-1}(p)$ and satisfy the specification represented by $\mathcal{A}^s$ is given by*

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi}) \models \mathcal{A}^s\} \geq 1 - \sum_{\tilde{\mathbf{q}} \in \mathcal{R}^p} \prod_{\vartheta \in \mathcal{P}^p(\tilde{\boldsymbol{q}})} \begin{cases} \varepsilon_\vartheta & \vartheta \notin \mathcal{P}^p(\tilde{\boldsymbol{q}}_l), \\ 0 & \vartheta \in \mathcal{P}^p(\tilde{\boldsymbol{q}}_l) \text{ and } \varepsilon_\vartheta < 1, \\ 1 & \vartheta \in \mathcal{P}^p(\tilde{\boldsymbol{q}}_l) \text{ and } \varepsilon_\vartheta \geq 1, \end{cases} \tag{5.12}$$

where ε_ϑ is the probability upper bound obtained for the safety task $(\sigma_{X_0}, \sigma_{X_u})$ corresponding to $\vartheta = (q, q', q'')$ via Lemma 10.

Proof. Consider the set of lassos $\mathcal{R}^p$ and its corresponding set of finite paths and cycles $\mathcal{R}^p_f$ and $\mathcal{R}^p_l$ for all $p \in \mathcal{AP}$. Let the sets $\mathcal{P}^p(\tilde{\mathbf{q}})$, $\mathcal{P}^p(\tilde{\mathbf{q}}_f)$, and $\mathcal{P}^p(\tilde{\mathbf{q}}_l)$ be obtained from these sets, respectively. Following Remark 36, to compute the lower bound on the probability of satisfaction of the specification expressed by DSA $\mathcal{A}^s$, it is sufficient to compute the lower bound on the probability that the states in E are not visited infinitely often. This lower bound can then be computed by first computing the probability upper bound of the states in E visiting infinitely often.

To do this, we consider any $\vartheta = (q, q', q'') \in \mathcal{P}^p(\tilde{\mathbf{q}})$ and its corresponding safety task $(\sigma_{X_0}, \sigma_{X_u})$, and obtain from Lemma 10 the upper bound on the probability that the solution process of dt-SCS $\mathfrak{S}$ starts from $X_0 = L^{-1}(\sigma_{X_0})$ and reaches $X_u = L^{-1}(\sigma_{X_u})$. This is given by ε_ϑ.

In order to compute the upper bound on the probability that the states in E are visited infinitely often, one requires to compute the upper bound on the probability that the solution process follows the lasso $\bar{\mathbf{q}} = (q^f_0, q^f_1, \ldots, q^f_{a_f}, (q^l_0, q^l_1, \ldots, q^l_{a_l})^\omega)$ starting from $X_0 = L^{-1}(\sigma(q^f_0, q^f_1))$, which consists of finite paths $\tilde{\mathbf{q}}_f$ repeated once and the finite cycles $\tilde{\mathbf{q}}_l$ repeated infinitely many times. This is obtained as

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi}) \nvDash \mathcal{A}^s\} \leq \prod_{\vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}})} \begin{cases} \varepsilon_\vartheta & \vartheta \notin \mathcal{P}^p(\tilde{\mathbf{q}}_l), \\ 0 & \vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}}_l) \text{ and } \varepsilon_\vartheta < 1, \\ 1 & \vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}}_l) \text{ and } \varepsilon_\vartheta \geq 1. \end{cases}$$

Now given the initial condition $x_0 \in L^{-1}(p)$, the upper bound for a solution process $\mathbf{x}_{x_0,\varpi}$ of $\mathfrak{S}$ to satisfy the condition of visiting the states in E infinitely many times is basically the summation of probabilities of all possible lassos in $\mathcal{R}^p$, and is obtained by

$$\mathbb{P}\{L(\mathbf{x}^{a\varpi}) \nvDash \mathcal{A}^s\} \leq \sum_{\tilde{\mathbf{q}} \in \mathcal{R}^p} \prod_{\vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}})} \begin{cases} \varepsilon_\vartheta & \vartheta \notin \mathcal{P}^p(\tilde{\mathbf{q}}_l), \\ 0 & \vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}}_l) \text{ and } \varepsilon_\vartheta < 1, \\ 1 & \vartheta \in \mathcal{P}^p(\tilde{\mathbf{q}}_l) \text{ and } \varepsilon_\vartheta \geq 1. \end{cases}$$

Having the probability upper bound for visiting the states in E infinitely often, one obtains the lower bound on the probability of visiting the states in E finitely often, or in other words, the

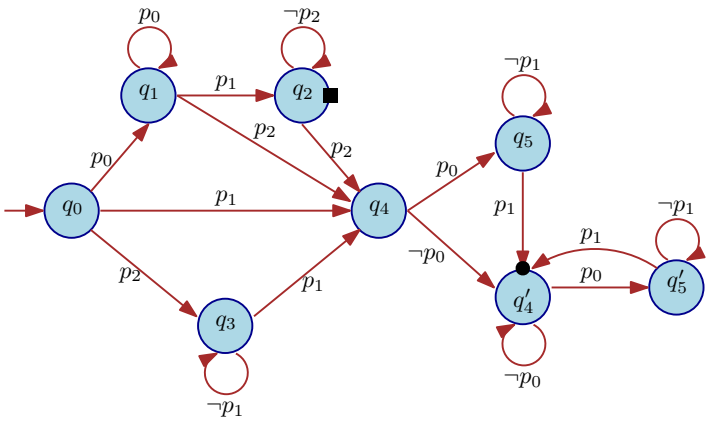


Figure 5.7: Reconstruction of DSA $\mathcal{A}^s$ according to Remark 39.

satisfaction of the specification expressed by DSA $\mathcal{A}^s$ as in inequality (5.12). This completes the proof. □

Remark 38. *Note that if any $\vartheta = (q, q', q'') \in \mathcal{P}^p(\tilde{\boldsymbol{q}}_l)$ and its corresponding safety task admits a CBC, then $\varepsilon_\vartheta < 1$ and correspondingly the probability upper bound of reaching a state in E infinitely many times following the loop $\tilde{\boldsymbol{q}}_l$ is 0. This is because the existence of CBC guarantees that the probability of the loop $\tilde{\boldsymbol{q}}_l$ being taken is less than 1, and correspondingly, the probability of those loops being taken infinitely often becomes 0. However, if no CBC exists for such $\vartheta \in \mathcal{P}^p(\tilde{\boldsymbol{q}}_l)$, then one has to consider CBCs for $\vartheta \notin \mathcal{P}^p(\tilde{\boldsymbol{q}}_l)$. In such a case, the probability of visiting states in E only finitely often is lower bounded by the probability of visiting those states at most once.*

Remark 39. *Note that we only provide probabilistic guarantees for visiting the states in E at most once in the case that the safety tasks in finite cycles do not admit CBCs. This leads to some conservatism in our approach. However, one can also obtain probabilistic guarantees for visiting the states in E at most twice. To do this, a DSA $\mathcal{A}^s$ is reconstructed by duplicating the states in E and the states reachable from the ones in E twice and adding extra transitions to these states such that the language of the reconstructed DSA remains the same. This is illustrated for Example 6 in Figure 5.7, where additional states q'_4 and q'_5 are added by duplicating the states q_4 and q_5 respectively, such that $q'_4 \in E$. Then, by ensuring that q'_4 is visited at most once in the reconstructed DSA, we accordingly provide guarantees for the state q_4 in the original DSA $\mathcal{A}^s$ (Figure 5.5) to be visited at most twice. Note that the formal definition of such reconstruction is omitted for the sake of simple presentation.*

Remark 40. *Note that if a safety task corresponding to $\vartheta \in \mathcal{P}(\mathcal{A}^s)$ does not admit a CBC, the probability lower bound for that safety task is considered to be 0. To obtain potentially a non-trivial probability lower bound for the satisfaction of the original property, at least one safety task should have a suitable CBC.*

Remark 41. *A trivial probability of 0 and an arbitrary controller is possible only in the worst-case scenario where our algorithm fails to compute CBCs for all safety tasks in the DSA. Note*

that safety tasks in the finite cycles of the lassos play a crucial role in obtaining tight lower bounds, as mentioned in Remark 38. Therefore, it is beneficial to first search for suitable CBCs and corresponding controllers for $\vartheta \in \mathcal{P}^p(\tilde{\boldsymbol{q}}_l)$ in order to obtain tight lower bounds.

Remark 42. *While the computation of probabilities of satisfaction for DSA specifications via Theorem 14 requires the enumeration of all the lassos, the maximum number of CBCs and associated probabilities we need to compute depends on the cardinality of the set of atomic propositions. For instance, if the set of atomic propositions has only three elements, we only require computing a maximum of six CBCs independently of the structure of DSA.*

5.3.4 Case Study

For our case study, we extend the results obtained for the safety specification considered in the room temperature network system of Section 3.4.6 for a specification that can be expressed by deterministic Strett automaton. However, for the sake of completeness in this section, we present the system dynamics once again, as

$$\mathfrak{S} : T(t+1) = AT(t) + \mu T_H \nu(t) + \theta T_E + 0.01\varsigma(t)T(t),$$

where $A \in \mathbb{R}^{n\times n}$ is a matrix with diagonal elements given by $\bar{a}_{ii} = (1 - 2\alpha - \theta - \mu\nu_i(t))$, off-diagonal elements $\bar{a}_{i,i+1} = \bar{a}_{i+1,i} = \bar{a}_{1,n} = \bar{a}_{n,1} = \alpha$, $i \in \{1, \ldots, n-1\}$, and all other elements are identically zero. The parameters $\alpha = 0.005$, $\theta = 0.06$ and $\mu = 0.145$ are conduction factors between rooms i and $i \pm 1$, external environment and room i, heater and room i, respectively. The heater temperature is maintained at $T_H = 40\,^\circ\mathsf{C}$ and the outside temperature $T_{ei} = -5\,^\circ\mathsf{C}$ for all rooms $i \in \{1, \ldots, n\}$. We also have $T(t) = [T_1(t); \ldots; T_n(t)]$, $T_E = [T_{e1}; \ldots; T_{en}]$, $\nu(t) = [\nu_1(t); \ldots; \nu_n(t)]$ and $\varsigma(t) = \mathsf{diag}(\varsigma_1(t), \ldots, \varsigma_n(t))$.

The regions of interest are given by $X = [0, 20]^n$, $X^0 = [17, 18]^n$, $X^1 = [0, 15]^n$, and $X^2 = X \backslash (X^0 \cup X^1)$. We consider these regions to be associated with a set of atomic propositions $\mathcal{AP} = \{p_0, p_1, p_2\}$ via a labeling function $L : X \to \mathcal{AP}$ such that $L(x \in X^z) = p_z$ for all $z \in \{0, 1, 2\}$. The requirement of our case study is to synthesize a controller $\nu : \mathbb{N} \to [0, 0.6]^n$ satisfying the specification represented by DSA $\mathcal{A}$ in Figure 5.8 with $Acc =< q_4, \varnothing >$. Note that this corresponds to the LTL specification given by $p_0 \to (\mathsf{X}\,\neg p_0 \vee \mathsf{F}(p_0 \wedge \mathsf{G}\,\neg p_1))$, where X, F, and G denotes the temporal operators next, eventually, and globally, respectively. In order to achieve this, we must perform sequential decomposition on the DSA $\mathcal{A}^s$. To do this, we first obtain the set of all lassos $\mathcal{R}^p$ for each $p \in \mathcal{AP}$. This can be obtained as $\mathcal{R}^{p_0} = \{(q_0, q_2, q_3, q_4, q_3, q_4)\}$. Correspondingly, the finite path set and the finite cycle set can be obtained as $\mathcal{R}_f^{p_0} = \{(q_0, q_2, q_3, q_4)\}$ and $\mathcal{R}_l^{p_0} = \{(q_4, q_3, q_4)\}$, respectively. As it can be seen, there is only one lasso $\tilde{\mathbf{q}} \in \mathcal{R}^{p_0}$ which are decomposed into $\mathcal{P}^{p_0}(\tilde{\mathbf{q}}) = \{(q_0, q_2, q_3), (q_2, q_3, q_4), (q_3, q_4, q_3), (q_4, q_3, q_4)\}$. Furthermore, we have $\mathcal{P}^{p_0}(\tilde{\mathbf{q}}_f) = \{(q_0, q_2, q_3), (q_2, q_3, q_4), (q_3, q_4, q_3)\}$ and $\mathcal{P}^{p_0}(\tilde{\mathbf{q}}_l) = \{(q_4, q_3, q_4)\}$. This constitutes four safety tasks for which we need to obtain CBCs and corresponding controllers. However, following Remark 38, we prioritize the computation of CBC and corresponding controller for the safety task (p_0, p_1) corresponding to $\vartheta = (q_4, q_3, q_4) \in \mathcal{P}^{p_0}(\tilde{\mathbf{q}}_l)$.

Since we are working with a large-scale interconnected system, it is not scalable to monolithically construct CBC as well as the controller. Therefore, we utilize the compositionality results

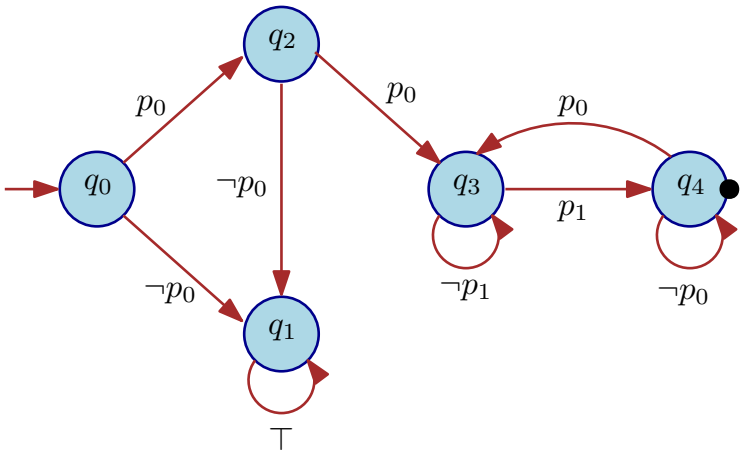


Figure 5.8: DSA $\mathcal{A}^s$ representing the specification with $Acc =< q_4, \varnothing >$.

presented in Section 3.4 to compositionally construct CBC and controller by computing CSBCs for subsystems. To do so, we first consider our network $\mathfrak{S}$ as an interconnection of $n = 300$ subsystems, each of which constitutes a room. The state evolution of these individual subsystems is given by

$$\mathfrak{S}_i : T_i(k+1) = \bar{a}T_i(t) + \mu T_H \nu_i(t) + \alpha w_i(t) + \theta T_{ei} + 0.01\varsigma_i(t)T_i(t).$$

It can be easily verified that $\mathfrak{S} = \mathcal{I}(\mathfrak{S}_1, \ldots, \mathfrak{S}_n)$ with coupling matrix M such that $m_{i,i+1} = m_{i+1,i} = m_{1,n} = m_{n,1} = 1$, $i \in \{1, \ldots, n-1\}$ and all other elements are identically zero. Now, for the safety task in consideration, we obtain CSBC as a 4^{th} order polynomial given by $\mathbb{B}_i(T_i) = 9.6445 - 0.6911T_i + 0.1396{T_i}^2 - 0.0163{T_i}^3 + 0.0005{T_i}^4$ and the corresponding controller is computed to be $\varpi_i(T_i) = 0.59 - 0.011T_i$, similar to the one obtained in Section 3.4.6. The compositionality results presented in 3.4 are then applied to obtain the CBC for the interconnected system as $\mathbb{B}(T) = \sum_{i=1}^{300}(9.6445 - 0.6911T_i + 0.1396{T_i}^2 - 0.0163{T_i}^3 + 0.0005{T_i}^4)$, while the suitable controller for the trajectories in $X^0 = L^{-1}(p_0)$ is obtained as $\varpi_{p_0}(T) = [0.59 - 0.011T_1; \ldots; 0.59 - 0.011T_{300}]$. The upper bound on the probability that the solution processes of the interconnected system $\mathfrak{S}$ start from $X_0 = X^0$ and reach $X_u = X^1$ is computed to be equal to 0.0594 by using Lemma 10. However, from Theorem 14 and Remark 38, we can conclude that having the CSBC for $\vartheta \in \mathcal{P}^{p_0}(\tilde{\mathbf{q}}_l)$ allows us to guarantee that the state q_4 is visited only finitely often with probability 1, thereby allowing the satisfaction of DSA $\mathcal{A}^s$ with probability 1. Therefore, it is not required to compute CBC for other safety tasks in $\mathcal{P}^{p_0}(\tilde{\mathbf{q}})$, and instead we assign a pessimistic upper bound of 1 for the violation of these safety tasks. The corresponding controllers are assumed to take any random value constrained within the input set. Finally, a switching mechanism for controllers is obtained as explained in Subsection 5.3.3.

We now compute an overall probability of satisfaction of specification expressed by DSA $\mathcal{A}^s$ when starting from an initial state $x_0 \in X^0$ by using Theorem 14:

$$\mathbb{P}\{L(\mathbf{x}_{x_0,\varpi}) \models \mathcal{A}^s\} = 1.$$

The simulation of the network satisfying the above specification directly follows from Figure 3.5.

The computation of CSBC and corresponding local controller take up to 240 seconds on a machine with Linux Ubuntu 18.04 OS (Intel i7-8665U CPU with 32GB RAM).

5.4 Formal Verification of Dynamical Systems against Hyperproperties

The results from Section 5.2 and Section 5.3 were focused on the analysis of trace properties like linear temporal logic or (in)finite automata. As already described in Chapter 2, trace properties are properties that can be described over individual execution traces of the system. However, such a description is limited to safety and liveness specifications, and as such, fails to describe many important security and planning properties, like opacity, non-interference, robustness, etc. For instance, consider the opacity property in a system that is prone to *intrusion attacks*. For this system, it may be required that some secret information is never revealed, *i.e.*, observations from the outside remain indistinguishable from each other, despite the secret. Since this property requires us to relate multiple execution traces simultaneously (*i.e.* to determine that they are indistinguishable), such a specification cannot be expressed using linear temporal logic or ω-regular properties. Therefore, the focus of this section is to provide a formal *verification* approach for specifications that can be characterized over multiple trace executions, called hyperproperties, for discrete-time dynamical systems with *exogenous inputs* (see Section 5.4.1). These hyperproperties may be specified using hyper-temporal logics (HyperLTL), which serves as an extension to standard LTL specifications, defined over a set of traces, rather than individual ones. The syntax and semantics of HyperLTL have already been introduced in Section 2.4 of Chapter 2, and is omitted here for the sake of brevity.

5.4.1 Problem Definition

We consider a discrete-time dynamical system with exogenous inputs (for brevity, we use the term system) defined by a tuple $\mathfrak{S} = (X, W, f)$, where $X \subseteq \mathbb{R}^n$ and $W \subseteq \mathbb{R}^m$ are the (potentially uncountable) state and exogenous input sets, and $f : X \times W \to X$ is the transition function that characterizes the state evolution. The evolution of the system $\mathfrak{S}$ for a given initial state $x_0 \in X$ and exogenous input sequence $\nu : \mathbb{N} \to W$, denoted by $\mathbf{x}_{x_0,\nu}$, is given by a state sequence $\mathbf{x} : \mathbb{N} \to X$, where

$$\mathbf{x}(t+1) = f(\mathbf{x}(t), \nu(t)), \tag{5.13}$$

starting from $\mathbf{x}(\mathbf{0}) = x_0$. For a system $\mathfrak{S}$, we define its P-fold self-composition as P-fold augmented system $\mathfrak{S}^{\mathrm{P}} = (X^{\mathrm{P}}, W^{\mathrm{P}}, f^{\mathrm{P}})$ where X^{P} and W^{P} are P-ary Cartesian powers of X, and W, respectively, and $f^{\mathrm{P}} : X^{\mathrm{P}} \times W^{\mathrm{P}} \to X^{\mathrm{P}}$ is equivalent to the P-ary Cartesian power of f, i.e. $f^{\mathrm{P}} : (X \times W)^{\mathrm{P}} \to X^{\mathrm{P}}$ by using the zip function. We use these two types interchangeably. We use $\tilde{x} = [x_1; \ldots; x_{\mathrm{P}}]$ and $\tilde{w} = [w_1; \ldots; w_{\mathrm{P}}]$ to denote the state and exogenous input of the augmented system $\mathfrak{S}^{\mathrm{P}}$, respectively. Similarly, we write $\tilde{\mathbf{x}}_{\tilde{x}_0,\tilde{\nu}}$ for the state sequence of $\mathfrak{S}^{\mathrm{P}}$ starting from an initial state $\tilde{x}_0 \in X^{\mathrm{P}}$ and under exogenous input sequence $\tilde{\nu} : \mathbb{N} \to W^{\mathrm{P}}$. Note that the augmented system is defined in order to reason about the multiple executions of the system simultaneously, *i.e.*,

execution of one component can be separated from another via the unzip function. Also note that the state sequence $\tilde{\mathbf{x}}_{\tilde{x}_0,\tilde{\nu}}$ may be considered as a P-tuple of state sequences of $\mathfrak{S}$ under different initial conditions and exogenous inputs, respectively, *i.e.*, $\tilde{\mathbf{x}}_{\tilde{x}_0,\tilde{\nu}} = (\mathbf{x}_{x_{01},\nu_1}, \ldots, \mathbf{x}_{x_{0\mathrm{P}},\nu_\mathrm{P}})$. Here, $\tilde{x}_0 = [x_{01}; \ldots; x_{0\mathrm{P}}]$ is the initial state of the augmented system and $\tilde{\nu} = (\nu_1, \ldots, \nu_\mathrm{P})$ is the P-tuple of input sequences that is applied to the augmented system.

The main problem is to formally verify that the system $\mathfrak{S}$ satisfies a given HyperLTL specification that is described using a formula $\phi = \mu_1\pi_1 \ldots \mu_\mathrm{P}\pi_\mathrm{P}\psi$, where $\mu_i \in \{\exists, \forall\}$, for all $i \in \{1, \ldots, m\}$, and ψ corresponds to some quantifier-free specification consisting of atomic propositions in the set $\mathcal{AP}$. For more details on the formula construction, we refer the reader to the syntax and semantics of HyperLTL presented in Section 2.4. Now, we want to relate the state sequences of the system $\mathfrak{S}$ to the HyperLTL specification through the labeling function $L : X \to \Sigma$, where $\Sigma = 2^{\mathcal{AP}}$.

Definition 28. *For the system $\mathfrak{S} = (X, U, \varsigma, f)$ and a HyperLTL specification $\phi = \mu_1\pi_1 \ldots \mu_P\pi_P\psi$, consider a labeling function $L : X \to \Sigma$. For an infinite-state sequence $\mathbf{x} = (\mathbf{x}(0), \mathbf{x}(1), \ldots) \in X^\omega$, the corresponding trace over Σ is given by $L(\mathbf{x}) := (\sigma_0, \sigma_1, \ldots) \in \Sigma^\omega$, where $\sigma_i = L(\mathbf{x}(i))$ for all $i \in \mathbb{N}$. Correspondingly, $T(\mathfrak{S}, L)$ denotes the set of traces T of $\mathfrak{S}$ corresponding to the labeling function L.*

Now, we state the main HyperLTL verification problem that we aim to solve in this part of the chapter.

Problem 9. *Given a discrete-time dynamical system with exogenous inputs $\mathfrak{S} = (X, W, f)$, a labeling function $L : X \to \Sigma$, and a HyperLTL specification $\phi = \mu_1\pi_1 \ldots \mu_P\pi_P\psi$, the HyperLTL verification problem is to decide whether $T(\mathfrak{S}, L) \models \phi$.*

Remark 43. *Note that, unlike the previous sections which considered the controller synthesis problem for trace properties, in this section, we only consider the formal verification procedure. As such, the controller synthesis procedure is out of the scope of our work and is reserved for future investigations (see the discussion in Section 6.2 of Chapter 6.)*

While model checking of finite-state systems against HyperLTL specifications is decidable [50], the verification problem stated above is in general undecidable for continuous state-space systems considered here. It follows readily from the fact that even simple reachability is undecidable for simple continuous state-space dynamical systems [13]. Our approach provides a sound procedure for Problem 9. To better illustrate our approach, we use the following case study as a running example.

Example 7. *Here, we consider the discrete-time evolution of the temperature $T(\cdot)$ in a room in the presence of a safety controller (as designed in [64]) given by*

$$\begin{aligned}\mathfrak{S} : T(k+1) &= T(t) + \tau_s\alpha_e(T_e - T(t)) \\ &\quad + \tau_s\alpha_h(T_h - T(t))(c_1T(t) + c_2),\end{aligned} \tag{5.14}$$

where parameters $\alpha_e = 0.008$ and $\alpha_h = 0.0036$ are heat exchange coefficients, $T_e = 15°C$ is the ambient temperature, $T_h = 55°C$ is the heater temperature, $c_1 = -0.0024$ and $c_2 = 0.5357$ are

controller parameters, and $\tau_s = 5$ minutes is the sampling time. We want to verify the initial-state robustness property, which requires that if a state sequence starting from a given initial condition remains safe, then all the state runs starting from ρ-close initial conditions must also remain safe. Such a specification is especially useful when there are uncertainties arising from not knowing the exact initial state. Note that robustness is a commonly studied property in the classical control theory [143]. However, we provide here an alternative method to verify robustness by formulating it as a HyperLTL formula. To describe our specification as a HyperLTL formula, we consider the state set $X = [20, 35]$. We further introduce the safe set as $X^1 = [20, 25]$ and the unsafe set as $X^2 = [25, 35]$. For the system $\mathfrak{S}$, the predefined initial state is given by $X^3 = \{21\}$. We also define the set $X^4 = [20.5, 21.5]$ to capture ρ-close states with respect to the initial state, where $\rho = 0.5$. The set of atomic propositions is $\mathcal{AP} = \{p_1, p_2, p_3, p_4\}$, where $L(x \in X^i) = p_i$, for all $i \in \{1, 2, 3, 4\}$. The HyperLTL formula for initial-state robustness specification is $\phi = \forall\pi_1 \forall\pi_2 (p_{3\pi_1} \wedge p_{4\pi_2}) \rightarrow \mathsf{G}(p_{1\pi_1} \wedge p_{1\pi_2})$.

5.4.2 Augmented Barrier Certificates

Verification of HyperLTL formulae in the context of finite systems has been well studied [50, 30]. The verification procedure is based on automata-theoretic model-checking, where quantifier-free fragments of the desired HyperLTL formulae are compiled into ω-automata, and trace quantification is handled by appropriately composing these automata with the underlying Kripke structure. The interleavings of automata and Kripke products lead to automata whose language emptiness decides the satisfaction of the specification. Unfortunately, this approach cannot be extended to continuous state-space systems by simply using abstraction-based techniques as system relations (*e.g.*, simulation relations) may not preserve hyperproperties [137].

To verify the specification $\phi = \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\psi$ against the system $\mathfrak{S}$, we compile the negation of the HyperLTL formula into an implicitly quantified Büchi automata $\mathcal{A}^b_{\neg\psi} = (Q, q_0, \Sigma^{\mathrm{P}}, \delta, F)$ (see Section 2.4 for details). Note that the acceptance condition of $\mathcal{A}^b_{\neg\psi}$ requires that the state runs visit F infinitely many times. By accordingly ensuring the violation of the acceptance condition of $\mathcal{A}^b_{\neg\psi}$, one can ensure the satisfaction of the original specification. Our verification procedure is depicted in Figure 5.9. Here, given a HyperLTL specification ϕ and system $\mathfrak{S}$, we construct the P-fold augmented system $\mathfrak{S}^{\mathrm{P}}$, and the NBA for $\neg\psi$. We then find an augmented barrier certificate (ABC) that acts as a proof certificate of conditional invariance (see Definition 29) for some transitions along every lasso of the NBA. This acts as a "scissor" and allows us to conclude that the accepting states in F are never visited, and correspondingly, the system $\mathfrak{S}$ satisfies the specification ϕ. We note that our approach is not complete in that if we cannot find an ABC for at least one transition along every lasso then we cannot conclude that the system does not satisfy ϕ. In the following, we introduce the idea of augmented barrier certificates and provide an automata-theoretic sound verification approach for Problem 9.

HyperLTL Evaluation Game Semantics

We provide a game semantics perspective to the HyperLTL verification problem as a two-player *stage-based evaluation game* played between two players, Eloise ($\exists$) and Abelard ($\forall$), where

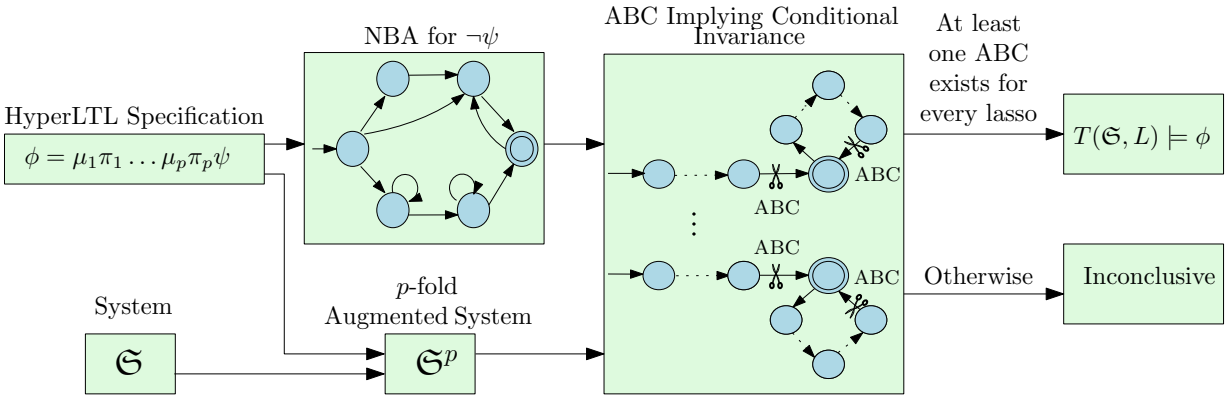


Figure 5.9: A schematic block diagram illustrating the verification procedure.

Eloise takes the role of a verifier and her goal is to prove that $T(\mathfrak{S}, L) \models \phi$, while the goal of Abelard (the spoiler) is the opposite. Given a HyperLTL formula $\mu_1\pi_1 \ldots \mu_{\text{P}}\pi_{\text{P}}\psi$, we say that Eloise controls the quantifier μ_i if $\mu_i = \exists$, otherwise we say that Abelard controls the quantifier. The game continues in stages. In the first stage, the game begins with a token in the initial position $(\mu_1\pi_1\mu_2\pi_2 \ldots \mu_{\text{P}}\pi_{\text{P}}\psi, \Pi = \emptyset)$ [56] and the player controlling the left-most quantifier μ_1 chooses a trace $\boldsymbol{\sigma}_1$ from $T(\mathfrak{S}, L)$ and moves the token to the next position $(\mu_2\pi_2 \ldots \mu_{\text{P}}\pi_{\text{P}}\psi, \Pi = \{\pi_1 \rightarrow \boldsymbol{\sigma_1}\})$. The game from the next position continues in a similar fashion until we reach a position with a quantifier-free HyperLTL formula. We call such positions *terminal*. We say that a terminal position $(\psi, \Pi = \{\pi_1 \rightarrow \boldsymbol{\sigma_1}, \ldots, \pi_{\text{P}} \rightarrow \boldsymbol{\sigma}_{\text{P}}\})$ is winning for Eloise if $\mathsf{zip}(\boldsymbol{\sigma_1}, \ldots, \boldsymbol{\sigma}_{\text{P}}) \models \psi$ with the standard LTL semantics for the formula ψ. We say that Eloise wins the multi-stage evaluation game if she has a way to choose her moves such that no matter how Abelard chooses his moves the game ends in a winning position for Eloise; otherwise, Abelard wins the game. Notice that at every step, both players have access to complete infinite traces that have been chosen by players in earlier positions as a part of the position description. It is trivial to see that Eloise wins the evaluation game if and only if $\Pi \models_T \phi$.

We consider another version of evaluation games that we dub *turn-based evaluation games*. These games are played on the P-fold self-composition $\mathfrak{S}^{\text{P}} = (X^{\text{P}}, W^{\text{P}}, f^{\text{P}})$ of the system $\mathfrak{S}$. These games start with a token in some initial configuration $[x_{01}; x_{02}; \ldots; x_{0\text{P}}] \in X^{\text{P}}$ and at every round first the player controlling μ_1 chooses an action $w_1 \in W$, followed by the player controlling μ_2, and so on. In this way, the players form a set of joint actions $[w_1; w_2; \ldots; w_{\text{P}}] \in W^{\text{P}}$ and the token is moved to a state $f^{\text{P}}([x_{01}; x_{02}; \ldots; x_{0\text{P}}], [w_1; w_2; \ldots; w_{\text{P}}])$. The game continues in this fashion indefinitely and the players thus form an infinite run $\tilde{\mathbf{x}}$ of $\mathfrak{S}^{\text{P}} = (X^{\text{P}}, W^{\text{P}}, f^{\text{P}})$. To relate the augmented system $\mathfrak{S}^{\text{P}}$ with letters in Σ^{P}, we extend the definition of the labeling function to the augmented system domain by using $L^{\text{P}} : X^{\text{P}} \rightarrow \Sigma^{\text{P}}$. We say that the trace $\tilde{\mathbf{x}}$ is winning for Eloise if $L^{\text{P}}(\tilde{\mathbf{x}}) \models \psi$. We say that Eloise has a winning strategy in the turn-based evaluation game if she can select her moves in such a way that no matter how Abelard chooses his moves (including using an arbitrary look-ahead), the resulting trace is winning for Eloise. Moreover, we say that Eloise has a positional winning strategy if to select actions in a given round her choice depends only on the current states and choices resolved before her turn for the other quantifiers in the current round.

Lemma 11. *If Eloise has a positional winning strategy in a turn-based evaluation game for $\mathfrak{S}$ with respect to labeling function L and HyperLTL specification ϕ, then she has a winning strategy in the stage-based evaluation game.*

Proof. If Eloise has a positional winning strategy in the turn-based evaluation game, then she can use the same strategy to choose traces in the stage-based evaluation game such that each index depends only on the states and actions at the current index in the traces quantified so far. Then, against an arbitrary policy chosen by Abelard, the resulting P-tuple of the traces satisfies the LTL specification ψ. As a result, the set of traces $T(\mathfrak{S}, L)$ of $\mathfrak{S}$ satisfy ϕ. □

Augmented Barrier Certificates

We reduce the search for a positional strategy for Eloise in a turn-based evaluation game to the search for barrier functions like certificates that we call augmented barrier certificates (ABCs). Just like barrier certificates provide proof that separates two sets over X for arbitrary traces of the system, ABCs provide a proof that separate two sets over X^P for appropriately chosen traces by players. To tie in the notion of ABCs with HyperLTL properties, we present a special class of properties that we call *conditional invariance* properties that generalize the notion of invariance.

Definition 29. *We say that a HyperLTL formula $\chi{=}\mu_1\pi_1 \dots \mu_P\pi_P\xi$ is a conditional invariance (CI) if ξ is of the form $\mathsf{G}(s_A \rightarrow \mathsf{G}(\neg s_B))$ where s_A, s_B are Boolean combinations of atomic propositions.*

We now present the definition of augmented barrier certificates, which are barrier certificate-type functions constructed over the augmented system $\mathfrak{S}^P$ for the satisfaction of conditional invariances.

Definition 30. *Consider a CI $\chi = \mu_1\pi_1 \dots \mu_P\pi_P\, \mathsf{G}(s_A \rightarrow \mathsf{G}(\neg s_B))$ and the sets $L^{P-1}(s_A) = A \subseteq X^P$ and $L^{P-1}(s_B) = B \subseteq X^P$. We say that $\mathbb{B} : X^P \rightarrow \mathbb{R}$ is an* augmented barrier certificate (ABC) *for a system $\mathfrak{S} = (X, W, f)$ and property χ from the set $A \subseteq X^P$ to set $B \subseteq X^P$ if*

$$\mathbb{B}(\tilde{x}) \leq 0, \qquad \text{for all } \tilde{x} \in A, \tag{5.15}$$

$$\mathbb{B}(\tilde{x}) > 0, \qquad \text{for all } \tilde{x} \in B, \tag{5.16}$$

and $\forall x_1 \in X, \mu_1 w_1 \in W, \forall x_2 \in X, \mu_2 w_2 \in W, \dots, \forall x_P \in X, \mu_P w_P \in W$ one has:

$$\mathbb{B}(f^P(\tilde{x}, \tilde{w})) - \mathbb{B}(\tilde{x}) \leq 0, \tag{5.17}$$

where $\tilde{x} = [x_1; \dots; x_P]$ and $\tilde{w} = [w_1; \dots; w_P]$.

Remark 44. *Note that all the states' components x_i of the augmented system $\mathfrak{S}^P$ in condition* (5.17) *are quantified universally, while their corresponding exogenous inputs' components w_i are quantified according to μ_i. The components x_i cannot be quantified according to μ_i as it may result in the state runs of the augmented system $\mathfrak{S}^P$ reaching the region B. To see this, consider a component $x_i(t)$ at some time step $t \in \mathbb{N}$ which is quantified by $\mu_i = \exists$. For that component, one may be able to pick a corresponding input component $w_i(t)$ such that the augmented barrier*

certificate is non-increasing according to condition (5.17)*. However, since x_i is quantified only existentially, one may fail to ensure the existence of input $w_i(t+1)$ for $x_i(t+1)$ at the next time step such that the augmented barrier certificate is still non-increasing. Due to this, one would fail to ensure that the augmented barrier certificate remains non-increasing at every time step, possibly resulting in safety violations. Note that such quantification of the states is without loss of generality and does not restrict the class of HyperLTL specifications considered for verification.*

Remark 45. *For a CI $\chi = \mu_1\pi_1 \dots \mu_P\pi_P\, \mathsf{G}(s_A \rightarrow \mathsf{G}(\neg s_B))$, if the set $L^{P-1}(s_A) \cap L^{P-1}(s_B)$ is non-empty, then there exists no ABC satisfying conditions* (5.15)-(5.17)*. This is due to the conflict in the satisfaction of conditions* (5.17) *and* (5.17)*.*

Lemma 12. *The existence of an ABC for a conditional invariance χ implies that $T(\mathfrak{S}, L) \models \chi$.*

Proof. We prove this by contradiction. Suppose an ABC exists for χ, but χ is not valid. Then, regardless of Eloise's strategy, Abelard always has a strategy that allows him to win. Let the set of traces selected by the players be $T = \{\boldsymbol{\sigma}_1, \dots \boldsymbol{\sigma}_m\}$. For Abelard to win, he must ensure that for some $j \in \mathbb{N}$, $\Pi[j, \infty] \models_T s_A$ and for some $i > j$, $\Pi[i, \infty] \models_T s_B$ (The fact that $i \neq j$ can be inferred from the existence of ABC, see Remark 45) to falsify $\mathsf{G}(s_A \rightarrow \mathsf{G}(\neg s_B))$. We consider the case where the strategy of Eloise is to select inputs according to condition (5.17). We note that selecting such a strategy leads to a non-increase in the value of the ABC for the corresponding state in the augmented system, regardless of Abelard's strategy. Let the set of traces at positions j and i correspond to states $\tilde{x}$ and $\tilde{x}'$ in the augmented system and let the corresponding input sequence that takes us from $\tilde{x}$ to $\tilde{x}'$ be $\tilde{v}$. From conditions (5.15) and (5.16), we have $\mathbb{B}(\tilde{x}) \leq 0$ and $\mathbb{B}(\tilde{x}') > 0$. For any $l \geq 0$, let $\tilde{w} = \tilde{v}(l)$, and $\tilde{x}_l = \mathbf{x}(l)$, then we have $\mathbb{B}(f^{\mathrm{P}}(\tilde{x}_l, \tilde{w})) \leq \mathbb{B}(\tilde{x}_l)$ from condition (5.17) regardless of Abelard's strategy. By induction on this condition, we can infer that $\mathbb{B}(\tilde{x}') \leq 0$. This is a contradiction to condition (5.16). So, we infer that χ is valid in the turn-based game setting. Therefore, χ is also valid in the stage-based game setting according to Lemma 11. □

5.4.3 Verification Procedure

To extend CI guarantees obtained via ABCs to arbitrary HyperLTL specifications, we first construct a non-deterministic Büchi automaton (NBA) $\mathcal{A}^b_{\neg\psi} = (Q, q_0, \Sigma^{\mathrm{P}}, \delta, F)$ corresponding to $\neg\psi$. Note that the definition of NBA and the corresponding construction of NBA for HyperLTL specifications is already presented in Section 2.4 of Chapter 2 and is omitted here for brevity.

Then, we employ ABCs as *scissors* disallowing transitions to the accepting states of $\mathcal{A}^b_{\neg\psi}$. To do so, we first present the following lemmas to guarantee disjunction or conjunction over a set of CIs.

Lemma 13. *Given a set of CIs $\{\chi_1, \dots, \chi_k\}$, the existence of an ABC $\mathbb{B}_j$ for some CI χ_j, where $\chi_j = \mu_1\pi_1 \dots \mu_P\pi_P\, \mathsf{G}(s_{A_j} \rightarrow \mathsf{G}(\neg s_{B_j}))$ for $1 \leq j \leq k$, implies that $T(\mathfrak{S}, L) \models \chi$, where*

$$\chi = \mu_1\pi_1 \dots \mu_P\pi_P \bigvee_{1 \leq j \leq k} \mathsf{G}(s_{A_j} \rightarrow \mathsf{G}(\neg s_{B_j})).$$

Proof. From Lemma 12, for a conditional invariance $\chi_j = \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\, \mathsf{G}(s_{A_j} \rightarrow \mathsf{G}(\neg s_{B_j}))$, existence of an ABC $\mathbb{B}_j$ implies that Eloise has a winning strategy to ensure that $T(\mathfrak{S}, L) \models \chi_j$. Therefore, for a set of conditional invariances $\chi_1, \dots, \chi_k$, Eloise may choose the same winning strategy corresponding to χ_j to ensure that at least one of the conditional invariances in the set holds. Therefore, we get $T(\mathfrak{S}, L) \models \chi$, where $\chi = \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}} \bigvee\limits_{1\leq j\leq k} \mathsf{G}(s_{A_j} \rightarrow \mathsf{G}(\neg s_{B_j}))$. □

We say that a function $\mathbb{B}$ is a common ABC for a set of CIs $\{\chi_1, \dots, \chi_k\}$ if $\mathbb{B}$ is an ABC for all of the CIs.

Lemma 14. *The existence of a common ABC for a set of CIs* $\{\chi_1, \chi_2, \dots, \chi_k\}$*, where* $\chi_i = \mu_1\pi_1 \dots \mu_P\pi_P\, \mathsf{G}(s_{A_i} \rightarrow \mathsf{G}(\neg s_{B_i}))$*, for* $1 \leq i \leq k$*, implies that* $T(\mathfrak{S}, L) \models \chi$*, where*

$$\chi = \mu_1\pi_1 \dots \mu_P\pi_P \bigwedge_{1\leq i\leq k} \mathsf{G}(s_{A_i} \rightarrow \mathsf{G}(\neg s_{B_i})).$$

Proof. The proof once again follows from Lemma 12. The existence of a common ABC $\mathbb{B}$ guarantees that condition (5.17) is satisfied for all conditional invariances χ_i, $1 \leq i \leq k$. This implies that Eloise may use the same strategy to disallow all the transition pairs (s_{A_i}, s_{B_i}), $1 \leq i \leq k$. Therefore, we have $T(\mathfrak{S}, L) \models \chi$, where $\chi = \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}} \bigwedge\limits_{1\leq i\leq k} \mathsf{G}(s_{A_i} \rightarrow \mathsf{G}(\neg s_{B_i}))$ □

We now provide guarantees for HyperLTL formulae that can be derived from CIs conjunctive normal form.

Lemma 15. *Given a set of sets of conditional invariances*

$$\{\{\chi_{1,1}, \dots, \chi_{1,v_1}\}, \{\chi_{2,1}, \dots, \chi_{2,v_2}\}, \dots \{\chi_{k,1}, \dots, \chi_{k,v_k}\}\},$$

where $\chi_{i,j} = \mu_1\pi_1 \dots \mu_P\pi_P\, \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$*, the existence of a common ABC for* $\chi_{i,j}$ *for every* $1 \leq i \leq k$ *and some* $1 \leq j \leq v_i$ *implies that* $T(\mathfrak{S}, L) \models \chi$*, where*

$$\chi = \mu_1\pi_1 \dots \mu_P\pi_P \bigwedge_{(1\leq i\leq k)} \bigvee_{(1\leq j\leq v_i)} \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}})).$$

Proof. From Lemma 13, for the set of conditional invariances $\{\chi_{i,1}, \dots, \chi_{i,v_i}\}$ for some $1 \leq i \leq k$, the existence of ABC $\mathbb{B}$ for some $\chi_{i,j}$ implies that $T(\mathfrak{S}, L) \models \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}} \bigvee\limits_{1\leq j\leq v_i} \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$. For each $1 \leq i \leq k$, if there exists a common ABC $\mathbb{B}$ for some $\chi_{i,j}$, $1 \leq j \leq v_i$, then by Lemma 14, we have that $T(\mathfrak{S}, L) \models \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}} \bigwedge\limits_{1\leq i\leq k} \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$. By combining these two results, for a family of the set of conditional invariances $\{\{\chi_{1,1}, \dots, \chi_{1,v_1}\}, \dots, \{\chi_{k,1}, \dots, \chi_{k,v_k}\}\}$, one has $T(\mathfrak{S}, L) \models \chi$, where $\chi = \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}} \bigwedge\limits_{(1\leq i\leq k)} \bigvee\limits_{(1\leq j\leq v_i)} \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$. □

Now, to find the solution to Problem 9, we consider the NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to the specification $\neg\psi$, obtained from HyperLTL specification $\phi = \mu_1\pi_1 \dots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\psi$. Then, our verification approach relies on reducing the complex HyperLTL specification into a collection of conditional

invariance guarantees over consecutive transition pairs. Consider the NBA $\mathcal{A}^b_{\neg\psi}$ compiling the negation of the desired specification. For this NBA, consider the set $\mathcal{R}$ to be consisting of all *lassos* $\tilde{\mathbf{q}} = (\tilde{\mathbf{q}}_f, \tilde{\mathbf{q}}_l)$, where $\tilde{\mathbf{q}}_f$ is the simple finite path from the initial state to the accepting state, and $\tilde{\mathbf{q}}_l$ is the simple finite cycle from the accepting state to itself.

Now, for each $p \in \Sigma^{\mathrm{P}}$, we define a set $\mathcal{R}^p$ as

$$\mathcal{R}^p := \Big\{\tilde{\mathbf{q}} = (\underbrace{q_0^f, q_1^f, \ldots, q_{a_f}^f, q_0^l}_{\tilde{\mathbf{q}}_f}, \overbrace{q_1^l, \ldots, q_{a_l}^l, q_0^l}^{\tilde{\mathbf{q}}_l}) \in \mathcal{R} \,\Big|\, q_1^f \in \delta(q_0^f, p), p \in \Sigma^{\mathrm{P}}\Big\}.$$

Now, in order to perform decomposition into conditional invariances, we define a set $\mathcal{P}^p(\tilde{\mathbf{q}})$ for any $\tilde{\mathbf{q}} = (q_0, q_1, \ldots, q_{a_f+a_l+3}) \in \mathcal{R}^p$ as

$$\mathcal{P}^p(\tilde{\mathbf{q}}) = \big\{(q_i, q_{i+1}, q_{i+2}) \,\big|\, 0 \le i \le a_f + a_l + 1\big\}.$$

We correspondingly define the set $\mathcal{S}^p(\tilde{\mathbf{q}})$ to be the set of all *consecutive transition pairs* $(s_A, s_B) \in \Sigma^{\mathrm{P}}$ such that

$$\mathcal{S}^p(\tilde{\mathbf{q}}) = \{(s_A, s_B) \mid q_{i+1} = \delta(q_i, s_A), q_{i+2} = \delta(q_{i+1}, s_B), (q_i, q_{i+1}, q + i + 2) \in \mathcal{P}^p(\tilde{\mathbf{q}})\}.$$

Correspondingly, we use $\mathcal{S}(\tilde{\mathbf{q}})$ to refer to the set of all consecutive pairs from $\tilde{\mathbf{q}}$, irrespective of p, *i.e.*, $\mathcal{S}(\tilde{\mathbf{q}}) = \bigcup_{p\in\Sigma^{\mathrm{P}}} \mathcal{S}^p(\tilde{\mathbf{q}})$. Consequently, we let the set $\mathcal{S}_{\mathcal{A}^b_{\neg\psi}} = \bigcup_{p\in\Sigma^{\mathrm{P}}} \bigcup_{\tilde{\mathbf{q}}\in\mathcal{R}^p} \mathcal{S}^p(\tilde{\mathbf{q}})$ be the set of all such consecutive transition pairs obtained from NBA $\mathcal{A}^b_{\neg\psi}$. These transition pairs correspond to different conditional invariance specifications for which a suitable ABC is synthesized. We now state the following theorem that characterizes a condition to solve Problem 9.

Theorem 15. *Given a HyperLTL specification $\phi = \mu_1\pi_1 \ldots \mu_P\pi_P\psi$, the existence of a common ABC $\mathbb{B}$ for some consecutive transition pair along every lasso of $A^b_{\neg\psi}$ guarantees that $T(\mathfrak{S}, L) \models \phi$.*

Proof. Let the NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to $\neg\psi$ have k lassos $\tilde{\mathbf{q}}$ from which the conditional invariances in $\mathcal{S}(\tilde{\mathbf{q}})$ are obtained, such that the i^{th} lasso has v_i pairs of consecutive transitions. Let the pair $(s_{A_{i,j}}, s_{B_{i,j}})$ correspond to the j^{th} pair of consecutive transitions along the i^{th} lasso. Let $\chi_{i,j} = \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\, \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$ denote a conditional invariance specification and let us consider the set of conditional invariances $\{\chi_{1,1}, \ldots, \chi_{1,v_1}, \chi_{2,1} \cdots \chi_{2,v_2}, \ldots, \chi_{k,1}, \cdots \chi_{k,v_k}\}$. Then the existence of a common ABC satisfying Lemma 15 implies that if the augmented system lands on a state satisfying $s_{A_{i,j}}$, Eloise has a strategy to ensure that it never reaches a state satisfying $s_{B_{i,j}}$ for every $1 \le i \le k$, and some $1 \le j \le v_j$. However, to satisfy $\neg\psi$, Abelard must have a strategy that allows him to visit a state satisfying some $s_{A_{i,j}}$ and then later visit a state satisfying $s_{B_{i,j}}$, for some $1 \le i \le k$ and every $1 \le j \le v_i$, to follow the transitions along the i^{th} lasso. Since this is not possible due to the existence of the ABC, we infer that ϕ is satisfied. □

Remark 46. *Note that the number of lassos in $\mathcal{A}^b_{\neg\psi}$ is finite since the NBA has finitely many edges that lead to finitely many simple paths to an accepting state and simple cycles over the accepting state.*

Algorithm 3 Algorithm for verification of HyperLTL formulae

Require: $\mathfrak{S}, \phi = \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\psi, L$
  Construct NBA $\mathcal{A}^b_{\neg\psi}$ for $\neg\psi$
  Identify lassos $\mathcal{R}$ of $\mathcal{A}^b_{\neg\psi}$
  for $i \leftarrow 1$ to k **do**
    Identify consecutive transition pairs
    $\mathcal{S}_{\mathcal{A}^b_{\neg\psi}} = \{(s_{A_{i,1}}, s_{B_{i,1}}), \ldots, (s_{A_{i,v_i}}, s_{B_{i,v_i}})\}$
    for $j \leftarrow 1$ to v_i **do**
      $\chi_{i,j} \leftarrow \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\, \mathsf{G}(s_{A_{i,j}} \rightarrow \mathsf{G}(\neg s_{B_{i,j}}))$
  Construct augmented system $\mathfrak{S}^{\mathrm{P}}$
  Find common ABC $\mathbb{B}$ for $\chi_{i,j}$ for all $i \in \{1, \ldots, k\}$ and some $j \in \{1, \ldots, v_i\}$
  if $\mathbb{B}$ exists **then**
    return $T(\mathfrak{S}, L) \models \phi$
  else
    return Inconclusive

We summarize the sound verification procedure for the verification of HyperLTL specifications using Algorithm 3. Now, we illustrate the decomposition of $\mathcal{A}^b_{\neg\psi}$ presented in this section via Example 7.

Example 7 (Continued)**.** *For the room temperature regulation system $\mathfrak{S}$ described in* (5.14)*, we first construct the augmented system $\mathfrak{S} = \mathfrak{S} \times \mathfrak{S}$. Then, for the HyperLTL specification $\phi = \forall\pi_1\forall\pi_2(p_{3\pi_1} \wedge p_{4\pi_2}) \rightarrow \mathsf{G}(p_{1\pi_1} \wedge p_{\pi_2})$, we construct the NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to $\neg\psi$. This is obtained as shown in Figure 5.10. Then, we decompose $\mathcal{A}^b_{\neg\psi}$ into consecutive transition pairs as explained in this section. We notice that there is one lasso for $\mathcal{A}^b_{\neg\psi}$, resulting in one transition pair $((p_2, p_3), \neg(p_1, p_1))$. Computing an ABC for this transition pair allows determining whether $T(\mathfrak{S}, L) \models \phi$.*

Remark 47. *We note that the problem of finding the collection of consecutive transition pairs (thus conditional invariances), one from each lasso, which admits a common ABC is intractable. To show this, we first assume that we are given an oracle that determines whether a collection of consecutive transition pairs admits a common ABC. We then consider a relaxed version of this problem as follows. We assume that for any state r in $\mathcal{A}^b_{\neg\psi}$ with some incoming edge labeled s_A and outgoing edges $s_{B_1}, \ldots, s_{B_r}$, if there exists an ABC $\mathbb{B}$ for the pair (s_A, s_{B_j}) for some $1 \leq j \leq r$, then the function $\mathbb{B}$ acts as an ABC for every pair (s_A, s_{B_j}) for all $1 \leq j \leq r$. Then, the problem of finding a suitable collection of transition pairs is reduced to finding a collection of edges such that their removal causes the accepting states to not be reachable from the initial state. This corresponds to a cut [33] that partitions the accepting states from the initial state. To determine whether a cut allows for a common ABC, we must make use of the oracle, and in the worst case, we need to enumerate all possible cuts in $A^b_{\neg\psi}$. Since, the number of possible cuts is exponential in the number of edges of $A^b_{\neg\psi}$ [33], the problem is clearly intractable.*

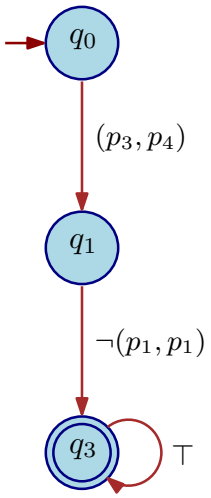


Figure 5.10: NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to $\neg\psi$.

The requirement of a common ABC for a collection of consecutive transition pairs is necessary to provide guarantees via Theorem 15. This is due to the fact that in condition (5.17) of Definition 30, existential quantifiers may precede the universal quantifiers depending on the HyperLTL specification. In such cases, Eloise does not have access to the full-state information of the augmented system and the choices made by Abelard in the turn-based game. However, Abelard has access to the states as well as Eloise's choices. Then, different ABCs for different transition pairs would imply that for each transition pair, Eloise picks a different strategy. This may lead to conflicts. For example, let us assume that the HyperLTL formula is of the form $\phi = \exists\pi_1\forall\pi_2\psi$, and consider two conditional invariances corresponding to pairs (s_{A_1}, s_{B_1}) and (s_{A_2}, s_{B_2}). The first component of the state of the augmented system is controlled by Eloise, and the second one by Abelard. Due to a lack of full-state information on the augmented system for Eloise, she is only able to observe the label of the first component and therefore may be unable to differentiate between s_{A_1} and s_{A_2}. Thus, having two different strategies corresponding to each of these pairs may result in ambiguity for Eloise. Moreover, picking the first strategy corresponding to (s_{A_1}, s_{B_1}) at state s_{A_2} could lead to Abelard choosing an input that violates the second conditional invariance corresponding to (s_{A_2}, s_{B_2}), and vice-versa as Abelard selects a trace after Eloise selects her trace. This results in a violation of the original specification. Unfortunately, even though a common ABC is necessary to provide verification guarantees, its existence may be difficult to find.

However, in specifications where Eloise has access to full state information and all of Abelard's choices, the requirement of a common ABC may be relaxed. This is especially true for specifications in the $\forall^*\exists^*$ fragment, where all the universal quantifiers precede the existential ones. In fact, the $\forall^*\exists^*$ fragment holds great importance as it comprises of many relevant security properties. For example, a variant of the noninterference property [83] requires that, for all traces, the low-security variables should not see any difference in observation when high-security variables are

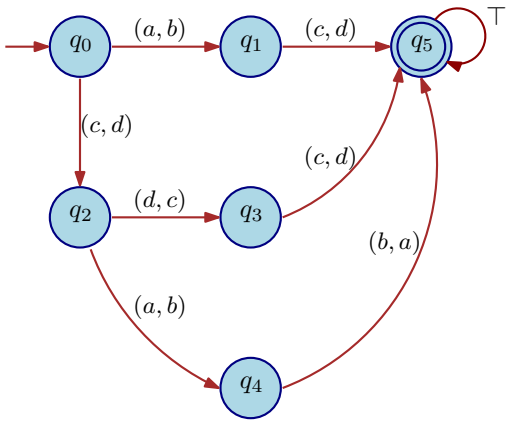


Figure 5.11: NBA $\mathcal{A}^b_{\neg\psi}$ for Example 8.

changed and replaced by dummy variables. This can be expressed by the HyperLTL specification

$$\forall\pi_1\exists\pi_2(\mathsf{G}\, h_{\pi_2}) \wedge \bigwedge_{l\in LS} l_{\pi_1} \leftrightarrow l_{\pi_2},$$

where h_{π_2} implies that the high security variables in π_2 are all set to a dummy variable p that is always true , and $LS \in \mathcal{AP}$ denotes the set of low security variables. Similarly, initial-state opacity specification [137] is also in the $\forall^*\exists^*$ fragment (see case study). Considering the importance of this fragment, we now provide a separate algorithm to allow for multiple ABCs for different lassos under some conditions.

5.4.4 Algorithm for $\forall^*\exists^*$- Fragment of HyperLTL

From the above discussion, it can be understood that specifications in the $\forall^*\exists^*$ fragment enable the relaxation of the common ABC requirement and allow for different ABCs in different lassos. In particular, Eloise can take advantage of the full state information of the augmented system available to her as well as the knowledge of Abelard's choices to use different ABCs for different consecutive transition pairs in every lasso. However, to do so, one must take the structure of the automata $\mathcal{A}^b_{\neg\psi}$ into consideration, as in the presence of states with two or more outgoing edges, there may be ambiguity for Eloise in selecting strategies. Moreover, in the presence of non-determinism in the automaton, Eloise may fail to select strategies due to a lack of information on the history of visited states. These challenges are demonstrated in the following examples.

Example 8 (States with a fork)**.** *In this example, we show the issue of utilizing multiple ABCs in the presence of a state with multiple outgoing edges. Consider the NBA $\mathcal{A}^b_{\neg\psi}$ shown in Figure 5.11 constructed from a set of atomic propositions $\mathcal{AP} = \{a, b, c, d\}$ corresponding to some HyperLTL specification $\phi = \forall\pi_1 \ldots \forall\pi_l\exists\pi_{l+1} \ldots \exists\pi_p\psi$.*

From $\mathcal{A}^b_{\neg\psi}$, we can identify $k = 3$ lassos as

$$\mathcal{R} = \{\tilde{\mathbf{q}}_1 = (q_0, q_1, q_5, q_5), \tilde{\mathbf{q}}_2 = (q_0, q_2, q_3, q_5, q_5),$$
$$\tilde{\mathbf{q}}_3 = (q_0, q_2, q_4, q_5, q_5)\}.$$

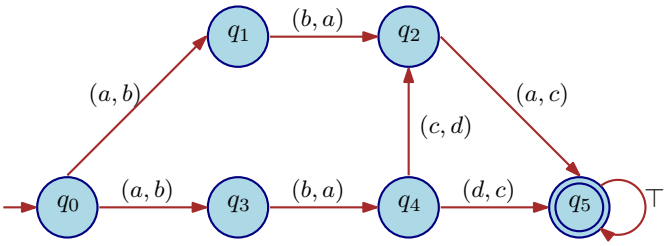


Figure 5.12: NBA $\mathcal{A}^b_{\neg\psi}$ for Example 9.

For every $\tilde{\mathbf{q}} \in \mathcal{R}$*, we obtain the consecutive transition pairs as*

$$\mathcal{S}(\tilde{\mathbf{q}}_1) = \{((a,b),(c,d)),((c,d),\top)\},$$
$$\mathcal{S}(\tilde{\mathbf{q}}_2) = \{((c,d),(d,c)),((d,c),(c,d)),((c,d),\top)\},$$
$$\mathcal{S}(\tilde{\mathbf{q}}_3) = \{((c,d),(a,b)),((a,b),(b,a)),((b,a),\top)\}.$$

Naturally, it is preferable to obtain different ABCs for at least one transition pair in every lasso to guarantee the satisfaction of the specification. However, this might cause problems for lassos $\tilde{\mathbf{q}}_2$ *and* $\tilde{\mathbf{q}}_3$*, where there are two outgoing edges from a single state* q_2*. This leads to two different transition pairs* $((c,d),(d,c))$ *and* $((c,d),(a,b))$*. Having different ABCs for these pairs would result in different winning strategies for Eloise to avoid the sets corresponding to* (d,c) *and* (a,b)*, from the set corresponding to* (c,d)*, respectively. Choosing the first ABC and its corresponding strategy could lead to the violation of condition* (5.17) *for the second ABC and vice versa. However, the existence of a common ABC for both the pairs guarantees that Eloise has a winning strategy to avoid both* (d,c) *and* (a,b) *if she encounters a state corresponding to* (c,d)*. Therefore, for this specification, one would require to obtain a common ABC for the pairs* $((c,d),(d,c))$ *and* $((c,d),(a,b))$ *from lassos* $\tilde{\mathbf{q}}_2$ *and* $\tilde{\mathbf{q}}_3$*, respectively, and a different ABC may be obtained for the pair* $((a,b),(c,d))$ *from the lasso* $\tilde{\mathbf{q}}_1$*. However, if such a common ABC cannot be found, one can consider other transition pairs in* $\tilde{\mathbf{q}}_2$ *and* $\tilde{\mathbf{q}}_3$*, and in that case, different ABCs may be used.*

Example 9 (Non-determinism)**.** *In this example, we show the issue of using multiple ABCs in the presence of non-determinism in the automaton. Consider the NBA* $\mathcal{A}^b_{\neg\psi}$ *shown in Figure 5.12 constructed from a set of atomic propositions* $\mathcal{AP} = \{a,b,c,d\}$ *corresponding to some HyperLTL specification* $\phi = \forall\pi_1 \ldots \forall\pi_l \exists\pi_{l+1} \ldots \exists\pi_p \psi$*. From* $\mathcal{A}^b_{\neg\psi}$*, we can identify* $k = 3$ *lassos as*

$$\mathcal{R} = \{\tilde{\mathbf{q}}_1 = (q_0,q_1,q_2,q_5,q_5), \tilde{\mathbf{q}}_2 = (q_0,q_3,q_4,q_5,q_5)$$
$$\tilde{\mathbf{q}}_3 = (q_0,q_3,q_4,q_2,q_5,q_5)\}.$$

For every $\tilde{\mathbf{q}} \in \mathcal{R}$*, we obtain the consecutive transition pairs as*

$$\mathcal{S}(\tilde{\mathbf{q}}_1) = \{((a,b),(b,a)),((b,a),(a,c)),((a,c),\top)\},$$
$$\mathcal{S}(\tilde{\mathbf{q}}_2) = \{((a,b),(b,a)),((b,a),(d,c)),((d,c),\top)\},$$
$$\mathcal{S}(\tilde{\mathbf{q}}_2) = \{((a,b),(b,a)),((b,a),(c,d)),((c,d),(a,c)),$$
$$((a,c),\top)\}.$$

Consider lassos $\tilde{\mathbf{q}}_1$, $\tilde{\mathbf{q}}_2$ and $\tilde{\mathbf{q}}_3$, where there is a non-deterministic transition from the initial state q_0 to the states q_1 and q_3 under the label (a, b). Ideally, a single ABC for the pair $((a, b), (b, a))$ would effectively disallow the transitions in all the lassos $\tilde{\mathbf{q}}_1$, $\tilde{\mathbf{q}}_2$ and $\tilde{\mathbf{q}}_3$. However, the problem arises when such an ABC cannot be found. In order to guarantee the satisfaction of the specification, other transition pairs in the lassos must be disallowed. Now, in $\tilde{\mathbf{q}}_1$, there is a transition from (b, a) to (a, c), while in $\tilde{\mathbf{q}}_2$ and $\tilde{\mathbf{q}}_3$, there are two transitions from (b, a) to (d, c) and (c, d), respectively. At any point in time, Eloise cannot uniquely determine the history of the states visited in $\mathcal{A}^b_{\neg\psi}$. As a result, after a nondeterministic transition from (a, b) to (b, a), Eloise has no way of knowing whether to block further transitions from (b, a) to (a, c), or from (b, a) to (d, c) and (c, d). Therefore, the approach of using different ABCs fails in the presence of non-determinism.

Unfortunately, problems arising due to non-determinism cannot be directly resolved. Therefore, to circumvent this issue, we instead consider the automaton $\mathcal{A}^b_{\neg\psi}$ to be deterministic. In particular, for our exposition, we focus on the case where $\mathcal{A}^b_{\neg\psi}$ is a deterministic Büchi automaton (DBA).

The general verification procedure for determining whether Eloise has a strategy to ensure that the acceptance condition of $\mathcal{A}^b_{\neg\psi}$ is violated for specifications in the $\forall^*\exists^*$ fragment is provided in Algorithms 4 and 5. Having a DBA $\mathcal{A}^b_{\neg\psi} = (Q, q_0, \Sigma^{\mathrm{p}}, \delta, F)$, we first identify all the lassos $\mathcal{R} = \{\tilde{\mathbf{q}}_1, \ldots, \tilde{\mathbf{q}}_k\}$ that reach and cycle on some state in F. We then prune $\mathcal{A}^b_{\neg\psi}$ and remove any states that are not in the lassos and all transitions to and from such states. Then, beginning from the initial state, for any label s_A, we identify the state that is reachable via s_A. Let the outgoing transitions from this state be $s_{B_1}, \ldots, s_{B_n}$. Let $S = \{(s_A, s_{B_1}), \ldots, (s_A, s_{B_n})\}$. A suitable common ABC is searched for all the transition pairs in a set $S_a \subseteq S$. For every pair $(s_A, s_{B_j}) \in S_a$, for all $0 \leq j \leq n$, we also have that $(s_A, s_{B_j}) \in \mathcal{S}(\tilde{\mathbf{q}}_i)$ for some $0 \leq i \leq k$. If such a common ABC exists, then such lassos $\tilde{\mathbf{q}}_i$ can be discarded from further consideration as the existence of ABCs disallows the transitions in those lassos and are collected in the set $\mathcal{R}_d$. Note that, unfortunately, there is no systematic way to obtain the set $S_a \subseteq S$ consisting of all transition pairs that admit a common ABC. S_a is first picked in a trial-and-error fashion and then an *oracle* (see Remark 47) is used to determine if the transition pairs in S_a admit a common ABC.

This procedure is then repeated for every transition label at the initial state, and the discarded lassos are iteratively added to $\mathcal{R}_d$. Once all the outgoing transition labels are covered, we move on to the next state reachable from the initial state and repeat the procedure to find ABCs for only those pairs that belong to $\mathcal{S}(\tilde{\mathbf{q}}_i)$ such that $\tilde{\mathbf{q}}_i \in \mathcal{R}\backslash\mathcal{R}_d$. This continues in a breadth-first search fashion until all the lassos are discarded, *i.e.*, $\mathcal{R}_d = \mathcal{R}$, in which case we can conclude that $T(\mathfrak{S}, L) \models \phi$, or all the states of $\mathcal{A}_{\neg\psi}$ have been considered. If $\mathcal{R}_d \subset \mathcal{R}$, it means that there are lassos for which no ABC could be found, rendering the verification procedure inconclusive. Algorithms 4 and 5 illustrate the procedure for finding ABCs systematically for the specifications in the $\forall^*\exists^*$ fragment of HyperLTL.

Algorithm 4 Algorithm for verification of $\forall^*\exists^*$ fragment of HyperLTL

Require: $\mathfrak{S}, \phi = \forall\pi_1 \ldots \forall\pi_l \exists\pi_{l+1} \ldots \exists\pi_{\mathrm{P}}\psi, L$
  Construct DBA $\mathcal{A}^b_{\neg\psi}$ for $\neg\psi$
  Identify lassos $\mathcal{R} := \{\tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2, \ldots, \tilde{\mathbf{q}}_k\}$ of $\mathcal{A}^b_{\neg\psi}$
  $R_d \leftarrow \varnothing$
  $\mathcal{A}'^b_{\neg\psi} := (Q', q_0, \Sigma^p, \delta', F') \leftarrow Prune(\mathcal{A}^b_{\neg\psi})$
  $visited \leftarrow [0, \ldots 0]$ ▷ Array of size $|Q'|$
  $V \leftarrow \{q_0\}$
  $visited[q_0] \leftarrow 1$
  while $V \neq \varnothing$ **do**
    for each $q \in V$ **do**
      $\mathcal{R}_d \leftarrow \mathcal{R}_d \cup ABC\_FIND(\mathcal{A}'^b_{\neg\psi}, q)$
      $\mathcal{R}_m \leftarrow \mathcal{R} \setminus \mathcal{R}_d$
      if $\mathcal{R}_m = \varnothing$ **then**
        return $T(\mathfrak{S}, L) \models \phi$
      for each $s_A \in \Sigma^{\mathrm{P}}$ **do**
        $q' \leftarrow \delta'(q, s_A)$
        $G \leftarrow \{\tilde{\mathbf{q}} \in \mathcal{R}_m \mid (q, q') \in \tilde{\mathbf{q}}\}$
        if $G \neq \varnothing$ and $visited[q'] < 1$ **then**
          $V \leftarrow V \cup \{q'\}$
          $visited[q'] \leftarrow visited[q'] + 1$
      $V \leftarrow V \setminus \{q\}$
  return Inconclusive

5.4.5 Computation of Augmented Barrier Certificates

In the previous subsections, we showed that ABC satisfying conditions (5.15)-(5.17) for a transition pair (s_A, s_B) is vital to verify that a system $\mathfrak{S}$ satisfies a desired HyperLTL specification ϕ. In this subsection, we focus on suitable synthesizing these ABCs. This can be done under some minor assumptions on the considered ABCs as well as the dynamics of the system. Specifically, we see that when the dynamics of the systems are restricted to polynomial functions and the state set X, exogenous input set W as well as the safe and unsafe sets obtained from (s_A, s_B) are semi-algebraic sets, one can utilize sum-of-squares (SOS) programming techniques [93] to compute polynomial ABCs of predefined degrees. We now formally state the following assumption. Note that this assumption is similar to Assumption 6 presented in the previous chapters, and is stated as follows.

Assumption 9. *The system $\mathfrak{S}$ has a continuous state set $X \subseteq \mathbb{R}^n$ and continuous exogenous input set $W \in \mathbb{R}^p$, and its transition function $f : X \times W \to X$ is a polynomial function of the state x and input w.*

Under Assumption 9, one can readily observe that the state and input sets of the augmented system $\mathfrak{S}^{\mathrm{P}}$ (i.e. X^{P} and W^{P}, respectively) are also continuous, and the function $f^{\mathrm{P}} : X^{\mathrm{P}} \times W^{\mathrm{P}} \to X^{\mathrm{P}}$

Algorithm 5 Function ABC_FIND

Require: $\mathcal{A}'^{b}_{\neg\psi}, q$
  $R_d \leftarrow \emptyset$
  for each $s_A \in \Sigma^{\mathrm{P}}$ **do**
    $S \leftarrow \emptyset$
    $q' \leftarrow \delta'(q, s_A)$
    for each $s_B \in \Sigma^{\mathrm{P}}$ **do**
      if $\delta'(q', s_B) \neq \emptyset$ **then** $S \leftarrow S \cup \{(s_A, s_B)\}$
    Find a common ABC for a set $S_a \subseteq S$.
    for each $(s_A, s_B) \in S_a$ **do**
      $q' \leftarrow \delta'(q, s_A)$
      $q'' \leftarrow \delta'(q', s_B)$
      $\mathcal{R}_d \leftarrow \mathcal{R}_d \cup \{\mathbf{q} \in \mathcal{R} \mid (q, q', q'') \in \tilde{\mathbf{q}}\}$
  return $\mathcal{R}_d$

is a P-tuple of polynomial functions. Having this, one can then reformulate conditions (5.15)-(5.17) as an SOS optimization problem to search for a polynomial ABC for augmented system $\mathfrak{S}^{\mathrm{P}}$. In order to present the result below, we assume that the number of quantifiers "$\exists$" in $\phi = \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\psi$ is equal to k and define $I_\exists = \{i \mid \mu_i = \exists, 1 \leq i \leq \mathrm{P}\}$.

Lemma 16. *Suppose Assumption 9 holds and sets X^p, A, B, and W^p are defined as $X^p = \{\tilde{x} \in \mathbb{R}^{np} \mid g(\tilde{x}) \geq 0\}$, $A = \{\tilde{x} \in \mathbb{R}^{np} \mid g_0(\tilde{x}) \geq 0\}$, $B = \{\tilde{x} \in \mathbb{R}^{np} \mid g_u(\tilde{x}) \geq 0\}$, and $W^p = \{\tilde{w} \in \mathbb{R}^{mp} \mid g_{in}(\tilde{w}) \geq 0\}$, where the inequalities are considered component-wise and functions g, g_0, g_u, and g_{in} are polynomials. Suppose there exist a polynomial $\mathbb{B}(\tilde{x})$ and k polynomials $h^i_j(\hat{x}_i, \hat{w}_i)$, $i \in I_\exists$, corresponding to the j^{th} entry of $w_i = [w_{i_1}; \ldots; w_{i_m}] \in W \subseteq \mathbb{R}^m$, where $\hat{x}_i$ refers to those components of the state with indices less than i and $\hat{w}_i$ denotes the inputs associated with "$\forall$" quantifiers with indices less than i. In addition, suppose there exist sum-of-squares polynomials $\lambda(\tilde{x}, \tilde{w}), \lambda_0(\tilde{x}), \lambda_u(\tilde{x})$, and $\lambda_{in}(\tilde{x}, \tilde{w})$ of appropriate dimensions, such that the following expressions are sum-of-square polynomials:*

$$-\mathbb{B}(\tilde{x}) - \lambda_0(\tilde{x})g_0^T(\tilde{x}), \tag{5.18}$$

$$\mathbb{B}(\tilde{x}) - \lambda_u(\tilde{x})g_u^T(\tilde{x}) - \epsilon, \tag{5.19}$$

$$-\mathbb{B}(f^p(\tilde{x}, \tilde{w})) + \mathbb{B}(\tilde{x}) - \lambda(\tilde{x}, \tilde{w})g^T(\tilde{x}) - \lambda_{in}(\tilde{x}, \tilde{w})g_{in}^T(\tilde{w}) - \sum_{i \in I_\exists}\sum_{j=1}^{P}(w_{i_j} - h^i_j(\hat{x}_i, \hat{w}_i)), \tag{5.20}$$

where ϵ is a small positive number. Then, $\mathbb{B}(\tilde{x})$ is an ABC from set A to set B satisfying conditions (5.15)-(5.15).

Proof. Since $\lambda_0(\tilde{x})$ is an SOS polynomial, we have that $\lambda_0(\tilde{x})g_0^T(\tilde{x})$ is non-negative over A. Therefore, if condition (5.18) is an SOS polynomial, and therefore non-negative, it would directly imply condition (5.15). Similarly, the SOS constraint (5.19) implies condition (5.16). Now we

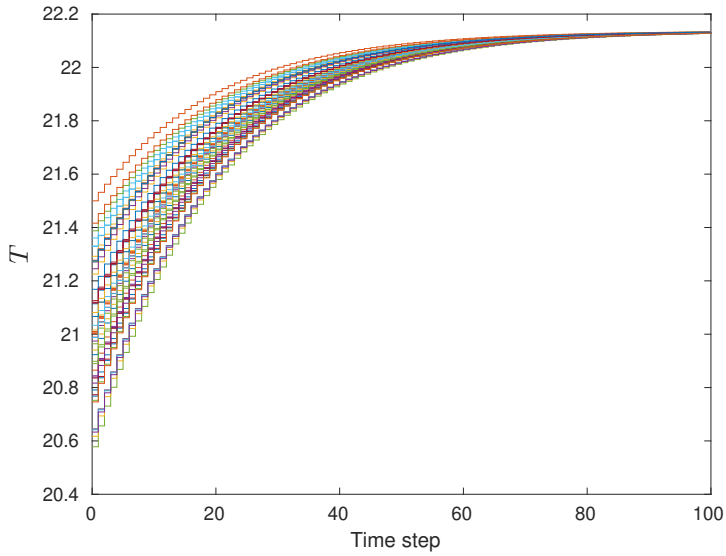


Figure 5.13: State runs of $\mathfrak{S}$ starting from initial set X^3.

show that condition (5.20) implies (5.17). By selecting inputs $w_{ij} = h^i_j(\hat{x}_i, \hat{w})$, the last term in (5.20) vanishes. Since the expression $\lambda(\tilde{x}, \tilde{w})g^T(\tilde{x})$ is non-negative over X^{P} and $\lambda_{in}(\tilde{x}, \tilde{w})g^T_{in}(\tilde{w})$ is non-negative over W^p, we have that for all $\tilde{x} \in X^{\mathrm{P}}$, $-\mathbb{B}(f^{\mathrm{P}}(\tilde{x}, \tilde{w})) + \mathbb{B}(\tilde{x}) \geq 0$. This implies that condition (5.17) holds, thus concluding the proof. □

Example 7 (Continued)**.** *We now utilize SOS programming to compute ABC for the transition pair* $((p_2, p_3), \neg(p_1, p_1))$ *obtained from Figure 5.10. We use the tools SOSTOOLS [98] and SeDuMi [119] on MATLAB to compute a polynomial ABC of degree* 2 *as* $\mathbb{B}(T_1, T_2) = 1.2454T_1^2 - 1.6722T_1T_2 - 18.5791T_1 + 1.1656T_2^2 - 14.9555T_2 + 377.4684$ *with a tolerance of* $\varepsilon = 0.001$*. The existence of the ABC proves that the safety controller designed for the system* $\mathfrak{S}$*, is indeed robust with respect to initial-state uncertainty with a robustness measure of* $\delta = 0.5$*. Figure 5.13 shows that the state runs obtained for* $\mathfrak{S}$ *remains in the safe set* $X^1 = [20, 25]$ *when starting from the initial set* $X^3 = [20.5, 21.5]$ *which captures uncertainties in the initial state. We performed these computations on a machine running Linux Ubuntu OS (Intel i7-8665U CPU with 32GB RAM) and it took around* 19 *seconds for the computation of ABC.*

Complexity Analysis. Suppose there exists a common ABC of degree $2d$ that verifies a system $\mathfrak{S}$ against a HyperLTL formula $\phi = \mu_1\pi_1 \ldots \mu_{\mathrm{P}}\pi_{\mathrm{P}}\psi$, consisting of P trace quantifiers. To find this ABC, one requires to consider all the consecutive transition pairs of the NBA $\mathcal{A}_{\neg\psi}$ corresponding to the specification, and then select suitable consecutive pairs in every lasso such that they admit a common ABC. Let the number of states of $\mathcal{A}^b_{\neg\psi}$ be $|Q|$. There are $O(2^{|Q|^2})$ possible subsets of consecutive transition pairs for $\mathcal{A}^b_{\neg\psi}$. Then, for each such subset, the common ABC needs to be computed over the P-fold augmented system with dimension $\mathrm{P}n$ via SOS programming, resulting in a polynomial complexity given by $O\left(\binom{\mathrm{P}n+d}{d}^2\right)$ [93]. Therefore, the final complexity of our approach is polynomial in $O\left(2^{|Q|^2}\binom{\mathrm{P}n+d}{d}^2\right)$.

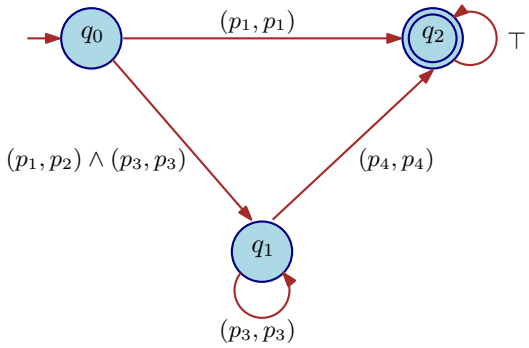


Figure 5.14: NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to $\neg\psi$.

5.4.6 Case Study

In this example, we consider the discrete-time, two-dimensional model of an autonomous vehicle on a single-lane road, with state variables as $x = [s, v]$, where s denotes the absolute position of the vehicle and v denotes the absolute velocity. The dynamics of the system are borrowed from [77] and governed by:

$$\mathfrak{S} : \begin{cases} s(t+1) = s(t) + \Delta\tau v(t) + \frac{\Delta\tau^2}{2} w(t), \\ v(t+1) = v(t) + \Delta\tau w(t), \end{cases} \tag{5.21}$$

where w is the exogenous input, *i.e.*, acceleration, and $\Delta\tau = 1$ is the sampling time. Here, we verify the ρ-approximate initial state opacity property [77] for this system. The specification requires that, for any state run of the system that begins from a secret state, there must exist another state run that begins from a non-secret state such that both state runs render ρ-close observations from the observer's (or intruder's) point of view. The significance of the specification can be motivated with the help of a simple example. Consider a scenario where the vehicle is assigned for a cash transit from a high-security bank to an ATM machine, and the initial locations of the vehicle must be kept secret. It is assumed that a malicious intruder is observing the velocity of the vehicle remotely and intends to gain access to the secret information and perform an attack. Therefore, it is critical to ensure that the secret states of the system are never revealed to the intruder. This security specification can be modelled as a ρ-approximate initial-state opacity problem, where $\rho \geq 0$ captures the measurement precision of the intruder.

To express ρ-approximate initial-state opacity as a HyperLTL specification, consider system (5.21) with state set $X = [0, 8] \times [0, 0.6]$ and exogenous input set $W = [-0.04, 0.04]$. The secret set is defined by $X^1 = [0, 1] \times [0, 0.6]$ and the non-secret set is consequently given by $X^2 = X \backslash X^1$. Here, we assume that the intruder can only observe the velocity of the car with a precision of δ, *i.e*, observations of two states $x_1 = [s_1, v_1] \in X$ and $x_2 = [s_2, v_2] \in X$ appear identical to the intruder if $\|v_1 - v_2\| \leq \rho$. We now construct atomic propositions as $\mathcal{AP} = \{p_1, p_2, p_3, p_4\}$ where p_1 and p_2 are such that $L(x \in X^z) = a_z$ for $z = \{1, 2\}$. The atomic propositions p_3 and p_4 are constructed over the augmented state set such that we have $(a_3, a_3) := \{(L(x_1{=}[s_1, v_1] \in X), L(x_2{=}[s_2, v_2] \in X)) \mid \|v_1 - v_2\|^2 \leq \rho^2\}$ and $(a_4, a_4) :=$

$\{(L(x_1{=}[s_1, v_1] \in X), L(x_2{=}s_2, v_2] \in X)) \mid \|v_1 - v_2\|^2 \geq \rho^2 + \bar{\epsilon}\}$, where $\bar{\epsilon}$ is a small positive number introduced to certify positivity using SOS programming. Note that atomic propositions for HyperLTL specifications are usually defined over a single system rather than the augmented one. On the other hand, the ρ-approximate initial state opacity specification requires the atomic propositions to capture the ρ-closeness between any two states of the augmented system. In a finite-state system, one could quantify ρ-closeness by using finite conjuncts of atomic propositions defined over the original system, but in the infinite-state case such as ours, that is not possible. Therefore, to handle this non-trivial case, we modify atomic propositions slightly and define them over the augmented state set. Such modifications can be made without any loss of generality in our approach. Now, one can formulate the ρ-approximate initial-state opacity specification as a HyperLTL formula given by $\phi = \forall\pi_1\exists\pi_2\psi$, where $\psi = p_{1\pi_1} \rightarrow (p_{2\pi_2} \wedge \mathsf{G}(p_{3\pi_1} \wedge a_{p\pi_2}))$.

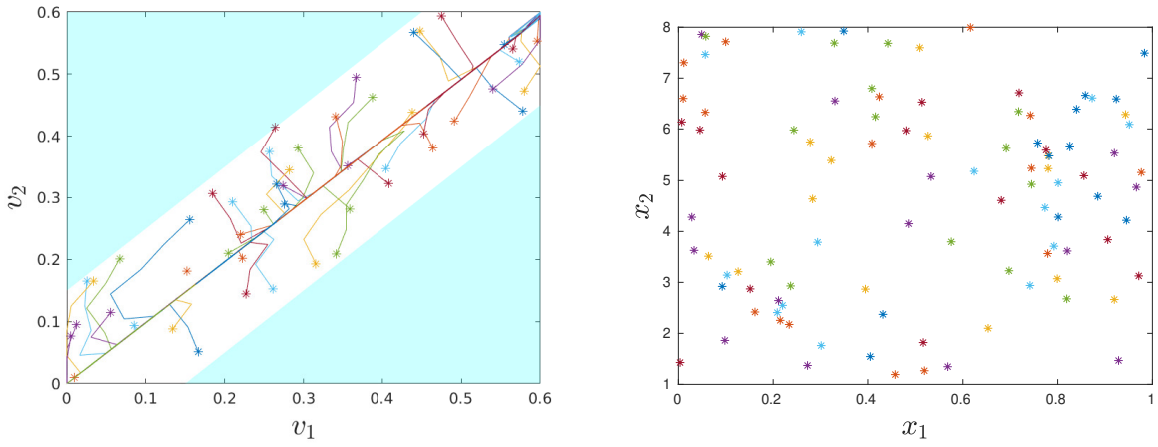
Consider the system $\mathfrak{S}^2{=}\mathfrak{S} \times \mathfrak{S}$ with states $(x_1{=}[s_1, v_1], x_2{=}[s_2, v_2]) \in X^2$ and input $(w_1, w_2) \in W^2$, and the NBA $\mathcal{A}^b_{\neg\psi}$ corresponding to $\neg\psi$ that is obtained as shown in Figure 5.14. We decompose $\mathcal{A}^b_{\neg\psi}$ to obtain transition pairs for all lassos. This is obtained as $((p_1, p_2) \wedge (p_3, p_3), (p_4, p_4))$, $((p_1, p_1), \top)$ and $((p_4, p_4), \top)$. The latter two do not admit ABC following Remark 45, and the transition pair $((p_1, p_1), \top)$ is ignored by assuming that the augmented system $\mathfrak{S}^2$ never starts from an initial condition corresponding to $\tilde{p}_1 = (p_1, p_1)$. Note that this assumption is only on the *virtual* copy of the system $\mathfrak{S}$ and does not restrict the initial states of the original system $\mathfrak{S}$ directly. For the transition pair $((p_1, p_2) \wedge (p_3, p_3), (p_4, p_4))$, we compute a suitable ABC by considering $\rho = 0.15$. Using SOSTOOLS and SeDuMi tools on MATLAB, and with tolerance parameters $\bar{\epsilon} = 0.01$ and $\epsilon = 0.015$, we obtain ABC as follows.

$$\begin{aligned}\mathbb{B}((s_1, v_1), (s_2, v_2)) &= 85.03v_1^2 - 170.3v_1v_2 + 0.0048v_1s_1\\ &- 0.0065v_1s_2 + 0.0413v_1 + 85.24v_2^2 - 0.004784v_2s_1\\ &+ 0.0063v_2s_2 - 0.0121v_2 + 0.0059s_1^2 - 0.0119s_1s_2\\ &+ 0.0241s_1 + 0.0061s_2^2 - 0.0825s_2 - 2.076,\end{aligned}$$

and the corresponding $\exists$ quantifier on the input is fulfilled by $w_2(s_1, v_1, s_2, v_2, w_1) = 0.983v_1 - v_2 + w_1$. Therefore, we conclude that the system $\mathfrak{S}$ satisfies the HyperLTL specification ϕ representing ρ-approximate initial-state opacity problem with $\rho = 0.15$. Figure 5.15a shows the projection of a few state runs on the velocity coordinate of the augmented system $\mathfrak{S}$, with initial conditions in $A = L^{\mathrm{P}-1}((p_1, p_2) \wedge (p_3, p_3))$. Figure 5.15b shows the initial conditions projected on the position coordinate. It follows that the state runs avoid reaching the unsafe regions, indicating that the original system is ρ-approximate initial-state opacity. We should add that the computation of ABCs using the mentioned tools on MATLAB takes roughly 35 seconds on a machine running with Linux Ubuntu OS (Intel i7-8665U CPU with a 32 GB of RAM).

5.5 Conclusion

This chapter was concerned with extending (control) barrier certificate-based methods beyond safety and reachability specifications. In particular, the aim of this chapter was to provide



(a) (b)

Figure 5.15: (a) State runs of $\mathfrak{S}^2$ projected over the velocity coordinate. The region in blue indicates the unsafe set (b) The initial conditions of the state runs projected over the position coordinate marked by $*$ which show that the first initial condition (i.e. x_1) is secret and the other one (i.e. x_2) is non-secret.

(probabilistic) guarantees for the satisfaction of different classes of specifications, namely, linear temporal logic specifications over finite time horizons or specifications that can be expressed by deterministic finite automata, ω-regular specifications described using deterministic Streett automata, as well as more general hyperproperties that can be described using HyperLTL. To tackle such complex logic specifications, we proposed automata-theoretic approaches that relied on the decomposition of the complex task into a collection of smaller tasks that could be handled easily via (control) barrier certificates. Then, the (probabilistic) guarantees that were obtained for these smaller tasks were utilized to provide guarantees for the original specification.

In the first two sections of the chapter, we tackled the controller synthesis problem for (interconnected) stochastic control systems against trace properties, *i.e.*, properties defined over individual execution traces, over (in)finite time horizons. Specifically, in the first section, we focused on the finite time horizon controller synthesis problem against specifications described using deterministic finite automata and tackled the problem by decomposing the automata corresponding to the negation of the specifications into smaller safety tasks. Then, c-martingale control barrier certificates were used to provide probabilistic guarantees over these safety tasks over finite time horizons. By combining these guarantees, we obtained the overall lower bound on the probability of satisfaction of the original specification. In comparison, the second section was focused on the infinite time horizon controller synthesis against specifications described using deterministic Streett automata. In this case, instead of working with the complement of the specification, we directly decomposed the original specification into smaller safety synthesis tasks. Then, supermartingale control barrier certificates were used to provide probabilistic guarantees over the smaller safety tasks over infinite time horizons. Then, we combined these

guarantees to obtain the probability with which the system satisfied the original specification. In both these cases, we demonstrated that computing different CBCs and corresponding controllers for each individual safety task can result in ambiguity when deploying the controllers in a closed-loop fashion. To prevent this, we proposed a switching controller structure when the controller is dependent on the state of the system as well as the state of the automaton concerned with the specification. Finally, we demonstrated our approaches by utilizing the case studies from Chapter 3 and extending the controller synthesis problem beyond safety specifications.

The third section was mainly concerned with the formal verification of non-stochastic dynamical systems with exogenous inputs. Here, we considered the verification against hyperproperties that were expressed using HyperLTL. Focusing on hyperproperties expressible as HyperLTL formulae, we presented an implicit automata-theoretic approach. In our approach, the specifications were reduced to a collection of conditional invariance properties by utilizing an implicitly quantified Büchi automata corresponding to the complements of the specifications. We were able to devise a notion of augmented barrier certificates over the self-composition of the original system as a certificate of conditional invariance. The existence of ABCs is sufficient proof that the conditional invariance holds, this provides a verification guarantee over the satisfaction of the hyperproperty. For a general HyperLTL specification, we showed that a common ABC for at least one conditional invariance in every lasso is required to provide verification guarantees. However, for a HyperLTL specification in the $\forall^*\exists^*$ fragment, we provided a systematic algorithmic procedure that leverages the structure of the automata to allow for different ABCs for different lassos. We exploited a sum-of-squares approach to efficiently compute suitable ABCs. We also demonstrated our approach by utilizing two case studies to verify relevant hyperproperties, namely, robustness and opacity, respectively.

Chapter 6

Conclusion

This thesis tackled the formal verification and/or synthesis of (possibly stochastic) control systems against safety, reachability and complex logic specifications via inductive approaches by utilizing (control) barrier certificates. We demonstrated the several challenges encountered by barrier certificate-based techniques such as scalability, conservatism, and tackling complex specifications, and proposed solutions to alleviate these challenges in a systematic manner. We now conclude the thesis by reviewing the contributions made, discussing the shortcomings of the research, and providing some insights on possible solutions and future work.

6.1 Summary

The first part of the thesis tackled the scalability issues presented by control barrier certificate-based approaches and consequently provided a controller synthesis approach for ensuring probabilistic safety satisfaction of large-scale stochastic control systems. Scalability challenges for constructing control barrier certificates were alleviated by considering large-scale stochastic control systems as interconnected ones consisting of smaller subsystems and then utilizing a compositional framework for synthesising control barrier certificates and corresponding safe controllers by analyzing the subsystems instead of interconnected systems as a whole. Two different compositional approaches based on existing theories, such as small-gain theory and dissipativity theory, were discussed. First, by proposing a notion of control sub-barrier certificates for subsystems and then using max-type small-gain compositionality conditions, we constructed control barrier certificates and controllers compositionally to provide probability lower bounds on the satisfaction of safety specifications over finite time horizons. We also discussed two different approaches based on sum-of-squares optimization and counterexample-guided inductive synthesis for the computation of suitable control sub-barrier certificates. Secondly, a compositional framework utilizing dissipativity-type compositionality conditions was proposed to compute probability lower bounds for safety specifications over infinite time horizons. We showed that the computation of control sub-barrier certificates and corresponding local controllers for the subsystems using a dissipativity-based approach can be performed in a systematic manner by utilizing a distributed optimization method based on the alternating direction method of multipli-

ers. This enables to search for control sub-barrier certificates with respect to the satisfaction of compositionality conditions. Finally, we briefly compared the two compositionality frameworks and showed the applicability of our results by utilizing several case studies.

The second part of the thesis tackled the conservatism of the barrier certificate-based conditions in the context of verification of stochastic and non-stochastic dynamical systems. We showed that, by utilizing k-induction rather than standard induction to define the barrier certificate conditions, one can obtain weaker conditions that are easier to satisfy, thus making the search for barrier certificates easier. First, we focused on the safety verification of non-stochastic dynamical systems and proposed two different notions of k-inductive barrier certificates to provide weaker alternatives to traditional barrier certificate conditions. We illustrated via finite state system examples the benefits of using our proposed k-inductive barrier certificates over standard ones. We also compared the effectiveness of the two proposed notions and demonstrated via simple examples that the second notion of k-inductive barrier certificates is more expressive than the first. We also demonstrated the computation of k-inductive barrier certificates via sum-of-squares programming and satisfiability modulo theory solvers. Secondly, we considered the probabilistic safety verification problem for stochastic dynamical systems via k-induction-based conditions and proposed a new definition of k-inductive barrier certificates for the same. In particular, we showed that by using k-inductive barrier certificates, one does not need to impose a supermartingale condition on the barrier certificate, while still providing probability lower bounds on safety satisfaction. Lastly, we also provided reachability guarantees via k-inductive barrier certificates for stochastic dynamical systems. In this case, we presented two different notions of k-inductive barrier certificates. While one definition was used to obtain lower bounds on the probability of satisfying reach-and-avoid specifications, the other definition was used to ensure the satisfaction of reachability specifications with probability one. In both these cases, we showed via examples the benefits of k-inductive barrier certificates over standard ones. We also provided a sum-of-squares approach for the computation of k-inductive barrier certificates. All of our proposed notions were demonstrated by applying them to suitable case studies.

The last part of the thesis was concerned with applying barrier certificate-based approaches to analyze complex logic specifications beyond safety and reachability. Specifically, we proposed automata-theoretic approaches for analyzing different classes of logic specifications, such as those expressed by linear temporal logic or (in)finite traces over automata, as well as hyperproperties. First, we considered the controller synthesis problem for stochastic control systems against specifications expressed by deterministic finite automata over finite time horizons. We then utilized the automata corresponding to the negation of the specifications and decomposed them into sequential safety synthesis problems. By utilizing control barrier certificates, we were able to obtain controllers along with the probability upper bounds of violating these safety specifications. These probability bounds were then combined to obtain an overall probability lower bound on the satisfaction of the original specifications. Correspondingly, a switching controller structure was proposed for ensuring the satisfaction of specifications with the obtained probability bounds. Secondly, we proposed a similar automata-theoretic approach for synthesizing controllers for stochastic control systems against ω-regular specifications characterized by deterministic Streett automata over infinite time horizons. In this case, the automata corresponding to the specifications were directly decomposed into safety specifications, and the safety tasks were solved using control

barrier certificates. The probability bounds obtained for the safety tasks were combined to obtain an overall lower bound on the probability of satisfying the original specifications. Then, a switching controller structure was proposed to ensure the probabilistic satisfaction of ω-regular specifications.

Finally, we worked on the problem of formal verification of non-stochastic dynamical systems against hyperproperties. Focusing on hyperproperties that can be expressed using HyperLTL, we provided an automata-theoretic approach to reduce the overall verification problem into a collection of conditional invariance specifications. This was done by considering lassos (*i.e.* simple paths plus simple cycles to the accepting state) of the automata corresponding to the specifications. The conditional invariance specifications were then guaranteed by using augmented barrier certificates that were constructed on the self-composition of the system. Existence of common augmented barrier certificates for at least one conditional invariance specification in every lasso guaranteed that the system satisfies the original HyperLTL specification. Once again, we presented a suitable approach based on a sum-of-squares algorithm to compute augmented barrier certificates. Moreover, all of our proposed approaches in this part of the thesis were supported by suitable case studies to demonstrate the effectiveness of our results.

6.2 Discussion and Future Work

In this section, we discuss some limitations of our approaches and propose some potential future research directions that could alleviate these challenges.

Compositional Framework for Dynamic Interconnection Structures

The results presented in Chapter 3 were focused on the probabilistic safety synthesis of interconnected stochastic control systems where interconnection structures are known and fixed a priori, *i.e.*, they do not change over time. However, this is not realistic in many scenarios since the interconnection structures may not be fixed. For example, consider a road traffic network where one needs to control the number of cars entering a junction. The interconnections between different roads (*i.e.*, turns and pathways at the junction) can be temporarily blocked due to construction or accidents, resulting in different interconnection topologies over a period of time. Such a stochastic control system can be considered as an interconnection between smaller stochastic control subsystems, and the switching between subsystems may be modelled by a Markov policy. This constitutes the requirement of a new definition of control sub-barrier certificates that depend on the switching Markov policy. One can also appropriately adapt the compositionality conditions presented in Chapter 3 to compositionally construct the control barrier certificates of the interconnected system by utilizing the control sub-barrier certificates of the switching subsystems. By then utilizing these control barrier certificates, one could provide probabilistic guarantees for the safety of stochastic control systems with dynamically changing interconnection topologies. We must mention that a similar methodology has been proposed for obtaining compositional abstractions of stochastic hybrid systems with dynamic interconnection structures in [14]. However, the compositional construction of control barrier certificates for dynamic interconnection topologies remains unexplored and could be an interesting topic for future research.

Data-driven Synthesis for Unknown (Stochastic) Control Systems

Barrier certificate-based approaches considered in the thesis assume that one has access to the mathematical models of the systems to be analyzed. However, in many cases, a true model of the system is not available due to the size or complexity of the system. In such cases, one cannot utilize control barrier certificates to provide safety guarantees for the system. Instead, one needs to utilize data-driven approaches for the construction of control barrier certificates. Specifically, a finite number of data samples are collected from the system, and control barrier certificates are constructed over these data samples. Then, available information about the system (such as Lipschitz continuity) is utilized to obtain formal guarantees of the safety satisfaction of the concerned system.

There have been several results in the literature concerning data-driven verification and synthesis of stochastic (control) systems against safety specifications. Some examples include [106, 108, 91] for verification and [107] for synthesis. These approaches provide formal guarantees using finitely many data samples by solving the so-called scenario convex problem (SCP) and ensuring that the feasible solution of the SCP also satisfies the barrier certificate conditions over unseen data points via robust convex problem (RCP) under some Lipschitz continuity assumptions. These approaches however suffer from the restriction of control barrier certificates to a specific parametric form (*e.g.*, polynomial functions), and are therefore difficult to find. More recently, data-driven approaches based on neural networks have become very popular. In this case, control barrier certificates and controllers are characterized as neural networks and these neural networks are trained to satisfy the required conditions via appropriate loss functions. Some results in this direction include safety verification of non-stochastic systems [139, 94], for hybrid systems [142], and for stochastic systems [81]. The controller synthesis problem was also addressed in [66, 141, 37] for non-stochastic systems. Since training over finite data sets does not guarantee the satisfaction of barrier certificate conditions over the entire state set of the system, one needs to verify the correctness of obtained certificates a posteriori. In [11], the training framework is incorporated with a so-called validity condition derived from the scenario convex problem to achieve formal safety guarantees after successful convergence. More investigation is needed to obtain controller synthesis for the probabilistic satisfaction of safety specifications in the context of stochastic control systems.

Moreover, the compositional construction of control barrier certificates presented in Chapter 3 makes two important assumptions: (1) the dynamics of the interconnected stochastic control system, as well as the dynamics of the subsystems constituting the interconnected system, are known (2) the interconnection topology of the interconnected system is known a priori. In the absence of such information, the results presented in Chapter 3 cannot be utilized to provide probabilistic safety guarantees. Taking inspiration from the data-driven approaches presented in the aforementioned literature, one could propose a graph neural network-based architecture to construct control barrier certificates for a large-scale interconnected (stochastic) control system in a distributed manner. Then, under some Lipschitz continuity assumptions of the subsystems, as well as some approximate interconnection topology, one may be able to achieve formal safety guarantees for the interconnected systems in a compositional manner.

Controller Synthesis via k-Inductive Barrier Certificates

The results presented in Chapter 4 are mainly focused on the verification problem of (stochastic) dynamical systems against safety and reachability specifications. An interesting problem that follows directly is the controller synthesis of these systems via k-inductive barrier certificates. Unfortunately, however, this is a difficult problem. To see this, consider a discrete-time non-stochastic control system $\mathfrak{S} = (X, U, f)$ as in Definition 1. For this system, we would like to construct k-inductive barrier certificates for controller synthesis by extending Definition 19. An immediate extension of conditions (4.4) and (4.5) would be as follows:

$$\begin{aligned} &\forall x \in X, \exists u_1 \in U, \dots, \exists u_k \in U : \\ &\quad \mathbb{B}(f(x, u_1)) - \mathbb{B}(x) \leq \epsilon, \qquad (6.1) \\ &\quad \mathbb{B}(f_k(x, u_1, \dots, u_k)) - \mathbb{B}(x) \leq 0, \qquad (6.2) \end{aligned}$$

where $f_k(x, u_1, \dots, u_k)$ is obtained by evolving the dynamics of $\mathfrak{S}$ recursively for k time steps starting from $x \in X$ under the application of inputs $u_1, u_2, \dots, u_k \in U$ at each time step, respectively. However, conditions (6.1) and (6.2) do not incorporate any memory in the state as well as control inputs, which can result in problems. For instance, consider a state x_1 for which a k-inductive barrier certificate satisfying conditions (6.1) and (6.2) is found under the sequence of control inputs $u_1, \dots, u_k$. After the first time step, the system moves to the state $x_2 = f(x_1, u_1)$. Once this transition occurs, the information of the previous state x_1 and the generated sequence of control inputs $u_2, \dots, u_k$ is automatically disregarded, and a new sequence of control inputs satisfying (6.1) and (6.2) are obtained. Unfortunately, this means that from the state x_1, one can no longer ensure the decrease in the value of k-inductive barrier certificate after k-time steps, resulting in possible safety violations. Therefore, it is important to incorporate information about all the states visited between the first and the k^{th} time step, as well as the corresponding control inputs into the k-inductive barrier certificate definition so that one can ensure the satisfaction of safety specifications. Note that a similar problem also persists for the other notions of k-inductive barrier certificates introduced in Chapter 4 for both stochastic and non-stochastic systems. We leave any further investigations on this subject as future work.

Reducing Conservatism in Decomposition of ω-Regular Specifications

The synthesis approach proposed in Section 5.3 of Chapter 5 handles the controller synthesis problem against ω-regular specifications described as DSA by decomposing them into a collection of safety properties. For these tasks, we construct control barrier certificates and corresponding control policies so that one may combine the safety guarantees to obtain guarantees over the satisfaction of the original specifications. There are two main sources of conservatism in this approach. First, as suggested in Remark 36, the decomposition of specifications into safety tasks leads to ignoring the states in F and considering only the states in E of the DSA $\mathcal{A}^s$ corresponding to the specification. Unfortunately, this is tailored to the nature of control barrier certificates which provide probabilistic guarantees over the satisfaction of safety specifications. However, by reformulating suitable notions of CBCs for reachability specifications and combining them with the existing notions of CBCs for safety, one may be able to consider the states in F and

provide guarantees for visiting such states infinitely often. The second source of conservatism comes from the observations made in Remark 39. The acceptance condition of DSA $\mathcal{A}^s$ for the specification allows the states in E to be visited finitely often. However, we allow these states to be visited at most twice. By reformulating a suitable control barrier certificate definition that allows to reach a state in E an arbitrary (but finite) number of times, one can reduce some conservatism in our approach. We leave further investigations in this direction to future work.

Controller Synthesis against HyperLTL Specifications

An interesting problem that follows HyperLTL verification for dynamical systems presented in Section 5.4 of Chapter 5 is the synthesis of controllers ensuring the satisfaction of HyperLTL specifications. In this case, for a system $\mathfrak{S} = (X, U, f)$ and a HyperLTL specification ϕ, one would view U as the control input set and the term ν in condition (5.17) as control signal rather than exogenous one and design ν such that the corresponding traces T of $\mathfrak{S}$ satisfy ϕ. Unfortunately, there are major challenges in synthesizing controllers even when the HyperLTL specification is a simple conditional invariance (CI). Let us consider a CI χ, and a controller $\mathcal{G} : X \to U$ such that $\nu(t) := \mathcal{G}(x(t))$. Then, condition (5.17) of the augmented barrier certificate for CI χ in the context of synthesis can be reformulated as: for any $\tilde{x} = (x_1, x_2, \ldots, x_p) \in X^p$,

$$\mathbb{B}(f(x_1, \mathcal{G}(x_1)), f(x_2, \mathcal{G}(x_2)), \ldots, f(x_p, \mathcal{G}(x_p))) - \mathbb{B}(\tilde{x}) \leq 0. \tag{6.3}$$

The above formulation ensures that the selection of the control input at any given state $x \in X$ according to map $\mathcal{G}$ is independent of the previous traces selected by the players.

However, to satisfy this condition, one must simultaneously search for suitable functions $\mathbb{B}$ and $\mathcal{G}$. This makes the above inequality non-convex in these unknown functions and unfortunately, one cannot leverage convex programming and correspondingly SOS and semi-definite programming to determine these functions even when they are assumed to be polynomials. However, given a map $\mathcal{G}$, one could search for a function $\mathbb{B}$ such that condition (6.3) is satisfied, which is technically a verification problem and not a synthesis one anymore. In general, even though one can verify whether a HyperLTL specification ϕ is realizable over a system, it is not possible to synthesize the control map $\mathcal{G}$ that ensures the satisfaction of ϕ. In other words, it is not possible to find a solution to the HyperLTL synthesis problem.

This is due to the fact that the inputs obtained satisfying condition (5.17) in the case of verification may depend on the previously quantified traces, which is not possible when considering controller synthesis. Remark that the problem of HyperLTL verification coincides with HyperLTL synthesis when the specification is of the form $\phi = \exists\pi_1 \forall\pi_2 \ldots \forall\pi_p \psi$. To verify such specifications, it is sufficient to synthesize a controller for the first trace. Since the remaining traces are controlled by Abelard, ABCs satisfying condition (5.17) implies the satisfaction of condition (6.3) with the controller $\mathcal{G}$ being the one synthesized for the first trace. Further investigations for the synthesis problem may lead to interesting results.

Analysis of HyperLTL Specifications in Other Classes of Systems

It may also be interesting to investigate the verification and synthesis of HyperLTL specifications for other classes of systems including but not limited to continuous-time dynamical systems,

stochastic systems, hybrid systems, switched systems, and unknown systems. Moreover, it would also be worthwhile to tackle the scalability issues in computing ABCs by proposing a compositionality framework similar to that of Chapter 3 for large-scale systems.

Bibliography

[1] 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.

[2] M. Ahmadi, B. Wu, H. Lin, and U. Topcu. Privacy verification in POMDPs via barrier certificates. In *IEEE Conference on Decision and Control*, pages 5610–5615, 2018.

[3] M. Anand, P. Jagtap, and M. Zamani. Verification of switched stochastic systems via barrier certificates. In *IEEE Conference on Decision and Control CDC*, pages 4373–4378, 2019.

[4] M. Anand, A. Lavaei, and M. Zamani. Compositional construction of control barrier certificates for large-scale interconnected stochastic systems. In *Proceedings of the 21st IFAC World Conference*, volume 53, pages 1862–1867, 2020.

[5] M. Anand, A. Lavaei, and M. Zamani. From small-gain theory to compositional construction of barrier certificates for large-scale stochastic systems. *IEEE Transactions on Automatic Control*, pages 1–8, 2022. arXiv:2101.06916v2.

[6] M. Anand, A. Lavaei, and M. Zamani. Compositional synthesis of control barrier certificates for networks of stochastic systems against ω-regular specifications. *Nonlinear Analysis: Hybrid Systems*, 2023. accepted, arXiv:2103.02226.

[7] M. Anand, V. Murali, A. Trivedi, and M. Zamani. Formal verification of hyperproperties for control systems. In *Proceedings of the Workshop on Computation-Aware Algorithmic Design for Cyber-Physical Systems*. Association for Computing Machinery, 2021.

[8] M. Anand, V. Murali, A. Trivedi, and M. Zamani. Safety verification of dynamical systems via k-inductive barrier certificates. In *IEEE Conference on Decision and Control (CDC)*, pages 1314–1320, 2021.

[9] M. Anand, V. Murali, A. Trivedi, and M. Zamani. K-inductive barrier certificates for stochastic systems. In *25th ACM International Conference on Hybrid Systems: Computation and Control*, 2022.

[10] M. Anand, V. Murali, A. Trivedi, and M. Zamani. Verification of hyperproperties for uncertain dynamical systems via barrier certificates. *Transactions on Automatic Control*, 2023. conditionally accepted, arXiv: 2105.05493.

[11] M. Anand and M. Zamani. Formally verified neural network control barrier certificates for unknown systems. In *22nd IFAC World Congress*, 2023. to appear.

[12] M. Arcak, C. Meissen, and A. Packard. *Networks of dissipative systems: compositional certification of stability, performance, and safety*. Springer Briefs in Control, Automation and Robotics. 2016.

[13] E. Asarin, O. Maler, and A. Pnueli. Reachability analysis of dynamical systems having piecewise-constant derivatives. *Theoretical Computer Science*, 138(1):35–65, 1995.

[14] A. U. Awan and M. Zamani. Abstractions of networks of stochastic hybrid systems under randomly switched topologies: A compositional approach. *Systems & Control Letters*, 175:105512, 2023.

[15] C. Baier and J.-P. Katoen. *Principles of model checking*. MIT press, 2008.

[16] S. Bak. t-Barrier certificates: a continuous analogy to k-induction. In *6th IFAC Conference on Analysis and Design of Hybrid Systems*, number 16, pages 145–150, 2018.

[17] C. Belta, B. Yordanov, and E. Göl. *Formal methods for discrete-time dynamical systems*. Studies in Systems, Decision and Control. Springer, 2017.

[18] F. Benhamou and L. Granvilliers. Continuous and interval constraints. In *Handbook of Constraint Programming*, volume 2 of *Foundations of Artificial Intelligence*, pages 571–603. Elsevier, 2006.

[19] A. Bisoffi and D. V. Dimarogonas. A hybrid barrier certificate approach to satisfy linear temporal logic specifications. In *American Control Conference*, pages 634–639, 2018.

[20] J. Bochnak, M. Coste, and M. F. Roy. *Real algebraic geometry*. Springer-Verlag, 1998.

[21] S. P. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Found. Trends Mach. Learn.*, 2011.

[22] M. Brain, S. Joshi, D. Kroening, and P. Schrammel. Safety verification and refutation by k-Invariants and k-Induction. In *Static Analysis*, Lecture Notes in Computer Science, pages 145–161, 2015.

[23] J. R. Büchi. On a decision method in restricted second order arithmetic. In *The Collected Works of J. Richard Büchi*, pages 425–435. Springer, 1990.

[24] A. Chakarov. *Deductive verification of infinite-state stochastic systems using martingales*. PhD thesis, University of Colorado Boulder, 2016.

[25] A. Chakarov and S. Sankaranarayanan. Expectation invariants for probabilistic program loops as fixed points. In *International Static Analysis Symposium*, pages 85–100. Springer, 2014.

[26] D. Chandler and J. Percus. *Introduction To Modern Statistical Mechanics*. Oxford University Press, 1987.

[27] A. Cimatti, A. Griggio, and R. Schaafsma, B. J.and Sebastiani. The MathSAT5 SMT solver. In *Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science, pages 93–107, 2013.

[28] A. Clark. Control barrier functions for complete and incomplete information stochastic systems. In *American Control Conference*, pages 2928–2935. IEEE, 2019.

[29] A. Clark. Control barrier functions for stochastic systems. *Automatica*, 130:109688, 2021.

[30] M R. Clarkson, B. Finkbeiner, M. Koleini, K. K. Micinski, M. N. Rabe, and C. Sánchez. Temporal logics for hyperproperties. In *Principles of Security and Trust*, Lecture Notes in Computer Science, pages 265–284, 2014.

[31] M. R. Clarkson and F. B. Schneider. Hyperproperties. *Journal of Computer Security*, 18(6):1157–1210, 2010.

[32] N. Coenen, B. Finkbeiner, C. Sánchez, and Leander Tentrup. Verifying hyperliveness. In *Computer Aided Verification*, pages 121–139, 2019.

[33] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms, third edition*. The MIT Press, 3rd edition edition, 2009.

[34] D. Cumin and C. P. Unsworth. Generalising the Kuramoto model for the study of neuronal synchronisation in the brain. *Physica D: Nonlinear Phenomena*, 226(2):181–196, 2007.

[35] S. Dashkovskiy, B. S. Rüffer, and F. R. Wirth. An ISS small gain theorem for general networks. *Mathematics of Control, Signals, and Systems*, 19(2):93–122, 2007.

[36] S. N Dashkovskiy, B. S. Rüffer, and F. R. 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.

[37] C. Dawson, S. Gao, and C. Fan. Safe control with learned certificates: A survey of neural lyapunov, barrier, and contraction methods for robotics and control. *IEEE Transactions on Robotics*, pages 1–19, 2023.

[38] G. De Giacomo and M. Y. Vardi. Linear temporal logic and linear dynamic logic on finite traces. In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, pages 854–860, 2013.

[39] L. De Moura and N. Bjørner. Z3: An efficient SMT solver. In *Proceedings of the International conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 337–340, 2008.

[40] L. de Moura, H. Rueß, and M. Sorea. Bounded model checking and induction: from refutation to verification. In *Computer Aided Verification*, Lecture Notes in Computer Science, pages 14–26, 2003.

[41] B. Delahaye, B. Caillaud, and A. Legay. Probabilistic contracts: A compositional reasoning methodology for the design of stochastic systems. In *International Conference on Application of Concurrency to System Design*, pages 223–232, 2010.

[42] A. F. Donaldson, L. Haller, D. Kroening, and P. Rümmer. Software verification using k-induction. In *Static Analysis*, Lecture Notes in Computer Science, pages 351–368, 2011.

[43] J. L. Doob. *Stochastic Processes*. John Wiley and Sons, 1953.

[44] A. Duret-Lutz, E. Renault, M. Colange, F. Renkin, A. Gbaguidi A., P. Schlehuber-Caissier, T. Medioni, A. Martin, J. Dubois, C. Gillard, and H. Lauko. From Spot 2.0 to Spot 2.10: What's new? In *Computer Aided Verification*, Lecture Notes in Computer Science, pages 174–187, 2022.

[45] J. C. Filliâtre. Deductive software verification. *International Journal on Software Tools for Technology Transfer*, 13(5):397, 2011.

[46] B. Finkbeiner, L. Haas, and H. Torfah. Canonical representations of k-safety hyperproperties. In *IEEE 32nd Computer Security Foundations Symposium*, pages 17–1714, 2019.

[47] B. Finkbeiner and C. Hahn. Deciding hyperproperties. In *27th International Conference on Concurrency Theory*, volume 59 of *Leibniz International Proceedings in Informatics*, pages 13:1–13:14, 2016.

[48] B. Finkbeiner, C. Hahn, J. Hofmann, and L. Tentrup. Realizing ω-regular hyperproperties. In *Computer Aided Verification*, Lecture Notes in Computer Science, pages 40–63, 2020.

[49] B. Finkbeiner, C. Hahn, and H. Torfah. Model checking quantitative hyperproperties. In *Computer Aided Verification*, pages 144–163, 2018.

[50] B. Finkbeiner, M. N. Rabe, and C. Sánchez. Algorithms for model checking HyperLTL and HyperCTL*. In *Computer Aided Verification*, 2015.

[51] S. Gao, J. Avigad, and E. M. Clarke. δ-complete decision procedures for satisfiability over the reals. In *Automated Reasoning*, Lecture Notes in Computer Science, pages 286–300, 2012.

[52] S. Gao, J. Kapinski, J. Deshmukh, N. Roohi, A. Solar-Lezama, N. Arechiga, and S. Kong. Numerically-robust inductive proof rules for continuous dynamical systems. In *Computer Aided Verification*, Lecture Notes in Computer Science, pages 137–154, 2019.

[53] S. Gao, S. Kong, and E. M. Clarke. dReal: An SMT solver for nonlinear theories over the reals. In *Automated Deduction - CADE* - 24, Lecture Notes in Computer Science, pages 208–214, 2013.

[54] J. Giraldo, E. Mojica-Nava, and N. Quijano. Tracking of Kuramoto oscillators with input saturation and applications in smart grids. In *American Control Conference*, pages 2656–2661, 2014.

[55] J. A. Goguen and J. Meseguer. Security policies and security models. In *IEEE Symposium on Security and Privacy*, page 11, 1982.

[56] V. Goranko, A. Kuusisto, and R. Rönnholm. Game-theoretic semantics for alternating-time temporal logic. In *International Conference on Autonomous Agents & Multiagent Systems*, pages 671–679, 2016.

[57] J. G. Henriksen, J. Jensen, M. Jørgensen, N. Klarlund, R. Paige, T. Rauhe, and A. Sandholm. MONA: Monadic second-order logic in practice. In *Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science, pages 89–110. Springer, 1995.

[58] J. P. Hespanha. Modeling and analysis of networked control systems using stochastic hybrid systems. *Annual Reviews in Control*, 38(2), 2014.

[59] J. E. Hopcroft, R. Motwani, and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation (3rd Edition)*. Addison-Wesley Longman Publishing Co., Inc., 2006.

[60] T Hsu, C Sánchez, and B. Bonakdarpour. Bounded model checking for hyperproperties. In *Tools and Algorithms for the Construction and Analysis of Systems*, Lecture Notes in Computer Science, pages 94–112, Cham, 2021.

[61] C. Huang, X. Chen, W. Lin, Z. Yang, and X. Li. Probabilistic safety verification of stochastic hybrid systems using barrier certificates. *ACM Trans. Embed. Comput. Syst.*, 16(5s):1–19, 2017.

[62] P. Jagtap, S. Soudjani, and M. Zamani. Temporal logic verification of stochastic systems using barrier certificates. In *Proceedings of the International Symposium on Automated Technology for Verification and Analysis*, pages 177–193, 2018.

[63] P. Jagtap, S. Soudjani, and M. Zamani. Formal synthesis of stochastic systems via control barrier certificates. *IEEE Transactions on Automatic Control*, 2020.

[64] P. Jagtap, A. Swikir, and M. Zamani. Compositional construction of control barrier functions for interconnected control systems. In *Proceedings of the 23rd International Conference on Hybrid Systems: Computation and Control*, pages 1–11. Association for Computing Machinery, 2020.

[65] 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.

[66] W. Jin, Z. Wang, Z. Yang, and S. Mou. Neural certificates for safe control policies. *arXiv: 2006.08465*, 2020.

[67] A. Kivilicim, Ö. Karabacak, and R. Wisniewski. Safe reachability verification of nonlinear switched systems via a barrier density. In *IEEE Conference on Decision and Control (CDC)*, pages 2368–2372, 2019.

[68] J. Klein. ltl2dstar- LTL to deterministic Streett and Rabin automata.

[69] S. Kong, A. Solar-Lezama, and S. Gao. Delta-decision procedures for exists-forall problems over the reals. In *Computer Aided Verification*, pages 219–235, 2018.

[70] H. J. Kushner. *Stochastic Stability and Control*. Mathematics in Science and Engineering. Elsevier Science, 1967.

[71] 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.

[72] A. Lavaei. *Automated Verification and Control of Large-Scale Stochastic Cyber-Physical Systems: Compositional Techniques*. PhD thesis, Department of Electrical Engineering, Technische Universität München, Germany, 2019.

[73] A. Lavaei, S. Soudjani, A. Abate, and M. Zamani. Automated verification and synthesis of stochastic hybrid systems: A survey. *Automatica*, 146:110617, 2022.

[74] 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, 2019.

[75] L. Lindemann and D. V. Dimarogonas. Control barrier functions for signal temporal logic tasks. *IEEE Control Systems Letters*, 3(1):96–101, 2019.

[76] J. Liu and N. Ozay. Abstraction, discretization, and robustness in temporal logic control of dynamical systems. In *Proceedings of the 17th international conference on Hybrid systems: computation and control*, pages 293–302, 2014.

[77] S. Liu and M. Zamani. Verification of approximate opacity via barrier certificates. *IEEE Control Systems Letters*, 5(4):1369–1374, 2021.

[78] J. Löfberg. YALMIP : a toolbox for modeling and optimization in matlab. In *IEEE International Conference on Robotics and Automatio*, pages 284–289, 2004.

[79] J. Löfberg. Pre- and post-processing sum-of-squares programs in practice. *IEEE Transactions on Automatic Control*, 54(5):1007–1011, 2009.

[80] R. Majumdar, K. Mallik, A. Schmuck, and S. Soudjani. Symbolic qualitative control for stochastic systems via finite parity games. In *IFAC Conference on Analysis and Design of Hybrid Systems*, volume 54, pages 127–132, 2021.

[81] F. B. Mathiesen, S. C. Calvert, and L. Laurenti. Safety certification for stochastic systems via neural barrier functions. *IEEE Control Systems Letters*, 7:973–978, 2023.

[82] L. Mazaré. Using unification for opacity properties. In *Proceedings of the Workshop on Issues In The Theory of Security*, pages 165–176, 2004.

[83] J. McLean. A general theory of composition for trace sets closed under selective interleaving functions. In *Proceedings of 1994 IEEE Computer Society Symposium on Research in Security and Privacy*, pages 79–93, 1994.

[84] C. Meissen, L. Lessard, M. Arcak, and A. K. Packard. Compositional performance certification of interconnected systems using ADMM. *Automatica*, 61:55–63, 2015.

[85] 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, 2018.

[86] D. E. Muller. Infinite sequences and finite machines. In *Proceedings of the Fourth Annual Symposium on Switching Circuit Theory and Logical Design*, pages 3–16, 1963.

[87] A. Nejati, S. Soudjani, and M. Zamani. Compositional construction of control barrier certificates for large-scale stochastic switched systems. *IEEE Control Systems Letters*, 4(4):845–850, 2020.

[88] A. Nejati, S. Soudjani, and M. Zamani. Compositional construction of control barrier functions for networks of continuous-time stochastic systems. *IFAC-PapersOnLine*, 53(2):1856–1861, 2020.

[89] A. Nejati, S. Soudjani, and M. Zamani. Compositional construction of control barrier functions for continuous-time stochastic hybrid systems. *Automatica*, 145:110513, 2022.

[90] L. V. Nguyen, J. Kapinski, X. Jin, J. V. Deshmukh, and T. T. Johnson. Hyperproperties of real-valued signals. In *15th ACM-IEEE International Conference on Formal Methods and Models for System Design*, pages 104–113. ACM, 2017.

[91] N. Noroozi, A. Salamati, and M. Zamani. Data-driven safety verification of discrete-time networks: A compositional approach. *IEEE Control Systems Letters*, 6:2210–2215, 2022.

[92] A. V. Novikov and E. N. Benderskaya. Oscillatory neural networks based on the Kuramoto model for cluster analysis. *Pattern Recognition and Image Analysis*, 24(3):365–371, 2014.

[93] P. A. Parrilo. Semidefinite programming relaxations for semialgebraic problems. *Mathematical Programming*, 96(2):293–320, 2003.

[94] A. Peruffo, D. Ahmed, and A. Abate. Automated and formal synthesis of neural barrier certificates for dynamical models. In *Tools and Algorithms for the Construction and Analysis of Systems*, pages 370–388, 2021.

[95] S. Prajna. Barrier certificates for nonlinear model validation. *Automatica*, 42(1):117–126, 2006.

[96] S. Prajna and A. Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *Hybrid Systems: Computation and Control*, Lecture Notes in Computer Science, pages 477–492, 2004.

[97] 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*, 52:1415–1428, 2007.

[98] 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.

[99] S. Prajna and A. Rantzer. Convex programs for temporal verification of nonlinear dynamical systems. *SIAM Journal on Control and Optimization*, 46, 2007.

[100] S. Quinton and S. Graf. Contract-based verification of hierarchical systems of components. In *2008 Sixth IEEE International Conference on Software Engineering and Formal Methods*, pages 377–381, 2008.

[101] M. O. Rabin. Decidability of second-order theories and automata on infinite trees. *Bulletin of the American Mathematical Society*, 74(5):1025–1029, 1968.

[102] M. O. Rabin and D. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3(2):114–125, 1959.

[103] A. W. Roscoe. CSP and determinism in security modelling. In *Proceedings of the IEEE Symposium on Security and Privacy*, pages 114–127, 1995.

[104] B. S. Rüffer. Monotone inequalities, dynamical systems, and paths in the positive orthant of euclidean n-space. *Positivity*, 14(2):257–283, 2010.

[105] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education, 2 edition, 2003.

[106] A. Salamati, A. Lavaei, S. Soudjani, and M. Zamani. Data-driven safety verification of stochastic systems via barrier certificates. In *7th IFAC Conference on Analysis and Design of Hybrid Systems*, volume 54, pages 7–12, 2021.

[107] A. Salamati, A. Lavaei, S. Soudjani, and M. Zamani. Data-driven verification and synthesis of stochastic systems through barrier certificates, 2021. arXiv:2111.10330.

[108] A. Salamati and M. Zamani. Data-driven safety verification of stochastic systems via barrier certificates: A wait-and-judge approach. In *Learning for Dynamics and Control Conference*, volume 168, pages 441–452, 2022.

[109] C. Santoyo, M. Dutreix, and S. Coogan. Verification and control for finite-time safety of stochastic systems via barrier functions. In *IEEE Conference on Control Technology and Applications*, pages 712–717, 2019.

[110] C. Santoyo, M. Dutreix, and S. Coogan. A barrier function approach to finite-time stochastic system verification and control. *Automatica*, 125:109439, 2021.

[111] M. Sharf, B. Besselink, A. Molin, Q. Zhao, and K. H. Johansson. Assume/guarantee contracts for dynamical systems: Theory and computational tools. In *IFAC Conference on Analysis and Design of Hybrid Systems*, volume 54, pages 25–30, 2021.

[112] M. Sheeran, S. Singh, and G. Stalmarck. Checking safety properties using induction and a SAT-solver. In *Formal Methods in Computer–Aided Design*, Lecture Notes in Computer Science, pages 127–144, 2000.

[113] P. S. Skardal and A. Arenas. Control of coupled oscillator networks with application to microgrid technologies. *Science Advances*, 1(7), 2015.

[114] S. Soudjani. *Formal abstractions for automated verification and synthesis of stochastic systems*. PhD thesis, Delft University of Technology. 2014.

[115] 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.

[116] M. Srinivasan and S. Coogan. Control of mobile robots using barrier functions under temporal logic specifications. *IEEE Transactions on Robotics*, 37(2):363–374, 2021.

[117] J. Steinhardt and R. Tedrake. Finite-time regional verification of stochastic non-linear systems. *The International Journal of Robotics Research*, 31:901–923, 2012.

[118] R. S. Streett. Propositional dynamic logic of looping and converse is elementarily decidable. *Information and Control*, 54(1):121–141, 1982.

[119] 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.

[120] A. Swikir. *Compositional Synthesis of Symbolic Models for (In)Finite Networks of Cyber-Physical Systems*. PhD thesis, Technical University of Munich, Germany, 2020.

[121] 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.

[122] P. Tabuada. *Verification and control of hybrid systems: a symbolic approach*. Springer Science & Business Media, 2009.

[123] P. Tabuada and G. J. Pappas. Linear time logic control of discrete-time linear systems. *IEEE Transactions on Automatic Control*, 51(12):1862–1877, 2006.

[124] W. Thomas. Automata on Infinite Objects. In *Formal Models and Semantics*, Handbook of Theoretical Computer Science, pages 133–191. Elsevier, 1990.

[125] A. Tiwari, H. Rueß, H. Saïdi, and N. Shankar. A technique for invariant generation. In *Proceedings of the 7th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 113–127, 2001.

[126] D. Vodenicarevic, N. Locatelli, F. Abreu Araujo, J. Grollier, and D. Querlioz. A nanotechnology-ready computing scheme based on a weakly coupled oscillator network. *Scientific Reports*, 7(1):1–13, 2017.

[127] Y. Wang, S. Nalluri, and M. Pajic. Hyperproperties for robotics: Planning via hyperltl. In *IEEE International Conference on Robotics and Automation*, pages 8462–8468, 2020.

[128] Y. Wang, M. Zarei, B. Bonakdarpour, and M. Pajic. Statistical verification of hyperproperties for cyber–physical systems. 18(5s), 2019.

[129] R. Wisniewski and M. L. Bujorianu. Stochastic safety analysis of stochastic hybrid systems. In *IEEE Conference on Decision and Control*, pages 2390–2395, 2018.

[130] E. M. Wolff, U. Topcu, and R. M. Murray. Automaton-guided controller synthesis for nonlinear systems with temporal logic. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4332–4339, 2013.

[131] 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, 2016.

[132] S. Yaghoubi, K. Majd, G. Fainekos, T. Yamaguchi, D. Prokhorov, and B. Hoxha. Risk-bounded control using stochastic barrier functions. *IEEE Control Systems Letters*, 5:1831–1836, 2021.

[133] G. Yang, C. Belta, and R. Tron. Continuous-time signal temporal logic planning with control barrier functions. In *American Control Conference*, pages 4612–4618, 2020.

[134] B. Yordanov, J. Tumova, I. Cerna, J. Barnat, and C. Belta. Temporal logic control of discrete-time piecewise affine systems. *IEEE Transactions on Automatic Control*, 57(6):1491–1504, 2012.

[135] M. Zamani, G. Pola, M. Mazo Jr., and P. Tabuada. Symbolic models for nonlinear control systems without stability assumptions. *IEEE Transaction on Automatic Control*, 57(7):1804–1809, 2012.

[136] M. Zamani, I. Tkachev, and A. Abate. Towards scalable synthesis of stochastic control systems. *Discrete Event Dynamic Systems*, 27(2):341–369, 2017.

[137] K. Zhang, X. Yin, and M. Zamani. Opacity of nondeterministic transition systems: A (bi)simulation relation approach. *IEEE Transactions on Automatic Control*, 64(12):5116–5123, 2019.

[138] R. Y. Zhang and J. Lavaei. Efficient algorithm for large-and-sparse LMI feasibility problems. In *IEEE Conference on Decision and Control*, pages 6868–6875, 2018.

[139] H. Zhao, X. Zeng, T. Chen, and Z. Liu. Synthesizing barrier certificates using neural networks. In *International Conference on Hybrid Systems: Computation and Control*, 2020.

[140] H. Zhao, X. Zeng, T. Chen, Z. Liu, and J. Woodcock. Learning safe neural network controllers with barrier certificates, 2020. arXiv:2009.09826.

[141] H. Zhao, X. Zeng, T. Chen, Z. Liu, and J. Woodcock. Learning safe neural network controllers with barrier certificates. *Formal Aspects of Computing*, 33(3):437–455, 2021.

[142] Q. Zhao, X. Chen, Y. Zhang, M. Sha, Z. Yang, W. Lin, E. Tang, Q. Chen, and X. Li. Synthesizing relu neural networks with two hidden layers as barrier certificates for hybrid systems. In *International Conference on Hybrid Systems: Computation and Control*, 2021.

[143] K. Zhou and J. C. Doyle. *Essentials of robust control*. Prentice-Hall, 1998.