



THÈSE DE DOCTORAT DE

NANTES UNIVERSITÉ

ÉCOLE DOCTORALE N° 641

*Mathématiques et Sciences et Technologies du numérique,
de l'Information et de la Communication*

Spécialité : *Informatique*

Par

David JULIEN

Probabilistic Abstraction and Verification of Hybrid Dynamical Systems

En vue de la soutenance de Thèse à Nantes, le 9 décembre 2025

Unité de recherche : LS2N - UMR 6004

Rapporteur-euse-s avant soutenance :

Patricia BOUYER-DECITRE
Pedro R. D'ARGENIO

Directrice de Recherche - Laboratoire de Méthodes Formelles, Paris
Full Professor - FAMAF, Universidad Nacional de Córdoba, Argentina

Composition du Jury :

Attention, en cas d'absence d'un des membres du Jury le jour de la soutenance, la composition du jury doit être revue pour s'assurer qu'elle est conforme et devra être répercutée sur la couverture de thèse

Présidente : Assia MAHBOUBI
Examinatrices : Nathalie BERTRAND
Cristiana J. SILVA
Dir. de thèse : Benoît DELAHAYE
Co-dir. de thèse : Gilles ARDOUREL
Guillaume CANTIN

Directrice de Recherche - INRIA, LS2N, Nantes
Directrice de Recherche - Centre INRIA de l'Université de Rennes, Rennes
Assistant Professor with Habilitation - Iscte, Instituto Universitário de Lisboa, Portugal
Professeur des Universités - Nantes Université
Maître de Conférences - Nantes Université
Maître de Conférences - Nantes Université

Invitée :

Valentina LANZA Maître de Conférences - Université du Havre

REMERCIEMENTS

Comme beaucoup de choses dans ce monde, cette thèse n'aurait pu aboutir sans l'aide de toutes les personnes que j'ai croisées lors de mon parcours, et que je souhaite remercier ici.

Je remercie tout d'abord mes rapporteuses de thèse, Patricia BOUYER-DECITRE et Pedro R. D'ARGENIO, d'avoir pris le temps de lire ce manuscrit dans le détail, de le commenter mais surtout de le comprendre.

Je remercie également les membres de mon jury de soutenance de m'avoir fait l'honneur de leur présence. Merci à Assia MAHBOUBI d'avoir présidé ce jury, bien que ses thématiques de recherches soient assez éloignées des miennes. Merci à Cristiana J. SILVA d'avoir fait le déplacement depuis le Portugal ; je suis triste que son séjour ait dû être écourté, et j'ai hâte de reprendre nos discussions sur un projet de recherche commun. Merci à Valentina LANZA, qui m'a également suivi au sein du Comité de Suivi Individuel, d'avoir été présente lors de cette soutenance malgré la distance. Merci à Nathalie BERTRAND d'avoir accepté de faire partie de ce jury, même si elle n'a malheureusement pas pu participer à la soutenance.

Merci également à mes trois encadrants de thèse, pour la confiance qu'il m'ont accordé au cours de ces (presque) 4 ans de travail ensemble. Merci à Gilles ARDOUREL pour ses remarques précises sur la faisabilité ou non de nos idées, et de s'être toujours demandé quel sens nous mettions derrière celles-ci. Merci à Guillaume CANTIN pour le cadrage qu'il a maintenu lors de toutes nos réunions, et ses remarques sur les travaux des modélisateurs et modélisatrices en biologie et géographie. Enfin, merci à Benoît DELAHAYE pour son engagement permanent dans cette aventure ; son envie de me pousser plus loin, son ouverture d'esprit et sa disponibilité ont très largement participé à ce que cette thèse soit plus qu'une travail de recherche, et qu'elle soit aussi l'occasion pour moi de m'investir dans la vie du Laboratoire et de l'Université.

Merci aussi à tous les chercheurs et toutes les chercheuses avec qui j'ai eu le plaisir de travailler et d'échanger au cours de ces trois ans, notamment Géraldine DERROIRE et Beatriz M. FUNATSU.

Je tiens également à remercier toutes les membres du LS2N et de l'UFR Sciences et Techniques pour leur bonne humeur et la bonne ambiance qu'ils cultivent. C'est toujours un plaisir de croiser des collègues à la cafétéria le midi ou pour une pause café (même quand l'équipe de direction préempte la table à manger le jeudi !).

Je remercie à nouveau les enseignants et enseignantes de l'UFR Sciences et Techniques pour la confiance qu'ils m'ont accordée pendant ces trois années où je suis intervenu dans leurs cours. J'ai eu la chance de prendre part à des cours intéressants et dans lesquels j'ai, à chaque fois, été impliqué dans les questions pédagogiques.

Un grand merci à toutes mes amis et amies à Nantes, qui m'ont accompagné dans cette aventure : Sophie, Albane, Ben, Mathieu B., Mathieu V., Rémi, Josselin, Julie, Anna, Émile, Thibault, Louis, Briec, Ali, Yasmina, Sully, Hugo, Léane, Julien, et les autres copaines de mes années au labo. Merci également aux copaines hors du labo, Marie D., Louis, Lisa, Aimeric, Marie A., Damien, Lucie, Clément et Magneuch. La pente fut forte et la route pas tout à fait droite, mais chaque occasion de se retrouver a été une vraie bouffée d'air frais. Merci pour tous les repas, toutes les soirées jeux, toutes les sorties que nous avons faites ensemble. Merci également aux copaines de la salle Pic et Paroi, notamment Jonathan, Lucie, PM, Florent, Gaël et Flavio pour les supers séances passées à grimper ensemble dans le dévers !

Un merci assez particulier à mes copaines de Lyon, Paris, Marseille et autres villes de France : merci à Johann, Eve-Line, Simon, Jacob, Louise, Emma, Mewen, Corentin P., Charlotte, Gatien, Justine, Corentin D. et toutes les fanfarons et fanfaronnes de la [Farigoule](#). Chaque retrouvaille était un plaisir fou, bien que trop rare.

Merci à ma famille qui m'a toujours épaulé et a toujours cru en moi. Merci à mes parents qui m'ont toujours aidé et écouté dans les moments de doute. Merci à mes frères et sœurs d'avoir pris de mes nouvelles quand je ne le faisais pas, et d'avoir toujours fait en sorte que l'on se croise quand je rentrais à Lyon. Merci à mes oncles, tantes, cousins, cousines dont la bonne humeur a transformé chaque cousinade en un moment hors du temps ; ces moments de retrouvailles familiales sont précieux.

Finalement, un dernier merci du fond du cœur pour Marinna. Merci d'avoir su m'encourager, me motiver, me supporter pendant ces trois ans. Je me réveille chaque jour à tes côtés et j'ai encore du mal à mesurer la chance que j'ai de t'avoir. Je souhaite que cette complicité perdure et que notre bonheur ne fasse que grandir.

TABLE OF CONTENTS

Introduction	7
1 Background and Definitions	13
1.1 The question of reachability	13
1.2 Ordinary Differential Equations	14
1.2.1 Historical context	15
1.2.2 Definition of ODEs	16
1.2.3 Solving ODEs	20
1.3 Hybrid Dynamical Systems	22
1.3.1 Historical context	23
1.3.2 Definition of Hybrid Dynamical Systems	25
1.4 Markov processes	30
1.4.1 Markov chains	31
1.4.2 Markov decision processes	32
2 Statistical Verification of Dynamical Systems	37
2.1 Statistical verification for the parameterization of ODEs	38
2.1.1 Running example: a study on <i>Aurelia Aurita</i> population growth [52]	39
2.1.2 Monte-Carlo procedure	40
2.1.3 Overcoming ODE limitations	46
2.1.4 Complete study of the running example	54
2.2 Stability analysis of a model	57
2.2.1 Running example: a study on Duffing's oscillator [33]	58
2.2.2 Adaptation of mathematical notions	59
2.2.3 Estimating the shape of the basin of attraction	65
3 Probabilistic Verification of Dynamical Systems	71
3.1 Running example: a stability study on epidemic spread [60]	72
3.2 Abstraction procedure	77
3.2.1 Representation as a Hybrid Dynamical System	77

TABLE OF CONTENTS

3.2.2	Abstraction of the dynamics as a Markov chain	85
3.2.3	Aggregation of the Markov chains	91
3.2.4	Enhancement of the Markov decision process	101
3.3	Complete study of the running example	107
3.3.1	Setting of the study	107
3.3.2	Abstraction as a MDP	111
3.3.3	Devising controllers	112
Conclusion		119
A Proofs for Chapter 1		121
A.1	Proof of Lemma 1.3.2	121
B Proofs for Chapter 2		122
B.1	Proof of Lemma 2.1.12	122
B.2	Proof of Theorem 2.1.15	125
B.3	Proof of Lemma 2.2.5	127
B.4	Proof of Theorem 2.2.8	129
C Proofs for Chapter 3		132
C.1	Proof of Corollary 3.1.2	132
C.2	Proof of Lemma 3.2.5	132
C.3	Proof of Lemma 3.2.9	133
C.4	Proof of Corollary 3.2.10	135
C.5	Proof of Lemma 3.2.13	135
C.6	Proof of Lemma 3.2.16	137
C.7	Proof of Theorem 3.2.21	139
C.8	Proof of Corollary 3.2.22	141
Bibliography		145

INTRODUCTION

Context

When a scientist wishes to study a real phenomenon, they can either study it directly or through an abstraction, which is called a model. These models may be of very different forms: they can be concrete objects, e.g. a human-size representation of molecules using balls of various colors, a map as a depiction of geographic territory, etc.; they can also be conceptual with no physical existence other than their description, e.g. probabilities to represent the likelihood of observing “heads” after flipping a coin, or a graph to represent a decision process. These representations are leveraged to gain knowledge on the underlying phenomenon, to better understand it and, maybe, predict its future. They can also be leveraged to optimise said phenomenon: if one gains knowledge on the rules of evolution of a specific system, it becomes possible to take advantage of this knowledge to interact with this system in a way that achieves a specific goal or, at least, bring the system closer to that goal.

These considerations inevitably bring up the necessity to formalise the study of these models. To that end, the discipline of modelisation emerged during the 20th century thanks to the development of computer science, notably that of first-order logic [23], to provide the formal canvas required to properly study models, their properties as well as the proof of their (dis)satisfaction. Some of these models can be seen as “classical” mathematical objects, in the case of e.g. Ordinary Differential Equations (ODEs), since they have been used and studied since well before our time [57] and still remain uncomplete, mostly due to their intrinsic complexity; on the contrary, other models have been developed specifically for this field to represent concrete behaviors, by using pre-existing mathematical objects and concepts (e.g. ODEs in Hybrid Automata [4]), or by defining new ones (e.g. tokens in Petri nets [20]). This work is interested in a subgroup of mathematical models called dynamical models, which describe the evolution (w.r.t. time) of a concrete system by the means of a mathematical function. More precisely, we will focus on the setting of differential models, i.e. models involving differential equations. These models have a wide variety of applications: they are used to forecast weather [55], study

climate changes [74], predict stock market behaviors [40], understand the dynamics of an epidemic [25], etc.

Nowadays, substantial work is focused on improving the capabilities of these models; this is the goal of this thesis. Indeed, models are defined in a generic manner, which leads to difficulties regarding the proofs *in the general case*. In the case of differential models, two main limits are encountered: i) the impossibility to solve them in the general case, i.e. computing a closed-form expression of the solution; ii) the expressiveness of continuous behavior, i.e. the amount of behaviors induced by the continuity of a variable.

These limits may be overcome on a case-by-case basis, e.g. by rearranging the model. Here, we aim at providing more generic results that do not depend on the specific context of a particular study, but rather take into account reasonable¹ hypotheses which, while restricting the possibilities offered by the abstraction, are common among the communities of applied science (e.g. Biology and Physics).

Contributions

This thesis presents new contributions in studying models involving ODEs, specifically ODEs themselves as well as Hybrid Systems (i.e. systems whose dynamics may be described by several ODEs between which the system switches). These results can be divided into two main contributions that are described below.

We start this work by providing the context and mathematical background for the tools that will be used in the following chapters. More specifically, we formally define ODEs and the resolution method that will be put to use in chapter, and then introduce the Hybrid Systems which will be used in chapter.

In Chapter 2, we focus on the statistical study of ODEs by adapting Statistical Model Checking [78] to that setting. Taking advantage of probabilistic and statistical tools, such as the Law of Large Numbers [48] and the Central Limit Theorem [62], Statistical Model Checking [46] is a method to estimate the value of a quantity of interest by repeatedly computing it under variability of the initial state. Since its inception in 2002 by Younes and Simmons [78], it has been successfully applied to parameterize real-life models expressed by e.g. Markov chains [12] or Python programs [65]; it has even been used to parameterize ODEs [47]. Unfortunately, the guarantees obtained by SMC techniques only apply to the simulation space (i.e. the produced sequences of states) of the

¹w.r.t. the field of study

model, not the original model itself. In some cases, these simulations correspond exactly to behaviours of the model (in the case of e.g. Markov chains), or it is possible to define an equivalence relation between the simulations and the exact behaviors (e.g. through time-abstraction [14] or space partition); this leads to the *decidability* of certain properties such as reachability. However, when the model consists in ODEs as in [47], the approximations introduced during the resolution process prohibit a direct transfer of the properties from the simulations to the model. We tackle this limitation in Chapter 2, by bounding these errors and verifying auxiliary properties that take these bounds into account; the method is unfolded on a running example to show its potential.

In Chapter 3, we extend our knowledge on Hybrid Systems and the type of ODEs that can be used to define them while conserving verification. Indeed, it is known since 1998 [31] that the general theory of Hybrid Automata is undecidable: given a hybrid automaton and a property, one cannot guarantee that the satisfaction of this property can be evaluated. Since then, many scientists have worked to refine the boundaries of the systems that can be checked, either directly [31, 73] or by the means of equivalence relations on the behaviours of the system [11, 14, 43, 73], i.e. the possible (and probable) sequences of states of a given system. Here, we abstract the state of the system as a pair (R, λ) where R is a semi-closed subset of states of the system, and λ represents the dynamics it is subject to. This allows us to construct a discrete, finite set of states for an abstraction of the system, which will be used to construct a Markov process that will serve as a proxy for the original hybrid system. This Markov process can then be studied to verify specific properties, which can then be derived back to the hybrid system. It can also be leveraged to produce strategies, i.e. choices of change of dynamics, in order to fulfill a given property. Again, we show the potential of this method by unfolding a running example.

Notations

Among the different obstacles to overcome when writing mathematics, defining a notation system that is both consistent and easy on the eye is without a doubt one of the more complicated one. Indeed, besides being constrained by a finite alphabet, one must also compose with different conventions depending on the context (that is, the scientific field they are working on); more complicated even, is that mathematical conventions differ from country to country: a “greater than” order relation is usually strict in the United Kingdom, but it is usually loose in France. In the following, we give a few conventions that will be used throughout this thesis:

- the words “positive” and “negative” are used in their strict form, i.e. excluding 0;
- the order relations “lower than” and “greater than” are used in their strict form and denoted $<$ and $>$, respectively;
- the loose order relations “lower than or equal to” and “greater than or equal to” are denoted $\leq$ and $\geq$, respectively;
- the function $\log$ refers to the natural logarithm of base e .

We also define (some of) the notations that we will use:

- $\mathbb{N}$ is the set of positive integers and 0;
- $\mathbb{Z}$ is the set of positive and negative integers, and 0;
- $\mathbb{R}$ is the set of real numbers;
- if E is a set of both positive and negative numbers, $E_{\geq 0}$ (resp. $E_{\leq 0}$) is its restriction to positive (resp. negative) numbers and 0;
- if E is a set of both positive and negative numbers, $E_{> 0}$ (resp. $E_{< 0}$) is its restriction to positive (resp. negative) numbers only;
- a closed interval $I \subset \mathbb{Z}$ (resp. $I \subset \mathbb{R}$) of integers (resp. reals) is denoted $I = [a..b]$ for some $a, b \in \mathbb{Z}$ (resp. $[a, b]$ for some $a, b \in \mathbb{R}$);
- an open interval $I \subset \mathbb{Z}$ (resp. $I \subset \mathbb{R}$) of integers (resp. reals) is denoted $I = (a..b)$ for some $a, b \in \mathbb{Z}$ (resp. (a, b) for some $a, b \in \mathbb{R}$);

- if $(E, <)$ is an ordered structure and $x, y \in E$, the relation “ $x \ll y$ ” means that x is lower than y by at least an order of magnitude;
- given a set E and an integer $n \in \mathbb{N}$, the set E^n is the set of n -tuple of elements of E , i.e. $E^n = \{(e_1, e_2, \dots, e_n) \mid \forall i \in [1..n], e_i \in E\}$;
- given a set E , the set E^+ is the set of all finite, non-empty tuples of elements of E , i.e. $E^+ = \bigcup_{n \in \mathbb{N}_{>0}} E^n$;
- given a set E , the set E^0 is the set $\{\varepsilon\}$ containing only the empty tuple of elements of E ;
- given a set E , the set E^* is the set of all finite tuples of elements of E , i.e. $E^* = \bigcup_{n \in \mathbb{N}} E^n = \{\varepsilon\} \cup E^+$;
- if $n \in \mathbb{N}$, $\mathbb{R}^n$ is the set of vector of size n whose coordinates are real numbers;
- if $n \in \mathbb{N}$, the dimension of $E \subseteq \mathbb{R}^n$ is denoted $\dim(E) = n$;
- if $n \in \mathbb{N}$, the projection of $v \in \mathbb{R}^n$ on its i th component, $1 \leq i \leq n$, is denoted v_i ;
- if E is a set, its cardinal is denoted $|E|$;
- for any $x \in \mathbb{R}$, the absolute value of x is denoted $|x| = \sqrt{x^2}$;
- unless specified, the notation $\|\cdot\|_E$ denotes the Euclidean norm on $E \subseteq \mathbb{R}^n$, i.e. $\forall x \in E, \|x\|_E = (\sum_{i=1}^n |x_i|^2)^{\frac{1}{2}}$;
- if $n \in \mathbb{N}_{>0}$ and $x \in \mathbb{R}^n$ is a variable, the formula $x < \infty$ means that its value is finite;
- if X is a random variable, its expected value is denoted $\mathbb{E}[X]$.
- the operator “ $\wedge$ ” is the operator for logical conjunction, i.e. given two formulae ϕ and ψ , the expression $\phi \wedge \psi$ is true if and only if both expressions ϕ and ψ are true;
- the operator “ $\vee$ ” is the operator for logical disjunction, i.e. given two formulae ϕ and ψ , the expression $\phi \vee \psi$ is true if and only if either expression ϕ or ψ are true.

BACKGROUND AND DEFINITIONS

1.1 The question of reachability

In the vast, vast realm of theoretical computer science, the question of decidability is critical: intuitively, it is about determining whether a problem can be solved *in general*, i.e. for any input that satisfies the hypotheses. If a problem can be solved, it is frequent that scientists also try to estimate the complexity of the general solution, that is the difficulty to solve that problem in terms of computational time (the amount of operations to perform) and memory space (the amount of information to store at any given point). While the question of complexity can be fascinating, this is not the topic of this thesis; instead, we will focus here on determining if an object satisfies some properties of interest. This question revolves around computational problems, i.e. problems for which the answer can be algorithmically evaluated. If it a yes-no question, we talk about *decision* problem.

Example 1.1.1 (4-color problem). Given a World map, is it possible to color each country such that every two adjacent countries have different colors?¹ ◀

Then, the question of decidability is the following:

Definition 1.1.2 (Decidability). A decision problem is decidable if there exists a finite-time, deterministic method to find the correct answer. ◀

A decision problem that is not decidable is undecidable. A rather famous example is the halting problem:

Example 1.1.3 (Halting problem). Given the description of an arbitrary program and a finite input, does the program terminate?² ◀

¹Yes, as proved by Gonthier [26].

²Alonzo Church and Alan Turing proved [18] in 1936 that a general solution does not exist.

In this work, we will be interested in determining whether a model $\mathcal{M}$, that is a mathematical representation of a concrete system, satisfies a property φ . These models and these properties can be of many forms; we will focus on differential models, i.e. models that involve differential equations (which we define in Section 1.2), and the question of reachability:

Definition 1.1.4 (Reachability). Given a model $\mathcal{M}$ with a transition function between its states, and a target set $\mathcal{S}$ of states, decide whether $\mathcal{M}$ can enter one of the states in $\mathcal{S}$. ◁

We will see in Section 1.3.1 that many mathematical objects are too complex to decide whether reachability holds for them. Indeed, decidability for transition systems, i.e. systems with discrete states and rules to move from one state to another, generally boils down to state enumeration; if that set cannot be reduced to a finite one, this enumeration cannot be performed in finite time and reachability cannot be assessed without further assumptions (see [73], Theorem 6.19 p.69 and the remark thereafter). In the remaining of this chapter, we will define the models that we will study as well as standard notations for the objects we will manipulate (e.g. parameters, behaviours, etc.). In Section 1.2, we present the differential equations with which we will work in the whole thesis. In Section 1.3 and Section 1.4, we respectively present the hybrid systems and the Markov processes with which we will work in Chapter 3.

1.2 Ordinary Differential Equations

A differential equation is a relation between a (possibly multi-variate) function and its derivatives. This thesis will focus on the case of *ordinary* differential equations (ODEs for short), i.e. equations that relate functions of a single one-dimensional variable to their derivatives with respect to that variable; the case of *partial* differential equations, that is the generalization of the ODEs to the setting of multi-variate functions to their partial derivatives, is left for another work.

In this work, the first layer of abstraction of a concrete, real system is the representation of its dynamics, that is the “rules of evolution” it is subject to, as an ODE. ODEs have been commonly used for several centuries, and are particularly adapted to the description of the continuous evolution of physical quantities (e.g. position, temperature,

etc.). In this section, we provide a brief historical context of their usage as well as a description of the method we use for solving them.

1.2.1 Historical context

Differential calculus is a vast field of research, and has been for a long time. Indeed, while its formalization is usually credited to Newton and Leibniz at the end of the 17th century, some of its core concepts can be traced back to the Antiquity in Egypt [57] and Greece [58]. Since then, many mathematicians have worked on the topic, and scientists applied its results to concrete problems of their respective fields, such as Physics [33], Biology [68] or Economics [15], demonstrating how powerful differential equations are to describe our world. They are so powerful indeed that they permitted the discovery of Neptune, the eighth and furthest known planet from the Sun. In fact, Neptune is not visible to the eye: contrary to other planets, it has not been discovered by empirical observation. Rather, Le Verrier detected irregularities in the orbit of Uranus, which led him to look for an unknown massive body supposedly perturbing that orbit with its gravitational field. The position of this body was computed (by hand) by incorporating the observed data on Uranus into Newton's law of gravitation, and Neptune was subsequently observed with a telescope on September 23rd, 1846³ with an error margin of one degree relative to predictions.

At that time mathematical logic was only emerging as a subfield of mathematics, hence the study of differential *models* (rather than differential equations) was living its early days. Since then there has been tremendous development regarding model theory, which notably lead to the verification of models, that is the formal assessment that a model does (or does not) satisfy a given property. Depending on the structure of the model and the property of interest, the verification process may or may not be performed on the model itself. For instance, it is possible to decide if a finite, directed graph $\mathcal{G}$ is acyclic by trying to perform a topological ordering [36]: the algorithm always terminates and one can decide (see Definition 1.1.2 on page 13), depending on the output, if $\mathcal{G}$ was indeed acyclic or not.

When it comes to ODEs on the other hand, the process rapidly gets complicated. As a matter of fact, while the solution of an ODE always exists under the right assumptions

³Parallel to Le Verrier, John Couch Adams independently carried the same deed in England, only to mail its results to the Royal Greenwich Observatory two days after Le Verrier's public announcement to the French Academy.

(see [19]), it is in general not possible to determine a closed-form expression of that solution, thus complicating the verification of properties on the solution itself; instead, one may resort to compute an approximate solution by interpolating the solution to the ODE, which cannot be used for verifying a property as it is not exact. We will see in Chapter 2 that despite this obstacle, it is still possible to perform formal verification on ODEs, in particular by considering safety margins.

1.2.2 Definition of ODEs

In this Section 1.2 and in Chapter 2, the models at use are ODEs; they belong to a larger class of systems called *Dynamical Systems*. Their definition involves that of an open set:

Definition 1.2.1 (Open set). A subset O of a metric space (E, d_E) is *open* if every point $x \in O$ admits a radius $\rho \in \mathbb{R}_{>0}$, such that

$$\forall \tilde{x} \in E, d_E(\tilde{x}, x) \leq \rho \Rightarrow \tilde{x} \in O. \quad (1.1)$$

◁

Dynamical systems can then be defined in the following manner:

Definition 1.2.2 (Dynamical System (see [61], Definition 1 p.182)). Let E be an open subset of $\mathbb{R}^n$.

A dynamical system is a $\mathbf{C}^1$ -map $\Phi : \mathbb{R} \times E \rightarrow E$ which satisfies:

- $\forall x_0 \in E, \Phi(0, x_0) = x_0;$
- $\forall t, s \in \mathbb{R}, \forall x_0 \in E, \Phi(t + s, x_0) = \Phi(t, \Phi(s, x_0)).$

◁

Intuitively, a dynamical system gives a functional description of the solution of a physical (as in “concrete”) problem, or of the mathematical model describing that physical problem. Definition 1.2.2 is rather broad and a dynamical system may not necessarily be described with a differential equation, as explained in Example 1.2.3; in Chapter 3, we will describe another subset of dynamical systems, called *Hybrid Dynamical Systems*.

Example 1.2.3 (Free fall). Consider a 3-dimensional Euclidean vector space $\mathbb{R}^3$ with orthonormal basis $(\vec{x}, \vec{y}, \vec{z})$. Consider a mass P falling from a height $z_0 \in \mathbb{R}_{\geq 0}$, released without initial acceleration nor initial speed; suppose it is only subject to the gravity of Earth, which we write $-g\vec{z}$.

Newton's Second Law of Motion establishes that the acceleration vector $\vec{a}$ of P satisfies

$$\vec{a}(t) = -g\vec{z}. \quad (1.2)$$

It follows that the speed vector $\vec{v}$ of P satisfies

$$\begin{cases} \vec{v}(t) = \int_0^t \vec{a} dt = -gt\vec{z} + v_0, \\ v_0 \in \mathbb{R}. \end{cases} \quad (1.3)$$

We can now rewrite Eq. (1.3) as $\vec{v}(t, v_0) = -gt\vec{z} + v_0$, which fits Definition 1.2.2 as it is linear w.r.t. both t and v_0 . In that case, neither $\vec{a}$ nor $\vec{v}$ are differential equations (although $\vec{a}$ could be defined as the derivative $\frac{d\vec{v}}{dt}$ of $\vec{v}$ w.r.t. t , and $\vec{v}$ as the derivative $\frac{d\vec{z}}{dt}$ of $\vec{z}$ w.r.t. t). ◀

We now turn to the definition of ODEs. Let $n \in \mathbb{N}_{>0}$ be a positive integer. Let $E \subseteq \mathbb{R}^n$ be a subset of real vectors, then let $f : \mathbb{R} \times E \rightarrow \mathbb{R}^n$ be a function. We say that the function f defines an ODE if the following equation is well-formed, i.e. that its expression is syntactically correct (e.g. no division by 0):

$$\frac{dz}{dt} = f(t, z), \quad (1.4)$$

where $\frac{d}{dt}$ is the differential operator which produces the derivative of z w.r.t. variable t . In this work, we will focus on the case of *autonomous* ODEs, i.e. such that f does not explicitly depend on t ; this is especially important in Chapter 3, where we want the evolution of the system to not depend on the starting point in time. Therefore, we rewrite Eq. (1.4) as follows:

$$\frac{dz}{dt} = f(z). \quad (1.5)$$

Depending on the context, we may want f to explicitly depend on a second variable, called structural parameter λ , so that we can study its influence on the behaviour of a system described by an equation analogous to Eq. (1.5). For the sake of generality, we will place ourselves in this context for the remaining of the section and work with

parameterized ODEs; we consider a positive integer $m \in \mathbb{N}_{>0}$, and rewrite Eq. (1.5) so that the structural parameter $\lambda \in \mathbb{R}^m$ explicitly appears:

$$\frac{dz}{dt} = f(z, \lambda). \quad (1.6)$$

Then, we say that $\mathbf{z}(t)$ (resp. $\mathbf{z}(t, \lambda)$) is a solution to the ODE defined by f , iff $\mathbf{z}(t)$ (resp. $\mathbf{z}(t, \lambda)$) is $\mathbf{C}^1$ w.r.t. t and satisfies Eq. (1.5) (resp. Eq. (1.6)).

The addition of an initial value $z(t_0) = z_0$ defines an initial value problem to which we will refer to as a Cauchy problem later on:

$$\begin{cases} \frac{dz}{dt} = f(z, \lambda), \\ z(t_0) = z_0. \end{cases} \quad (1.7)$$

In that case, the solution to the Cauchy problem defined by f and z_0 is called *trajectory*, and denoted $\mathbf{z}(t, z_0, \lambda)$. The projection of $\mathbf{z}(t, z_0, \lambda)$ on its i th component is denoted $\mathbf{z}_i(t, z_0, \lambda)$, $1 \leq i \leq n$. We will see that these trajectories are unique w.r.t. z_0 (see Theorem 1.2.6 on page 19) and completely determined by λ (see Eq. (1.19) on page 26). In Chapter 2 as well as in this section, the models with which we will work are defined by a Cauchy problem.

In the following, we assume the existence of a set $\Lambda \subseteq \mathbb{R}^m$, such that $f(z, \lambda)$ is well-formed for any $\lambda \in \Lambda$; we call it the *parameter space of f* . Given a parameter $\lambda \in \Lambda$ and a set $\mathcal{I}$ of initial conditions, the set of all possible positions of the system, i.e. the points crossed by the trajectories $\mathbf{z}(\cdot, \cdot, \lambda) : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ for the ODE defined by f and parameter λ , denoted Ω_λ , is called *phase portrait of z induced by λ and $\mathcal{I}$* (or phase portrait of z if the context is sufficiently clear) and defined in the following manner:

$$\Omega_\lambda = \{z^* \in \mathbb{R}^n \mid \exists t^* \in \mathbb{R}_{>0} \wedge \exists z_0 \in \mathcal{I}, z^* = \mathbf{z}(t^*, z_0, \lambda)\}. \quad (1.8)$$

Several approaches for solving differential equations establish conditions on f and E for the existence and unicity of the solution to the ODE induced by f . Notably, a topological approach [61] is built upon compactity and manifolds. First, we recall a few definitions.

Definition 1.2.4 (Compactity). A metric space (E, d_E) is *compact* if every open cover

of E admits a finite subcover, i.e. if for every collection C of open subsets of E such that

$$\bigcup_{S \in C} S = E, \quad (1.9)$$

there exists a finite sub-collection C' of C such that

$$\bigcup_{S \in C'} S = E. \quad (1.10)$$

◁

Definition 1.2.5 (Manifold (see [45], Chapter 2)). Let $k \in \mathbb{N}_{>0}$ be a positive integer. A k -dimensional manifold E is a topological structure such that every point $x \in E$ has a neighborhood homeomorphic to (an open ball of) $\mathbb{R}^k$.

◁

Intuitively, a k -dimensional manifold E is a structure which “looks like” $\mathbb{R}^k$ locally, i.e. such that for every such neighborhood there exists a continuous and invertible mapping between it and (an open ball of) $\mathbb{R}^k$.

Using Definitions 1.2.4 and 1.2.5, Theorem 1.2.6 states that the Cauchy problem defined by Eq. (1.6) and an initial value $z_0 \in E$ admits a unique solution, provided f is $\mathbf{C}^1$ on E and E is a compact manifold.

Theorem 1.2.6 (Chillingworth (see [61], Theorem 4 p.189)).

Let E be a compact manifold, and let $f : \mathbb{R} \times E \rightarrow \mathbb{R}^n$ be a $\mathbf{C}^1$ function on $\mathbb{R} \times E$. Then for $z_0 \in E$, the Cauchy problem induced by f and z_0 has a unique solution $\mathbf{z}(t)$ defined for all $t \in \mathbb{R}$.

◁

Note that Theorem 1.2.6 establishes the existence and uniqueness of a *global* (w.r.t. t) solution to the ODE defined by f . In practical applications, which constitute the final purpose of this work, it is easy to define a suitable subset E of $\mathbb{R}^n$ so that such a solution exists, as E only needs to be compact (since $\mathbb{R}^n$ is, by definition, a manifold).

Note that other approaches, such as Lipschitz continuity of f in the Cauchy-Lipschitz Theorem (see [61], Theorem 3 p.188), may be leveraged to find equivalent results of existence and unicity.

1.2.3 Solving ODEs

It is known that an ODE cannot be solved in general, i.e. it is not possible to explicitly state a closed-form expression for a suitable function $z(t)$ in Eq. (1.6) if the expression of f is not restricted to certain settings (such as being linear w.r.t z). Note that the lack of such an expression does not prevent properties of ODEs, e.g. their stability, the existence of attracting points, the periodicity of their trajectories, etc., to be studied (see [13, 61, 63]). Still, in this work we want to compare the trajectory of the system to a set of data points, in order to find adequate values (w.r.t. a property) for the structural parameter λ or initial condition z_0 . Fortunately, many methods (see [16]) exist to compute a so-called *approximate* solution $\tilde{\mathbf{z}}(\cdot)$ to the ODE defined by f , which mainly consists in the interpolation of the exact solution: the timeline is discretized as a sequence $\mathcal{T} = \{t_0 < t_1 < \dots < t_i < \dots\}$ of timepoints and one uses an integration method to compute an approximate (w.r.t. the exact solution $\mathbf{z}(t)$) value $\tilde{\mathbf{z}}(t_i)$ at each timepoint $t_i \in \mathcal{T}$.

Definition 1.2.7 (ODE approximation method). Let $\mathcal{F} : [0, T] \times \mathbb{R}^n \times \mathbb{R} \rightarrow \mathbb{R}^n$ be a continuous function. Let $\mathcal{T} = \{t_0 < t_1 < \dots < t_i < \dots\}$ be a discretization of the timeline. Let $h \in \mathbb{R}_+$ be an integration step.

An approximation method (or scheme) is given by the following system:

$$\begin{cases} \tilde{\mathbf{z}}(t_0) = z_0, \\ \tilde{\mathbf{z}}(t_{i+1}) = \tilde{\mathbf{z}}(t_i) + h\Phi(t_i, \tilde{\mathbf{z}}(t_i), h). \end{cases} \quad (1.11)$$

◁

One of the most standard methods of interpolation is the Runge-Kutta method (see [72], Section 12.5 p.325). It is an iterative method, meaning that several intermediate points are iteratively computed before computing $\tilde{\mathbf{z}}(t_{i+1})$; this improves the precision of the final solution of the interpolation scheme. It is a generic method, which knows various implementations; in this work, we will focus on the “classical fourth-order Runge-Kutta method”, or “Runge-Kutta 4” (see [72], Eq. 12.33 p.328):

$$\begin{cases} \tilde{\mathbf{z}}(t_{n+1}) &= \tilde{\mathbf{z}}(t_n) + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4), \\ k_1 &= f(t_n, \tilde{\mathbf{z}}(t_n)), \\ k_2 &= f(t_n + \frac{h}{2}, \tilde{\mathbf{z}}(t_n) + h\frac{k_1}{2}), \\ k_3 &= f(t_n + \frac{h}{2}, \tilde{\mathbf{z}}(t_n) + h\frac{k_2}{2}), \\ k_4 &= f(t_n + h, \tilde{\mathbf{z}}(t_n) + hk_3). \end{cases} \quad (1.12)$$

As explained at the beginning of this section, the method used to compute the successive points of a trajectory is an approximation method: it introduces errors during the process. While these errors cannot be exactly evaluated (otherwise they could be taken into account and the method would produce exact, though discrete, solutions), they can be estimated under some conditions. Specifically, we formalize the (global) approximation error in Definition 1.2.8.

Definition 1.2.8 ((Global) approximation error associated to λ). Let $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining a parameterized ODE for a variable $t \in \mathbb{R}$, $z \in \mathbb{R}^n$ and $\lambda \in \mathbb{R}^m$, then let z_0 be an initial condition for z . Let $\lambda \in \mathbb{R}^m$ be a parameter value for f . Let $T > 0$ be a trajectory duration, then let $h > 0$ be a time duration. Let $\mathcal{T} = \{t_i \in [0, T) \mid \exists i \in \mathbb{N}, t_i = i \times h\}$ be the discretization of $[0, T)$ with time step h . Let $\mathbf{z}(t, z_0, \lambda)$ be the exact solution to the Cauchy problem defined by f , λ and z_0 , and let $\tilde{\mathbf{z}} = \{\tilde{\mathbf{z}}(t) \mid t \in \mathcal{T}\}$ be the computed, approximate solution.

The approximation error $\varepsilon^1(t, \lambda)$ associated to λ at timepoint $t \in \mathcal{T}$ between the approximate solution $\tilde{\mathbf{z}}$ and the exact solution $\mathbf{z}$ is defined as follows:

$$\varepsilon^1(t, \lambda) = \|\mathbf{z}(t, \lambda) - \tilde{\mathbf{z}}(t, \lambda)\|_{\mathbb{R}^n}. \quad (1.13)$$

Then, the *global* approximation error associated to λ between $\mathbf{z}$ and $\tilde{\mathbf{z}}$ can be defined as:

$$\varepsilon(\mathcal{T}, \lambda) = \max_{t \in \mathcal{T}} \varepsilon^1(t, \lambda). \quad (1.14)$$

◁

If the context is sufficiently clear, we may drop the λ and write $\varepsilon^1(t)$ and $\varepsilon(\mathcal{T})$. An equivalent (global) approximation error associated to the initial value z_0 , $\varepsilon^1(z_0, t)$ (resp. $\varepsilon(\mathcal{T}, z_0)$), can be defined in the same manner if the value of interest is the initial condition

z_0 instead of the structural parameter λ ; this will be the topic of Section 2.2.

In the case of Runge-Kutta 4, an upper-bound for the approximation error introduced in one step of computations can be produced (see [49]) for all $\lambda \in \Lambda$, provided Λ is a compact subset of $\mathbb{R}^m$. More precisely, this error can be bounded in terms of the successive derivatives of function f involved in Eq. (1.6), up to order 4; in dimension 1, if there exists two real numbers M and L such that the following inequalities hold everywhere in the phase space Ω of f

$$\|f(z, t)\|_{\mathbb{R}^n} \leq M, \quad \|f_{z^i t^j}\|_{\mathbb{R}^{i+j}} \leq L^{i+j}/M^{j-1}, \quad (1.15)$$

where $f_{z^i t^j} = \frac{\partial^{i+j} f}{\partial^i z \partial^j t}$ is the $(i+j)$ th derivative of f with respect to z and t , then we have

$$\varepsilon^1(h) \leq \frac{73}{720} M L^4 h^5. \quad (1.16)$$

Similar bounds can be computed for (finite) higher-dimension problems by applying Taylor expansion to $\tilde{\mathbf{z}}(t_{n+1})$ in Eq. (1.12) (see [49]).

Note that the bound computed in Eq. (1.16) is relative to a single step in the approximation method; in our case, given a trajectory duration $T > 0$ and a time step $h > 0$ we have $n = \lfloor \frac{T}{h} \rfloor$ steps to perform (where $\lfloor \cdot \rfloor$ is the floor operator) and these errors stack up, hence the bound for the global approximation error $\varepsilon(\mathcal{T}, \lambda)$, where $\mathcal{T} = \{kh \mid k \in \mathbb{N} \wedge kh \leq T\}$ for the approximate solution of duration T with time step h is

$$\varepsilon^1(\mathcal{T}, \lambda) \leq T \frac{73}{720} M L^4 h^4. \quad (1.17)$$

1.3 Hybrid Dynamical Systems

In Section 1.2, we defined models described by a single ODE induced by a function f and a parameter λ ; a downside of this approach is that it is not possible to interact with the model once it is instantiated, and everything is purely deterministic. As such, these models are isolated from their environment; while this may be a good first approximation of the behaviour of a system, real systems are often subject to arbitrary change in their dynamics or their position. In this section, we define models that involve ODEs but are more complex, to take these external changes into account and be able to have control (to some degree) on the behaviour of a system through the elaboration of strategies.

1.3.1 Historical context

During the 20th century, the development of computer science opened the possibilities of *modélisation* of concrete systems, that is their representation as a mathematical object. While the representation of systems as ODEs can be traced back to Newton or Leibniz in the 17th century, the integration of these models would have been done by hand; computers allow the automation of this process and greatly speed up the calculations, as well as allowing longer simulations times.

Parallel to that, the need to study models that are more complex than ODEs arose. Indeed, while some systems are subject to a single dynamics (e.g. Foucault's pendulum [59]), real systems are more often than not subject to discrete changes in their dynamics (e.g. the hands of a watch), because they are not isolated from their environment (someone may touch the pendulum's ball) or simply because their evolution is not continuous (a clock's second hand will generally tick rather than rotate smoothly). As such, different theories, including control-Lyapunov Functions [24] in the case of general control problem, and Ladder Logic [39] in the case of electronic systems, were developed to model and study these complex systems.

In 1993, Alur et al. introduced the theory of *hybrid automata* [4, 73], notably to study real-life processes that are controlled by programs, such as elevators or aircrafts. This framework aims at formalizing a class of transition systems that can both *flow* (a continuous evolution described by a differential equation) and *jump* (a discrete change in the differential equation or the position of the system). Such systems consist in graphs whose locations (called *modes*) describe the flow of the system, and transitions describe the jumps, as depicted in Fig. 1.1.

Hybrid automata are known to be expressive, and decidability has been a topic of interest since their conception. Notably, it has been proved [31] that reachability is decidable for initialized rectangular automata, but becomes undecidable as soon as variable reinitialization is removed, i.e. if discrete transitions do not reset every variable whose flow changes: this excludes automata that solely changes dynamics from generic results.

Bisimulation

One of the main difficulties encountered when working with hybrid automata is the uncountably infinite state space of a nontrivial system; indeed, (one of) the first requirements for a reachability problem to be decidable is the finiteness of the state space of the studied

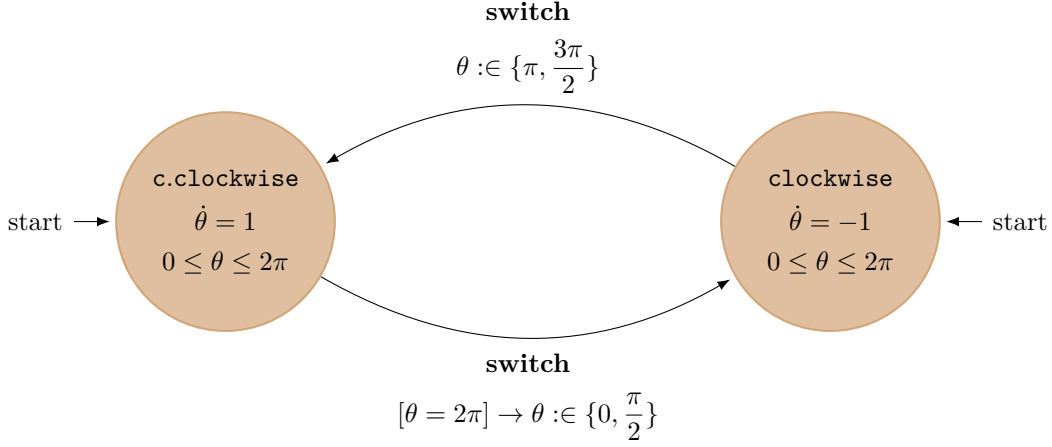


Figure 1.1: Hybrid automaton modelling a watch hand. Each mode encompasses a label, an ODE describing the evolution of the system in this mode, and the phase space of the quantity of interest (here, the angle θ of a watch hand). The starting mode is non-deterministically chosen among the modes pointed by a “start” label. Transitions allow to switch from one mode to another, and non-deterministically reset the value of θ ; they may have a *guard*, that is a condition (here, written in square brackets) that must be satisfied for the transition to be taken.

system, since decidability for transition systems often boils down to state enumeration. Still, while an obvious property of a continuous system is the infiniteness of its state space, thus forbidding the study of the system itself, it is possible to overcome this limit by the mean of a bisimulation relation [9, 73]: this type of relation ensures a specific type of equivalence between the behaviours of two systems, so that any behaviour expressed by one system may be mimicked by the other. Then, it is possible to transform a system A of interest, with an infinite state space, into an equivalent (w.r.t. behaviour) system B whose state space is finite.

Of course, the difficulty of this approach is twofold: i) finding a proper, finite system, and ii) proving the behavioural equivalence between the two systems. Despite this, different approaches [14, 42, 71] take advantage of this method and extend decidability to a restricted class of linear hybrid automata, by building an equivalent, finite state system for which reachability is decidable; specifically, they build an equivalence relation between the possible trajectories of the system, and study the (finite) quotient system instead of the original one. Unfortunately, these works restrict themselves to the setting of ODEs defined on a decidable theory, that is an ordered structure $\{E, <, +, -, \cdot, \dots\}$ where every formula is decidable. This excludes, for instance, the theory of real numbers with function

$\exp(\cdot)$ and/or $\sin(\cdot)$ [66] and *de facto* a large number of models found in e.g. Biology [8, 52] or Chemistry [34], which are often nonlinear.

Markov Processes

Lately, scientists have put effort into studying dynamical systems through Markov processes [10, 28, 44]. In this formalism, the dynamics of the original dynamical system are abstracted as stochastic processes in which the transitions are probabilistic; in doing so, a large part of the limitations due to ODEs (such as the expressiveness of the formalism or the necessity to work with a decidable theory) are circumvented, at the cost of obtaining probabilistic results instead of deterministic ones. In [28], Gyori et al. produce a Markov chain to emulate the behaviour of the dynamical system. While their abstraction is valid from a mathematical point of view, a major limitation of this work is that they encapsulate all the history of a given behaviour into each state of the resulting process; hence, it defeats the purpose of simplification of the dynamics of the original system into a stochastic graph, as i) the resulting Markov chain is infinite and ii) the produced chain is actually a tree with no possible access to a given state if the wrong path has been followed. On the other hand, Lavaei et al. [44] focus on the description of the construction of a Markov process abstracting a *Stochastic Control System*, which considers the stochasticity directly inside the ODE (intuitively, describing a model subject to internal variation) without consideration for external events which we aim at describing here. Finally, Banse et al. [11] propose a data-driven abstraction of a dynamical system through a Markov process, and provides an efficient procedure to dynamically partition the state space; this work is restricted to that of non-switched systems, i.e. systems whose dynamics are not changed during the life of the system. Here, we propose to extend the abstraction of a differential system to the setting of Markov *Decision* Processes, on which control can be performed.

1.3.2 Definition of Hybrid Dynamical Systems

As intuited in the beginning of the section, the systems we will study in Chapter 3 are based upon ODEs. We recall their form:

$$\begin{cases} \frac{dz}{dt}(t) = f(z(t), \lambda), \\ z(0) = z_0. \end{cases} \quad (1.18)$$

It is important to note that the problem defined in Eq. (1.18) is *autonomous*, as f does not explicitly depend on t . Solutions to an autonomous problem are invariant under horizontal translation: if $\mathbf{z}(t, z_0, \lambda)$ is the solution to Eq. (1.18), then for any $t_0^* \in \mathbb{R}$, $\mathbf{z}^*(t, z_0, \lambda) = \mathbf{z}(t - t_0^*, z_0, \lambda)$ is the solution to the problem

$$\begin{cases} \frac{dz}{dt}(t) = f(z(t), \lambda), \\ z(t_0^*) = z_0. \end{cases} \quad (1.19)$$

Intuitively, it means that $\mathbf{z}^*$ is merely a delayed copy of $\mathbf{z}$, and that their paths are indistinguishable modulo t . This ensures that the systems we will study are Markovian, as the forward trajectory of any point solely depends on the dynamics of the system, which we identify to parameter λ .

Let $n, m > 0$ be two integers, then let $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining an autonomous ODE. Let $\Lambda \subset \mathbb{R}^m$ be a set of parameters, such that $f(\cdot, \lambda)$ is well-formed for any $\lambda \in \Lambda$. The phase space Ω of our model is an arbitrary compact (see Definition 1.2.4) subset of $\mathbb{R}^n$ such that $f(z, \lambda) \in \Omega$ for any $z \in \Omega$ and any $\lambda \in \Lambda$. It must be positively invariant, i.e. such that any forward trajectory $\mathbf{z}(\cdot, z_0, \lambda)$ stays within Ω if the initial condition z_0 is in Ω . In practical cases, a suitable set Ω may generally be found.

As explained in Section 1.3.1, we want to abstract a hybrid system as an object whose state space is finite. A first step in this abstraction is the discretization of the timeline. Indeed, since a trajectory is continuous, considering any position along a given trajectory yields an uncountable set of states; by discretizing the timeline, we transform every trajectory into a countable set of positions, thus starting the reduction of the state space of the model. Without loss of generality, we consider a time step $\tau > 0$, and produce the following discretization of the timeline:

$$\mathcal{T} = \{k\tau \mid k \in \mathbb{N}\}. \quad (1.20)$$

From now on, we will consider that control over the hybrid model can be performed only at time points $t_k \in \mathcal{T}$. To further reduce the size of the state space, we reduce the amount of possible resets for both the parameter and the position of the model; to this end, we consider a finite set $\Lambda^* \subseteq \Lambda$ of parameter values as well as a finite set $\mathbb{Z}^* \subset \Omega$ of initial conditions, and only allow the control resets to target one (or several) of these values. While this is an obvious loss of possible behaviours for the system, this is not critical since we are working with computer simulations: the sets of possible values for λ and z_0

are finite in any case, and in practice one must work with finite (though arbitrarily large) sets.

Since we will abstract our model as a Markov decision process, the control we perform on the initial hybrid model must be appropriately expressed by probability distributions; in this study, we consider that each control action is purely probabilistic. Note that since a Dirac distribution concentrated on a single value is still a probability distribution, deterministic control can be emulated. Thus, we consider a set $\mathcal{D}$ of distributions, which is split into two subsets $\mathcal{D}^\Lambda$, $\mathcal{D}^\Omega$ depending on whether they act on the parameter λ (if they target Λ^*) or the position z of the model (if they target $\mathbb{Z}^*$). Note that we do not consider distributions that act on both the parameter λ and the position z of the system; indeed, a change of parameter would then require one distribution for each possible position z of the system, leading to an uncountable amount of distributions which would then force an uncountable set of behaviours. While this does restrict the possible behaviours of a model (since the set $\mathcal{D}^{\Omega \times \Lambda}$ is strictly larger than $\mathcal{D}^\Lambda \times \mathcal{D}^\Omega$ and it is not possible to account for e.g. conditional probability between Λ and Ω), this is for now an essential requirement for the enumeration of trajectories in the system.

Finally, we need to interact with the system. Hence, at each time point $t \in \mathcal{T}$, an action will be performed by the system; this action may be of three kinds:

- an $\emptyset$ forces the system to “wait” by performing a transition toward its current state;
- an action ζ_τ to let the system evolve according to its current dynamics and position until the next timepoint $t' \in \mathcal{T}$;
- an action $\zeta_{\mathbb{D}}$ for some $\mathbb{D} \in \mathcal{D}$, which changes the state of the hybrid system according to the realization of $\mathbb{D}$.

We will see in Eq. (1.23) how these actions induce an evolution of the system.

These actions form the action set $\mathcal{A} = \{\emptyset, \zeta_\tau\} \cup \{\zeta_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$ of the system.

We now have all the ingredients to characterize our systems, and we define a *hybrid dynamical system* as follows:

Definition 1.3.1 (Hybrid Dynamical System). A *hybrid dynamical system* is a tuple $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ where

- $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a function defining an ODE as in Eq. (1.18);
- $\Lambda \subseteq \mathbb{R}^m$ is the parameter space of the system;

- $\Omega \subseteq \mathbb{R}^n$ is the phase space of the system, and is compact and positively invariant;
- $\mathcal{T}$ is a discretization of the timeline, as defined in Eq. (1.20);
- $\mathcal{D} = \mathcal{D}^\Lambda \cup \mathcal{D}^\Omega$ is a finite set of discrete distributions on Λ and on Ω ;
- $\mathcal{A} = \{\emptyset, \zeta_\tau\} \cup \{\zeta_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$ is the set of actions that can be performed by the system.

◁

For the remaining of this section, we fix the system $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$.

A state of a hybrid dynamical system $\mathcal{S}$ is a pair $(z, \lambda) \in \Omega \times \Lambda$ comprising the position $z \in \Omega$ of the system and the dynamics it is subject to, which is identified to $\lambda \in \Lambda$ as the latter completely determines the solutions to a Cauchy problem in the form of Eq. (1.18). Given a state $s \in \Omega \times \Lambda$, we assume that the set $\mathcal{A}(s)$ of actions that can be performed in s is known, and not empty.

The first thing to note with Definition 1.3.1 is that the system is syntactically correct:

Lemma 1.3.2.

Let $\mathcal{S} = (f, \Omega, \Lambda, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid dynamical system, as defined in Definition 1.3.1.

For any $(z_0, \lambda) \in \Omega \times \Lambda$, the ODE induced by Eq. (1.18) is well-formed and the Cauchy problem induced by f , λ and z_0 admits a single continuous solution on $[t_i, t_j]$ for any $t_i < t_j \in \mathcal{T}$.

◁

The proof of Lemma 1.3.2 is given in Section A.1 on page 121.

To properly assess reachability properties, we need to define the trajectories of the system; first, we define a *continuous* transition of the system:

Definition 1.3.3 (Continuous transition in a hybrid dynamical system). Let $\tau \in \mathbb{R}_{>0}$ be the time step of $\mathcal{T}$. Let $\lambda \in \Lambda$ be a parameter. Let $z_0, z_\tau \in \Omega$ be two positions in the phase space of the system.

We say that there exists a continuous transition of duration τ from state (z_0, λ) to state (z_τ, λ) , if λ , τ , z_0 and z_τ satisfy $z_\tau = \mathbf{z}(\tau, z_0, \lambda)$, where $\mathbf{z}(\cdot, z_0, \lambda)$ is the solution to the Cauchy problem induced by f , λ and z_0 . In this case, this transition is denoted $(z_0, \lambda) \xrightarrow{\tau} (z_\tau, \lambda)$, and (z_τ, λ) is a τ -successor of (z_0, λ) .

◁

Intuitively, a continuous transition $(z_0, \lambda) \xrightarrow{\tau} (z_\tau, \lambda)$ means that a trajectory of duration τ and starting at z_0 stops on z_τ . Then, we define a *discrete* transition of the system:

Definition 1.3.4 (Discrete transition in a hybrid dynamical system). Let $\lambda_i, \lambda_f \in \Lambda$ be two parameter values for the system. Let $z_i, z_f \in \Omega$ be two positions for the system.

We say that there exists a discrete transition from state (z_i, λ_i) to state (z_f, λ_f) , if either:

- $z_f = z_i$ and there exists a distribution $\mathbb{D} \in \mathcal{D}^\Lambda$ such that $\mathbb{D}(\lambda_f) > 0$, or
- $\lambda_f = \lambda_i$ and there exists a distribution $\mathbb{D} \in \mathcal{D}^\Omega$ such that $\mathbb{D}(z_f) > 0$.

In this case, this transition is denoted $(z_i, \lambda_i) \xrightarrow{\mathbb{D}} (z_f, \lambda_f)$, and (z_f, λ_f) is a $\mathbb{D}$ -successor of (z_i, λ_i) $\triangleleft$

As we will see in Section 3.3, a controller may want to be able to wait. Since the framework of controllers require them to pick an action when it is their turn, we must create a special set of distributions, as well their corresponding actions, in order to emulate that wait. Formally, given a parameter value $\lambda \in \Lambda$, we construct the following set of distributions:

$$\mathcal{W} = \{\mathbf{1}_\lambda \mid \lambda \in \Lambda\}. \quad (1.21)$$

We can now define the transition function of the system as

$$\rightarrow := \{\xrightarrow{\tau}\} \cup \{\xrightarrow{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D} \cup \mathcal{W}\}. \quad (1.22)$$

When executing the system, a controller Φ is used for resolving non-determinism at each time point by choosing an action a in the set $\mathcal{A}(s)$ of actions of the current state $s \in \Omega \times \Lambda$. The action a picked by Φ forces the system to undergo an evolution through a transition:

$$\rightarrow_a = \begin{cases} \xrightarrow{\mathbf{1}_\lambda}, & \text{if } a = \emptyset, \\ \xrightarrow{\tau}, & \text{if } a = \zeta_\tau, \\ \xrightarrow{\mathbb{D}}, & \text{if } a = \zeta_{\mathbb{D}}. \end{cases} \quad (1.23)$$

If there exists an action $a \in \mathcal{A}$ that induces a transition $(z, \lambda) \rightarrow_a (z', \lambda')$ between (z, λ) and (z', λ') , we say that (z', λ') is a a -successor of (z, λ) .

Formally, a controller is defined as follows:

Definition 1.3.5 (Controller). Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. A controller Φ for $\mathcal{S}$ is a function

$$\begin{aligned} \Phi : (\Omega \times \Lambda)^+ &\rightarrow \mathcal{A} \\ s_0 s_1 \dots s_k &\mapsto a. \end{aligned} \tag{1.24}$$

such that $a \in \mathcal{A}(s_k)$. $\triangleleft$

The behaviours (which may be called traces) of the system controlled by Φ are the finite sequences of states $s \in \Omega \times \Lambda$. These behaviours form the set $\mathcal{B}_\Phi$ defined as follows:

$$\mathcal{B}_\Phi = \{s_0 s_1 \dots s_{k-1} s_k \in (\Omega \times \Lambda)^+ \mid \forall i \in [0..k], \Phi(s_0 s_1 \dots s_i) = a_i \wedge s_i \rightarrow_{a_i} s_{i+1}\}. \tag{1.25}$$

Note that since the controller Φ is deterministic, the action performed between s_i and s_{i+1} is readily available by evaluating $\Phi(s_0 s_1 \dots s_i)$.

Given an initial state $s_0 \in \Omega \times \Lambda$ and a controller Φ that resolves non-determinism, a probability distribution $\mathbb{P}_\Phi^k$ on the possible behaviours of length $k \in \mathbb{N}_{>0}$ can be defined in a straightforward manner. Indeed, since the continuous transitions are deterministic, the overall probability of a realization of the behaviour $\pi = s_0 s_1 \dots s_{k-1} s_k$ is exactly the product of the probabilities of realization of the successive discrete transitions; we abuse notations and write $\mathbb{D}(s_i)$ no matter if $\mathbb{D} \in \mathcal{D}^\Lambda$ or $\mathbb{D} \in \mathcal{D}^\Omega$ and we have:

$$\mathbb{P}_\Phi^k(\pi) = \prod_{\substack{i \in [0..k) \\ \Phi(s_0 s_1 \dots s_i) = \zeta_{\mathbb{D}}}} \mathbb{D}(s_{i+1}). \tag{1.26}$$

1.4 Markov processes

Here, we define the Markov processes which will be used to abstract the hybrid systems defined in Section 1.3. A Markov process (see [9], Chapter 10) is a transition system in which transitions are probabilistic. They have been extensively studied and used [6, 7, 35, 37] for modelling random processes.

Intuitively, it can be seen as a graph whose vertices are the states of the system, and the edges are the transitions between states, which are weighted following probability distributions; during a simulation, the next state is chosen probabilistically depending on the current state of the system. In general, two families of Markov process are considered (see

[9], Chapter 10), namely Markov chains (MCs) and Markov decision processes (MDPs). In a MC, the next state is probabilistically determined depending on the current state of the system, following a fixed transition probability distribution over the pairs of states of the system; in a MDP, a probability distribution is first nondeterministically picked within a pool of available distributions, before the probabilistic choice of the next state (as in MCs) is resolved. Thus, a MC is no more than a MDP in which the probability transition for each state is uniquely determined. Both MCs and MDPs will be used throughout Chapter 3 as abstractions of ODEs and Hybrid Systems, respectively. Note that this section as well as Chapter 3 focus on the case of *discrete time* Markov processes; the more complex framework of continuous time Markov processes, where transitions occur according to a given exponential random variable, is not considered in this work.

1.4.1 Markov chains

We adopt a traditional notion of Markov chains (MC) (see [9], Definition 10.1 p.747) without atomic propositions:

Definition 1.4.1 (Markov Chain (MC)). A Markov chain is a tuple $\mathbb{M} = (\mathbb{S}, \mathbb{P}, \mathbb{I})$ where

- S is a non-empty, countable set of states,
- $\mathbb{P} : S \times S \rightarrow [0, 1]$ is a transition probability distribution function such that for any state $s \in S$:

$$\sum_{s' \in S} \mathbb{P}(s, s') = 1, \quad (1.27)$$

- $\mathbb{I} : S \rightarrow [0, 1]$ is the initial distribution over the states of $\mathbb{M}$ such that $\sum_{s \in S} \mathbb{I}(s) = 1$.

◁

Classically, a behaviour of a Markov chain $\mathbb{M}$ is a (possibly infinite) sequence $\pi = q_0 q_1 \dots$ of states in S such that for every two consecutive states q_i, q_f in π , the probability $\mathbb{P}(q_i, q_f)$ is positive. Therefore, given a length $N \in \mathbb{N}_{>0}$ for the behaviours of $\mathbb{M}$, a probability distribution $\mathbb{P}_{\mathbb{M}}^N$ over all the possible behaviours of length N for $\mathbb{M}$ can be defined in a straightforward manner:

$$\mathbb{P}_{\mathbb{M}}^N(q_0 q_1 \dots q_{N-1}) = \mathbb{I}(q_0) \times \prod_{i \in [0..N-2]} \mathbb{P}(q_i, q_{i+1}). \quad (1.28)$$

MCs are suited to model random, isolated processes whose evolution only depend on their current state. They are particularly adapted to model repetitive random processes, such as an urn problem explained in Example 1.4.2.

Example 1.4.2 (Urn problem). An *urn problem* is a theoretical problem in which an urn contains colored balls, and the goal is to determine the probability of drawing one color or another, or some other properties such as “drawing color X then color Y”.

Let us consider an urn filled with 2 purple balls (p) and 1 orange ball (o). The urn problem for this exercise can be modelled by the MC $\mathbb{M} = (S, \mathbb{P}, \mathbb{I})$ depicted in Fig. 1.2 where:

- $S = \{(p, p, o), (p, o), (p, p), (p), (o), \emptyset\}$ is the state set of $\mathbb{M}$, i.e. the set of the remaining balls in the urn;
- the probability distribution $\mathbb{P}$ is defined in Table 1.1, and

	(p, p, o)	(p, o)	(p, p)	(p)	(o)	$\emptyset$
(p, p, o)	0	$\frac{2}{3}$	$\frac{1}{3}$	0	0	0
(p, o)	0	0	0	$\frac{1}{2}$	$\frac{1}{2}$	0
(p, p)	0	0	0	1	0	0
(p)	0	0	0	0	0	1
(o)	0	0	0	0	0	1
$\emptyset$	0	0	0	0	0	1

Table 1.1: Probability distribution $\mathbb{P}$ for $\mathbb{M}$. Rows represent the initial state of a transition in $\mathbb{M}$, columns represent the final state of a transition.

- $\mathbb{I}(s) = \begin{cases} 1, & \text{if } s = (p, p, o), \\ 0 & \text{otherwise.} \end{cases}$

◀

1.4.2 Markov decision processes

We adopt again a traditional notion of Markov decision processes (MDP) (see [9], Definition 10.81 p.852) without atomic propositions:

Definition 1.4.3 (Markov Decision Process (MDP)). A Markov decision process is a tuple $\mathbf{M} = (\mathbf{S}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ where

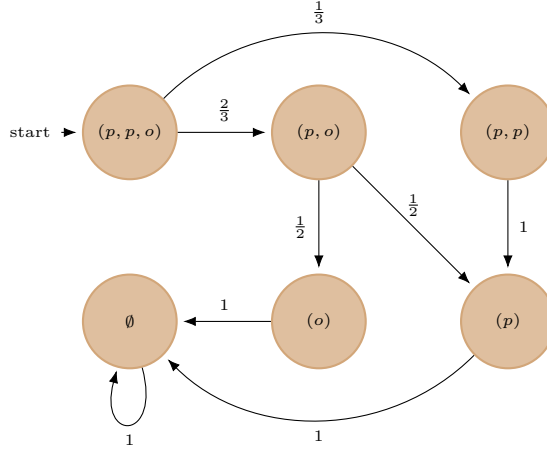


Figure 1.2: Graphical representation of $\mathbb{M}$. Nodes are states of the Markov chain. The initial distribution is a Dirac distribution concentrated on state (p, o, o) and denoted by a “start” label. Edges denote the possible transitions between states, and are weighted according to $\mathbb{P}$.

- $\mathbf{S}$ is a non-empty, countable set of states,
- $\mathbf{A}$ is a set of actions,
- $\mathbf{P} : \mathbf{S} \times \mathbf{A} \times \mathbf{S} \rightarrow [0, 1]$ is a transition probability distribution function such that for any state $s \in \mathbf{S}$ and any action $a \in \mathbf{A}$:

$$\sum_{s' \in \mathbf{S}} \mathbf{P}(s, a, s') \in \{0, 1\}, \quad (1.29)$$

- $\mathbf{I} : \mathbf{S} \rightarrow [0, 1]$ is the initial distribution over the states of $\mathbf{M}$ such that $\sum_{s \in \mathbf{S}} \mathbf{I}(s) = 1$.

An action $a \in \mathbf{A}$ is enabled in state $s \in \mathbf{S}$ iff $\sum_{s' \in \mathbf{S}} \mathbf{P}(s, a, s') = 1$. For any state $s \in \mathbf{S}$, it is required that the set $\mathbf{A}(s)$ of enabled actions is not empty. $\triangleleft$

In each state $s \in \mathbf{S}$, an action a is nondeterministically chosen before randomly selecting a state according to the distribution $\mathbf{P}(s, a, \cdot)$; each state $s' \in \mathbf{S}$ such that $\mathbf{P}(s, a, s') > 0$ is called an a -successor of s . Classically, a behaviour of a MDP $\mathbf{M}$ is an alternate (possibly infinite) sequence $\pi = s_0 a_0 s_1 a_1 s_2 \dots$ of states in $\mathbf{S}$ and actions in $\mathbf{A}$, such that for every three consecutive symbols s_i, a, s_f where s_i, s_f are states and a is an action, $\mathbf{P}(s_i, a, s_f)$ is positive. In order to reason about probabilities on the set of paths,

we need to resolve nondeterminism (of the choice of actions). In the same fashion as for hybrid systems (see Eq. (1.24)), this resolution is accomplished by a controller:

Definition 1.4.4 (Controller). Let $\mathbf{M} = (\mathbf{S}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ be a MDP. A controller Φ for $\mathbf{M}$ is a function

$$\begin{aligned} \Phi : \quad \mathbf{S}^+ &\rightarrow \mathbf{A} \\ s_0 s_1 \dots s_k &\mapsto a, \end{aligned} \tag{1.30}$$

such that $a \in \mathbf{A}(s_k)$. $\triangleleft$

As a controller resolves nondeterminism in $\mathbf{M}$, it induces a Markov chain $\mathbb{M}_\Phi = (\mathbf{S}^+, \mathbb{P}_\Phi, \mathbf{I}_\Phi)$ by unfolding $\mathbf{M}$ into a tree (or a set of trees if there are two or more states with $\mathbf{I}(s) > 0$), and adapting $\mathbf{I}$ to the setting of behaviours in $\mathbf{S}^+$; this Markov chain is such that given a behaviour $\pi = s_0 s_1 \dots s_k$ in $\mathbf{M}$ under Φ , we have

$$\mathbb{P}_\Phi(\pi, \pi s_{k+1}) = \mathbf{P}(s_k, \Phi(\pi), s_{k+1}). \tag{1.31}$$

The initial distribution $\mathbf{I}_\Phi$ is defined as follows:

$$\begin{aligned} \mathbf{I}_\Phi : \mathbf{S}^+ &\rightarrow [0, 1] \\ \pi &\mapsto \begin{cases} \mathbf{I}(s_0), & \text{if } \pi = s_0, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \tag{1.32}$$

Consequently, the behaviours in $\mathbf{M}$ induced by Φ can be identified to behaviours in $\mathbb{M}_\Phi$ and the probability that the pair $(\mathbf{M}, \Phi)$ satisfies a property φ is that of satisfying φ in $\mathbb{M}_\Phi$.

MDPs are suited to model random processes in general. Notably, since choices may be performed to influence the behaviour of the system, MDPs are indicated to mimic non-isolated processes, or processes where a choice may be performed before each modification, such as drawing from *two* urns, or a random walk as explained in Example 1.4.5.

Example 1.4.5 (Two-dimensional random walk). Let us consider a person walking on the 2-dimensional integer lattice $\mathbb{Z}^2$, i.e. a grid whose locations are the pairs (i, j) of $\mathbb{Z}^2$. We assume the walker (sometimes called the *drunkard*) is given two (fair) coins “I” and “J” which they flip in order to determine where they move: if the coin stops with the *heads* facing up, the component relative to the coin is incremented by 1, otherwise it is

decremented by 1. Before each step, they may chose if they move vertically (using coin I) or horizontally (using coin J).

This random walk can be modelled by a MDP $\mathbf{M} = (\mathbf{S}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ depicted in Fig. 1.3 where:

- $\mathbf{S} = \mathbb{Z}^2$;
- $\mathbf{A} = \{I, J\}$;
- $\mathbf{P}((i, j), a, s_f) = \begin{cases} \frac{1}{2}, & \text{if } a = I \text{ and } s_f \in \{(i-1, j), (i+1, j)\}, \\ \frac{1}{2}, & \text{if } a = J \text{ and } s_f \in \{(i, j-1), (i, j+1)\}, \\ 0 & \text{otherwise;} \end{cases}$
- $\mathbf{I} = \begin{cases} 1, & \text{if } s = (0, 0), \\ 0 & \text{otherwise.} \end{cases}$

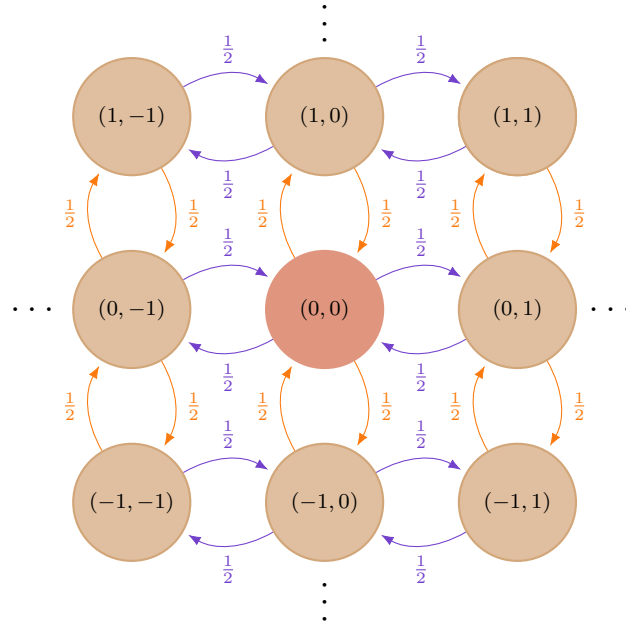


Figure 1.3: Graphical representation of $\mathbf{M}$, restricted to $[-1..1]^2$. The initial distribution is a Dirac distribution concentrated on the node $(0, 0)$, depicted in red. Orange arrows denote transitions whose probability is determined by the “I” coin, purple arrows denote transition whose probability is determined by the “J” coin.

Although $\mathbf{S}$ is infinite, it is countable, thus states can be enumerated at each step and finite-length sequences may be considered to evaluate the probability of apparition

of events. Consider the following controller:

$$\begin{aligned} \Phi : (\mathbb{Z}^2)^+ &\rightarrow \{I, J\} \\ \pi &\mapsto \begin{cases} I, & \text{if } |\pi| \text{ is odd,} \\ J, & \text{otherwise.} \end{cases} \end{aligned} \tag{1.33}$$

Given this scheduler, the probability of property “reaching $(1, 2)$ in exactly 3 steps” is:

$$\begin{aligned} \mathbf{P}(\pi) &= \mathbf{I}((0, 0)) \times \mathbb{P}_{\Phi}((0, 0), (1, 0)) \times \mathbb{P}_{\Phi}((1, 0), (1, 1)) \times \mathbb{P}_{\Phi}((1, 1), (1, 2)) \\ &= 1 \times \left(\frac{1}{2}\right)^3 \\ &= \frac{1}{8}. \end{aligned} \tag{1.34}$$

◀

STATISTICAL VERIFICATION OF DYNAMICAL SYSTEMS

As explained in Section 1.2, Ordinary Differential Equations (ODEs) constitute a very powerful framework to represent real phenomena, although a common limitation faced by scientists is the impossibility to express an analytical solution to a Cauchy problem. In this chapter, we circumvent this obstacle by adapting the framework of Statistical Model Checking [78] (SMC for short) to the setting of ODEs, essentially providing a statistical method to validate them. More precisely, we aim here at validating these models under condition of variability, i.e. while taking into account small variations in the initial parameters of the model (the parameter λ of the ODE or the initial condition z_0). Indeed, satisfaction of properties under variability is an essential aspect of e.g. biological modelling, since the processes involved vary slightly from individual to individual: two individuals of the same species, born at the same time and living together will develop differently because their respective metabolisms are different.

SMC is a general simulation-based framework used for asserting quantitative properties of a model. The main idea is to decide whether a model satisfies a given property by i) monitoring its behaviours and ii) inferring whether the samples provide statistical evidence of the (dis)satisfaction of said property. Contrary to symbolic or numerical approaches, SMC does not guarantee a correct result, but rather guarantees the likelihood that the observed result is correct. A key advantage of simulation-based techniques such as SMC is their lower memory and time requirements [79] compared to symbolic or numerical approaches ; they may also, as is our case here, be the only option if the other approaches are unavailable.

Even though several statistical techniques of evaluation can be used to achieve this goal (e.g. hypothesis testing [46, 77] via Single Sampling plan or Sequential probability ratio test), we will focus on the Monte-Carlo method [38]; this method involves repeatedly performing simulations of a model while introducing random variations of its parameters

(in our case, either the structural parameter λ or the initial condition z_0), associating a value to each simulation (in our case, a score of satisfaction of the property of interest, which we call *reward*), and then computing the empirical mean of these values. Mathematically, given a parameter λ^* (resp. initial condition z_0^*) of interest, this method produces an empirical estimation of the expected value of the reward associated to a model whose own parameter λ (resp. initial condition z_0) is sampled in a neighborhood of λ^* (resp. z_0^*). The resulting reward is then used as the satisfaction score of a given model, which we call *grade*.

Objective of this chapter

The goal of this chapter is to enable the statistical study of trajectories determined by an ODE in the form of Eq. (1.6). More precisely, given a property φ , we want to estimate the expected value of the reward obtained by a model $\mathcal{M}$ w.r.t. φ and under conditions of variability, with statistical guarantees. To this end, consider a model $\mathcal{M}$ of interest for which we will perform SMC, as well as a ball $\mathcal{B}_{\mathcal{M}}$ in which the models will be sampled for the simulations. Then, consider a radius $\alpha \in \mathbb{R}_{>0}$, a risk $\theta \in (0, 1)$, a reward function $r(\cdot)$ associating a real value to a model, and the value $\text{grade}(\mathcal{M})$ obtained through the Monte-Carlo technique. The objective of this chapter is to build the following confidence interval regarding the expected value $\mathbb{E}[r]$ of $r(\cdot)$ on $\mathcal{B}_{\mathcal{M}}$, i.e. the expected value of the reward of the model:

$$\mathbb{P}(\mathbb{E}[r] \in [\text{grade}(\mathcal{M}) - \alpha, \text{grade}(\mathcal{M}) + \alpha]) \geq 1 - \theta, \quad (2.1)$$

If a qualitative property φ is to be studied, the reward function can take the simple form of the boolean reward associated to the satisfaction of φ , i.e.

$$r_{\varphi}(\mathcal{M}) = \begin{cases} 1, & \text{if } \mathcal{M} \text{ satisfies } \varphi, \\ 0, & \text{otherwise.} \end{cases} \quad (2.2)$$

2.1 Statistical verification for the parameterization of ODEs

As stated above, this chapter revolves around statistical verification methods. We will estimate the expected value of a reward function on a given set, by repeatedly simulating

model behaviours and build a confidence interval for that value. While we only used a Monte-Carlo [38] method for statistical estimations, any other method can be used as long as a confidence interval may be constructed, though it may be necessary to assess different requirements w.r.t. the function f defining the ODE of interest or the reward function.

To ease the comprehension of the process, we will unfold it on an example throughout the Section.

2.1.1 Running example: a study on *Aurelia Aurita* population growth [52]

In 2014, Melica, Invernizzi, and Caristi published a paper studying the growth of a population of *Aurelia Aurita* individuals, a species of jellyfish very common in the Adriatic Sea (an arm of the Mediterranean Sea, between Italy, Slovenia, Croatia, Bosnia and Herzegovina, Montenegro and Albania). In that paper, the authors compare experimental data resulting from the culture of *Aurelia Aurita* polyps to ODE-based models. Assuming t is the time (expressed in days), x is the population density (expressed in cm^{-2}), a is the maximum rate of population growth (expressed in days^{-1}) and $b > 0$ is the equilibrium density (expressed in cm^{-2}), the studied model is defined as follows:

$$\begin{cases} \frac{dx}{dt}(t) = ax(t)(1 - \frac{x(t)}{b}), \\ x(0) = x_0. \end{cases} \quad (2.3)$$

In the paper, the authors prove that the dynamics of an *Aurelia Aurita* polyps population can indeed be modeled by the ODE presented above; they compute the values for a , b and x_0 that ensure the best fit w.r.t. experimental data. More precisely, they chose to minimize the residual square error $\chi = \sum_k |\mathbf{x}(t_k) - x_k|^2$, where k is the index of the day of experimental sampling, x_k is the experimental density of polyps at step k , and $\mathbf{x}(t_k)$ is the computed density at timepoint t_k corresponding to step k . These values are presented in Table 2.1, where *HD* and *LD* represent the experimental study for *High* and *Low Density*, respectively, which were both run by the researchers.

In order to validate the approach presented in this section, we will use Eq. (2.3) as our parametric ODE of interest, and focus on the *High Density* case to compare our result with that of the authors of [52].

	<i>HD</i>	<i>LD</i>
b	5.35 ± 0.11 (*** $p < 0.001$)	1.81 ± 0.08 (*** $p < 0.001$)
$x(0)$	7.59 ± 0.21 (*** $p < 0.001$)	0.081 ± 0.017 (*** $p < 0.001$)
a	0.130 ± 0.033 (** $p = 0.002$)	0.137 ± 0.012 (*** $p < 0.001$)
χ^2	0.775	0.056

Table 2.1: Estimation of parameters of the logistic curve fitting the laboratory experimental data [52].

2.1.2 Monte-Carlo procedure

The goal of the Monte-Carlo procedure is to estimate the expected value of the transformation of a random variable by repeatedly sampling it in its domain and computing the transformations of these samples. In this chapter, the transformation process is called “reward function”, i.e. a real function on the set of objects of interest (the parameters λ in Section 2.1 and the initial conditions z_0 in Section 2.2). For the sake of simplicity, we will only consider the variability of structural parameter λ to describe our approach in this section, but the same procedure will be applied to the initial condition z_0 in Section 2.2. Thus, the random variable is the parameter λ , which will be sampled in a ball around a parameter value λ^* of interest, and the transformation will be an arbitrary reward function r . For the rest of this Section 2.1, we fix the f defining an ODE in the form of Eq. (1.6), the parameter space Λ , the invariant phase space Ω associated to the model, and the initial condition z_0 ; the solution to the Cauchy problem associated to f , z_0 and a given parameter value $\lambda \in \Lambda$ is denoted $\mathbf{z}^\lambda = \mathbf{z}(\cdot, z_0, \lambda)$.

Remark 2.1.1. Any real function r (on the set of objects of interest) may be used to define a reward function, though one generally tries to encompass some meaning into it. Here, we require that these functions are integrable on the set of values of interest (Λ or Ω), so that the expected value $\mathbb{E}[r]$ of a reward function r on any compact subset of Λ exists and is finite. Note that non-integrable functions are rather uncommon and usually not at use for practical applications: it suffices (but it is not necessary) that r is piecewise-continuous on Λ for it to be integrable.

An example of integrable but not piecewise-continuous function is the indicator function $\mathbf{1}_{\mathbb{Q}}$ for the set $\mathbb{Q}$ of rational numbers: since the measure of $\mathbb{Q}$ in $\mathbb{R}$ is 0 (because $\mathbb{Q}$ is countable), it has no continuum on $\mathbb{R}$, thus $\mathbf{1}_{\mathbb{Q}}$ is not continuous on $\mathbb{R}$; yet, since $\mathbb{Q}$ is measurable, $\mathbf{1}_{\mathbb{Q}}$ is integrable and its integral equates to 0 on any interval of $\mathbb{R}$. ◀

Example 2.1.2 (Running example). Let $I \subset \mathbb{R}$ be a continuous non-empty interval, then let $\varphi =$ “the trajectory is positive or null on I ” be a property.

Consider real functions defined on I , then consider the boolean function $\text{ge}(\cdot, \cdot)$ which returns 1 if the left handside is greater than or equal to the right handside, and 0 otherwise. The function

$$\begin{aligned} g : I \rightarrow \mathbb{R}^n &\rightarrow \{0, 1\} \\ z &\mapsto \prod_{t \in I} \text{ge}(z(t), 0) \end{aligned} \quad (2.4)$$

returns 1 if $z(\cdot)$ is positive or null on I , and 0 otherwise. As such, g can be used to build a reward function $r(\lambda) = g(\mathbf{z}^\lambda)$ associated to φ which returns 1 if the exact trajectory $\mathbf{z}^\lambda$ induced by a particular value λ is positive, and 0 otherwise. Thus, it can be used as a first filter for the candidates for Eq. (2.3) in Section 2.1.1, as a biologist generally expects a population density to stay positive or null.

Note that $g(\cdot)$ can be used in this example because a symbolic, exact solution to Eq. (2.3) may be explicitly written; in the general case, such a solution cannot be written, thus the study must be carried over differently. ◀

In this chapter, two reward functions will be considered:

1. a reward function $r(\lambda)$ associated to a dataset γ , which returns 1 if the computed trajectory $\mathbf{z}^\lambda$ stays within a certain distance to the dataset, and 0 otherwise;
2. a reward function $d(\lambda)$ returning the distance between the dataset and the computed trajectory $\mathbf{z}^\lambda$.

Note that since Λ is a compact subset of $\mathbb{R}^m$ and the trajectories $\mathbf{z}^\lambda$ are continuous w.r.t. λ (see Remark B.1.3), $d(\cdot)$ is indeed continuous on Λ , thus integrable.

Once a reward function has been defined, the Monte-Carlo method can be unfolded. It is based on an intuitive idea that the empirical average of successive observations of a random variable is a good estimation of the expected value of that variable.

Example 2.1.3 (Fair coin experiment). Consider a fair coin: when flipping it, the probability of observing either “heads” or “tails” is exactly $\frac{1}{2}$. These probabilities can also be seen as the expected values of the frequency of apparition of “heads” and “tails”, respectively.

If one were to repeatedly flip a coin and write down the results as a sequence of

Definition 2.1.5 (Monte-Carlo procedure). Let $\mathring{\Lambda}$ be the interior of Λ . Let $\lambda^* \in \mathring{\Lambda}$ be a parameter candidate. Let $\rho > 0$ be a radius, such that the closed ball $\mathcal{B}_{\lambda^*} = \{\lambda \in \mathbb{R}^m \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\}$ of radius ρ around λ^* is a subset of Λ . Let N be the amount of values to sample for each candidate model. Let $r(\cdot) : \Lambda \rightarrow \mathbb{R}$ be an integrable reward function.

The Monte-Carlo procedure can be described as follows:

- sample a sequence $(\lambda_i)_{i \in [1..N]}$ of N parameter values in the ball $\mathcal{B}_{\lambda^*}$.
- compute the sequence $(r(\lambda_i))_{i \in [1..N]}$ of rewards associated to each λ_i ;
- compute the empirical mean $\hat{r}(\lambda^*) = \frac{1}{N} \sum_{i \in [1..N]} r(\lambda_i)$.

◁

Remark 2.1.6. For the sake of simplicity, we explained the Monte-Carlo procedure for the study of the structural parameter λ^* only, though the same procedure may be applied to initial condition z_0 . However, the question of simultaneous variations of both the parameter and initial condition is beyond the scope of this thesis, as we have not studied the impact of the variation of both variables on the approximation errors introduced during the computation of the approximate trajectory of the model. ◀

The Monte-Carlo procedure described in Definition 2.1.5 produces an estimation of the expected value $\mathbb{E}[r]$ of a given reward function r on the set $\mathcal{B}_{\lambda^*}$ within which input values are sampled. In our case, it means that the empirical mean $\hat{r}$ of the rewards obtained by the successive samplings of $\lambda \in \mathcal{B}_{\lambda^*}$ converges towards the expected value $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r]$ of r on $\mathcal{B}_{\lambda^*}$ as the number N of samples increases.

Of course, the precision of this estimation is highly dependent on the amount of samples: the more values are sampled, the more rewards can be computed and the better the precision becomes. It is possible, given a target precision $\alpha \in \mathbb{R}_{>0}$ (the distance between the computed value and the expected value of the random variable) and a target risk $\theta \in (0, 1)$ (the probability to have a sample set that does not represent the whole set), to compute the required amount N of samples to produce in order to meet these targets. Specifically, given the lowest (resp. highest) possible value L (resp. U) for $r(\lambda)$ on a neighborhood $\mathcal{B}_{\lambda^*}$, Hoeffding's inequality in the case of independent and identically

distributed random variables (see [32], Theorem 2 p.6) gives:

$$\begin{aligned}
 \theta &= \mathbb{P}(r \notin [\hat{r} - \alpha, \hat{r} + \alpha]), \\
 &\leq 2 \exp \left(\frac{-2\alpha^2 N^2}{\sum_{i=1}^N (U - L)^2} \right), \\
 &\Leftrightarrow \\
 N &\geq \frac{\log(2/\theta)}{2\alpha^2} (U - L)^2.
 \end{aligned} \tag{2.6}$$

Remark 2.1.7. It is important to note that the number N of samples to perform depends on three factors: the target precision α , the target risk θ (which are arbitrarily chosen in the setting of the study) and the values of L and U (which indirectly depend on the radius ρ of the sampling set $\mathcal{B}_{\lambda^*}$ and are more difficult to adjust). In the case of Bernoulli variables (i.e. a boolean reward function returning 1 if the sampled value satisfies a given property), we have $U - L = 1$ and Eq. (2.6) degenerates to

$$N \geq \frac{\log(2/\theta)}{2\alpha^2}. \tag{2.7}$$

◀

Remark 2.1.8. The following Central Limit Theorem [62] gives an estimate on the speed of asymptotic convergence of the Monte-Carlo method, i.e. its convergence when the size N of the sample set goes to infinity. While this cannot be transparently adapted to the case of a finite sample set, it gives a rough estimate of that speed when N is “big enough”, although it *never* is.

This theorem establishes the convergence of the cumulative average $\hat{X}_N$ in Theorem 2.1.4 towards the normal distribution $\mathcal{N}(\mu, \sigma^2)$:

Theorem 2.1.9 (Central Limit Theorem).

Let $X_1, X_2, \dots$ be a sequence of independent and identically distributed real random variables of expected value $\mu < \infty$ and variance $\sigma^2 < \infty$.

The distribution $\hat{X}_N = \frac{1}{N} \sum_{i=1}^N X_i$ converges almost surely to the normal distribution $\mathcal{N}(\mu, \sigma^2)$ as $N \rightarrow \infty$.

◁

Several versions of the Central Limit Theorem exist, with different assumptions on the variables (see for instance [62], Section IV.5 p.91). Here, this is simply a consequence

of Theorem 2.1.4.

Using Theorem 2.1.9 on the normalized variable $\bar{X}_N = \sum_{i=1}^N \frac{X_i - N\mu}{\sigma\sqrt{N}}$, we obtain its convergence towards the standard normal distribution $\mathcal{N}(0, 1)$ as $N \rightarrow \infty$. That is, for any $a \in \mathbb{R}$,

$$\lim_{N \rightarrow \infty} \mathbb{P}(\bar{X}_N \leq a) = \frac{1}{\sqrt{2}} \int_{-\infty}^a \exp(-x^2/2) dx. \quad (2.8)$$

Intuitively, Eq. (2.8) establishes that the difference between the empirical mean $\hat{X}_N$ and the expected value $\mathbb{E}[X]$ has a magnitude $\frac{1}{\sqrt{N}}$: performing 4 times more observations *should*² divide the error by 2 if N is “big enough”. ◀

The Monte-Carlo procedure given in Definition 2.1.5 is repeated for all candidate values λ^* that needs to be studied, yielding a grade $\hat{r}$ for each value of interest. Once this is done, the grades are used for sorting the candidates and determining the best parameter value(s) $\lambda^* \in \mathring{\Lambda}$ w.r.t. reward function r , under condition of variability. Since all candidates go through the same procedure, the precision and risk of the confidence intervals are the same for each candidate, such that the comparison is meaningful.

Example 2.1.10 (Running example). Let $\Lambda = [0, 3] \times [0, 9]$ be the set of parameters we want to study, then let $\mathring{\Lambda}$ denote its interior. Let $\Delta_\lambda = 0.01$ be the discretization step for Λ . Let $\rho = 0.005$ be a radius for the ball of variability around parameters. Finally, let

$$\Lambda^* = \{\lambda \in \mathring{\Lambda} \mid \exists i, j \in \mathbb{N}_{>0}, \lambda = (i\Delta_\lambda, j\Delta_\lambda)\} \quad (2.9)$$

be the set of parameters to be graded.

Let $T > 0$ be a time duration for the simulations, then let $\mathcal{T} = \{0 < t_1 < \dots < t_n = T\}$ be a discretization of $[0, T]$ with time step (assumed constant) t_1 . Given a parameter $\lambda \in \Lambda$, the solution to the Cauchy problem defined by Eq. (2.3), z_0 and λ is denoted $\mathbf{z}^\lambda$. Let $\varphi =$ “the solution induced by λ is positive or null on $\mathcal{T}$ ” be the property of interest. The function

$$\begin{aligned} r : \Lambda &\rightarrow \{0, 1\} \\ \lambda &\mapsto \prod_{t \in \mathcal{T}} (\text{ge}(\mathbf{z}^\lambda(t), 0)) \end{aligned} \quad (2.10)$$

defines a reward function for φ .

²As Theorem 2.1.9 is a statistical result, it must be interpreted in terms of “average” outcome even though outliers may appear among successive applications.

Let $\alpha = 0.05$ be the target precision of the estimation, let $\theta = 0.05$ be the target risk value for the estimation. Let $\hat{r}$ be the value that will be computed through the Monte-Carlo procedure. Equation (2.7) gives:

$$\begin{aligned} \theta &= \mathbb{P}(r \notin [\hat{r} - \alpha, \hat{r} + \alpha]) \leq 2 \exp(-2\alpha^2 N) \\ \Leftrightarrow \\ N &\geq \frac{\log(2/\theta)}{2\alpha^2} \\ &\geq 737.78 \end{aligned} \tag{2.11}$$

Hence, we choose to perform $N = 738$ simulations for each value of interest to grade.

Let $\lambda^* \in \Lambda^*$ be a parameter to grade, then let $\mathcal{B}_{\lambda^*} = \{\lambda \in \Lambda \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\}$ be a closed ball of radius ρ around λ^* . Let $\mathcal{C} = \{\lambda_1, \lambda_2, \dots, \lambda_N\}$ be N samples in $\mathcal{B}_{\lambda^*}$. For each value $\lambda \in \mathcal{C}$, we compute the exact solution $\mathbf{z}^\lambda$ associated to f , λ and z_0 , and consider the reward $r_\varphi(\mathbf{z}^\lambda)$. Then, the empirical mean $\hat{r}$ of all the rewards $r(\mathbf{z}^\lambda), \lambda \in \mathcal{C}$ can be computed as follows:

$$\hat{r} = \frac{1}{N} \sum_{\lambda \in \mathcal{C}} r(\mathbf{z}^\lambda). \tag{2.12}$$

This study is such that the probability that the true expected value $\mathbb{E}[r]$ lies outside of the interval $[\hat{r} - \alpha, \hat{r} + \alpha]$ (where $\hat{r}$ is the value computed through the Monte-Carlo procedure) is lower than 0.05. $\blacktriangleleft$

Remark 2.1.11. While Example 2.1.10 unfolds the Monte-Carlo procedure on an ODE model, it is **not** right in the general case and should not be used as a base to conduct an actual study. Indeed, here we assume that the reward function $r(\cdot)$ is able to determine if the trajectory induced by a given λ satisfies φ . While this is true in the running example because the solution to Eq. (2.3) can be explicitly written (see Verhulst equation [75]), this is not true in the general case because of the errors introduced in the approximation method, which are defined in Definition 1.2.8 on page 21.

Hence, one must take extra steps in order to evaluate satisfaction of properties, which we will discuss in Section 2.1.3. $\blacktriangleleft$

2.1.3 Overcoming ODE limitations

As explained in Section 1.2.3, it is generally not possible to explicitly write a closed-form expression for the solution of a given ODE. Moreover, while approximations methods

(such as Runge-Kutta 4) exist to produce a discrete, approximate solution to the ODE, they introduce errors in the process that prevent us from studying properties directly on them: since the errors are unknown (though bounded), verifying a property on the approximate solution is not equivalent to verifying it on the exact solution.

In [3], we circumvented this limitation by verifying two auxiliary properties φ_- and φ_+ , derived from the property of interest φ , that take the global approximation error into account. Intuitively, assuming an upper-bound for the global approximation error $\varepsilon > 0$ can be obtained, we can form two auxiliary properties that verify φ in the extreme scenarios where the computed solution is at a distance at most ε from the exact solution. While we focused on boolean properties in [3] (that is, “yes/no”-properties), here we will consider the more general setting of reward functions.

The first result of this thesis is Lemma 2.1.12, which ensures that the global approximation error, defined in Definition 1.2.8 on page 21, can indeed be arbitrarily upper-bounded for any parameter $\lambda \in \Lambda$, provided Λ is a subset of a compact subset of $\mathbb{R}^m$.

Lemma 2.1.12 (Uniform bound for the global approximation error w.r.t. λ).

Let $T > 0$ be a duration for the simulations. Let n be a positive integer. Let $f : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining an ODE in the form of Eq. (1.6). Let $\Lambda \subset \mathbb{R}^m$ be a compact parameter space for f , then let $\lambda^ \in \mathring{\Lambda}$ be a parameter. Let $\Omega \subseteq \mathbb{R}^n$ be the phase space of f and parameter space Λ , then let $z_0 \in \Omega$ be an initial condition. Let $\rho > 0$ be a radius, such that the ball $\mathcal{B}_{\lambda^*} = \{\lambda \in \mathbb{R}^m \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\} \subseteq \Lambda$.*

Let $(h_i)_{i \in \mathbb{N}} \in \mathbb{R}^+$ be a sequence of integration steps, such that $\lim_{i \rightarrow \infty} h_i = 0$; for each h_i , we denote $\mathcal{T}_i = \{t_k \in [0, T) \mid \exists k \in \mathbb{N}, t_k = k \times h_i\}$ the discretization it induces. For any $\varepsilon > 0$, there exists $i^ \in \mathbb{N}$ such that*

$$\forall i > i^*, \forall \lambda \in \mathcal{B}_{\lambda^*}, \quad \varepsilon(\mathcal{T}_i, \lambda) < \varepsilon. \quad (2.13)$$

◁

The proof of Lemma 2.1.12 is given in Section B.1, on page 122. We emphasize that this lemma can be supplemented by an explicit choice of a sufficiently small integration step h , provided the integration method comes with appropriate estimation of the global error w.r.t. that integration step. It has been discussed in Section 1.2.3 and Eq. (1.17).

The main result of this chapter is Theorem 2.1.15, which provides formal guarantees in the construction of a confidence interval regarding the expected value of a reward, while

taking the global approximation error into account. To that end, and since the reward function cannot directly be used for grading the values of interest, we use *auxiliary* reward function to bound the value of the grade with statistical guarantees. These auxiliary functions must meet some criteria, which are given in Definition 2.1.13.

Definition 2.1.13 (Auxiliary reward functions). Let $r : \Lambda \rightarrow \mathbb{R}$ be a Lebesgue-integrable reward function. Let $\varepsilon > 0$ be a tolerance value.

Auxiliary reward functions are functions $r_{-\varepsilon}, r_{+\varepsilon} : \Lambda \rightarrow \mathbb{R}$ which satisfy:

- $r_{-\varepsilon}, r_{+\varepsilon}$ are Lebesgue-integrable on Λ ;
- for any $\lambda \in \Lambda$, $r_{-\varepsilon}(\lambda) \leq r(\lambda) \leq r_{+\varepsilon}(\lambda)$;
- $r_{-\varepsilon}, r_{+\varepsilon}$ vary continuously w.r.t. ε , and satisfy

$$\forall \lambda \in \Lambda, \lim_{\varepsilon \rightarrow 0} r_{-\varepsilon}(\lambda) = \lim_{\varepsilon \rightarrow 0} r_{+\varepsilon}(\lambda) = r(\lambda). \quad (2.14)$$

◁

Since the tolerance ε is set once for the whole study, we ease the notation by dropping the subscript “ $_{\varepsilon}$ ” for auxiliary rewards functions, and denote them by r_{-}, r_{+} .

In this chapter, the auxiliary reward functions will be built upon each reward function of interest; intuitively, we will use the tolerance ε to increase (resp. decrease) the value of the standard error for each experimental datapoint, essentially introducing safety margins to account for the errors induced by the ODE approximation method (see Definition 1.2.7 on page 20). Formally, consider a finite discretization $\mathcal{T} = \{t_0 < t_1 < \dots < t_k\}$ of the timeline. Given an experimental dataset $\{(z_0, e_0), (z_1, e_1), \dots, (z_k, e_k)\}$ of values $z_k \in \mathbb{R}$ with standard errors $e_k \in \mathbb{R}_{\geq 0}$, we can express it as the function

$$\begin{aligned} \gamma : \mathcal{T} &\rightarrow \mathbb{R}^n \times \mathbb{R}_{\geq 0}^n \\ t_k &\mapsto (z_k, e_k). \end{aligned} \quad (2.15)$$

Then, given a tolerance value $\varepsilon > 0$ such that for any $t_k \in \mathcal{T}$, $\varepsilon \leq e_k$, the auxiliary datasets will be expressed as the functions

$$\begin{aligned} \gamma_{-\varepsilon} : \mathcal{T} &\rightarrow \mathbb{R}^n \times \mathbb{R}_{\geq 0}^n \\ t_k &\mapsto (z_k, e_k - \varepsilon), \end{aligned} \quad (2.16)$$

and

$$\begin{aligned}\gamma_{+\varepsilon} : \mathcal{T} &\rightarrow \mathbb{R}^n \times \mathbb{R}_{\geq 0}^n \\ t_k &\mapsto (z_k, e_k + \varepsilon).\end{aligned}\tag{2.17}$$

Example 2.1.14 (Running example). Let $\mathcal{T} = \{0, 1, 4, \dots, 42\}$ be the timepoints of experimental sampling, as provided in Table 1 of [52]. For each $t_k \in \mathcal{T}$, we consider the experimental mean value x_k and standard error e_k extracted from the same table. For convenience, we define the dataset as the function

$$\begin{aligned}\gamma : \mathcal{T} &\rightarrow \mathbb{R} \times \mathbb{R}_{\geq 0} \\ t_k &\mapsto (x_k, e_k).\end{aligned}\tag{2.18}$$

For convenience, we denote by $\gamma_1(\cdot)$ (resp. $\gamma_2(\cdot)$) the functions mapping a timepoint t_k to the experimental mean value (resp. standard error) at that timepoint.

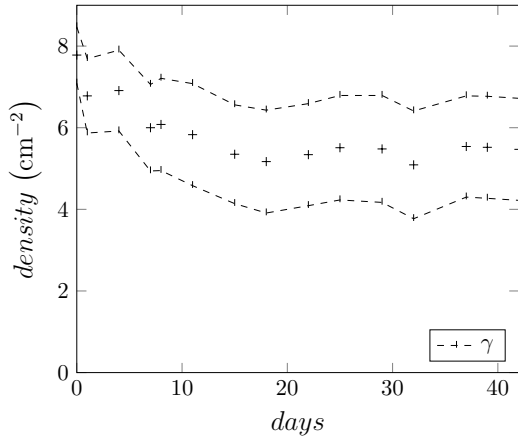
Combined, these values induce a tunnel of data, as depicted in Fig. 2.1a. Then, considering a tolerance ε , we define the auxiliary timestamped datasets $\gamma_{-\varepsilon}$ and $\gamma_{+\varepsilon}$ as follows:

$$\begin{aligned}\gamma_{-\varepsilon} : \mathcal{T} &\rightarrow \mathbb{R}^n \times \mathbb{R}_{\geq 0}^n \\ t_k &\mapsto (x_k, e_k - \varepsilon), \\ \gamma_{+\varepsilon} : \mathcal{T} &\rightarrow \mathbb{R}^n \times \mathbb{R}_{\geq 0}^n \\ t_k &\mapsto (x_k, e_k + \varepsilon),\end{aligned}\tag{2.19}$$

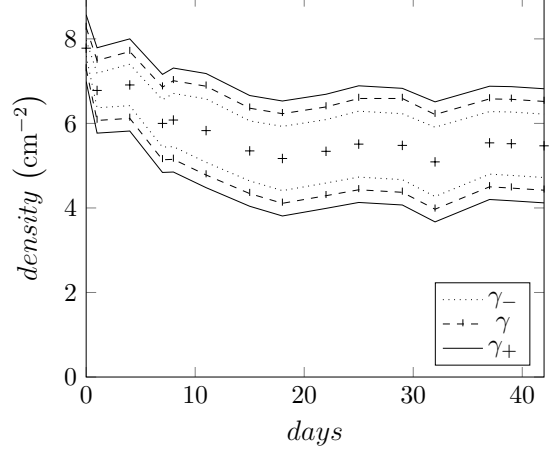
Both datasets also induce tunnels, which are depicted in Fig. 2.1b.

We are interested in answering the question “is the model faithful to the experiment?”, which we translate to “is the computed trajectory inside the tunnel induced by the experimental dataset?”. Intuitively, we want to decide if the trajectory induced by a parameter λ stays in the tunnel induced by the experimental mean and the standard error for each experimental datapoint, as represented in Fig. 2.2b. Note that in Figs. 2.1 and 2.2, the distance between the different tunnels is caricatural to be visible; in this study, we chose a value $\varepsilon = 0.01$ for the tolerance value, so that it is largely lower than the standard error in the original study, which is greater than 0.49 in the *High Density* case (see [52], Table 1). We then identified this tolerance to the upper-bound of the global approximation error $\varepsilon(\lambda, \mathcal{T})$ of the simulations.

Given a parameter value $\lambda \in \Lambda$, the exact solution to the ODE induced by Eq. (2.3), λ and $x_0 = \gamma_1(1)$ is denoted $\mathbf{x}^\lambda$. Now, let us consider the property $\varphi(\lambda) := \forall t \in \mathcal{T}, \|\gamma_1(t) -$



(a) Tunnel induced by the experimental dataset. The inner “|” denote the experimental value. The outer dashed lines serve to visualize the tunnel induced by the standard error above and below the experimental value.



(b) Tunnel induced by the experimental dataset and the auxiliary datasets. The dashed lines serve to visualize the tunnel induced by the experimental dataset, the dotted lines refer to the dataset with tolerance $-\varepsilon$, the solid lines refer to the dataset with tolerance ε .

Figure 2.1: Tunnels induced by the combination of values and standard error. Note that in these figures, the distance between the different tunnels are caricatural to be visible.

$\mathbf{x}^\lambda(t) \|\mathbb{R} \leq \gamma_2(t)$. In this scenario, the reward function associated to φ is simply

$$r(\lambda) = \begin{cases} 1, & \text{if } \varphi(\lambda) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.20)$$

Let $\varepsilon = 0.01$ be the tolerance of the study, then let γ_+ and γ_- be the functions encompassing the tolerance ε w.r.t. γ . Let $\lambda \in \Lambda$ be a parameter value, then let $\tilde{\mathbf{x}}^\lambda$ be the approximate solution to the ODE induced by Eq. (2.3), λ and $x_0 = \gamma_1(1)$. We consider the auxiliary properties φ_- and φ_+ defined as follows:

$$\begin{aligned} \varphi_-(\lambda) &:= \forall t \in \mathcal{T}, \|\gamma_{-,1}(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_{-,2}(t), \\ \varphi_+(\lambda) &:= \forall t \in \mathcal{T}, \|\gamma_{+,1}(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_{+,2}(t), \end{aligned} \quad (2.21)$$

Note that φ_-, φ_+ vary continuously w.r.t. ε . Their respective auxiliary reward functions

r_- and r_+ are

$$r_-(\lambda) = \begin{cases} 1, & \text{if } \varphi_-(\lambda) \text{ is true,} \\ 0, & \text{otherwise,} \end{cases} \quad r_+(\lambda) = \begin{cases} 1, & \text{if } \varphi_+(\lambda) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.22)$$

It is important to note here that r_-, r_+ meet the criteria given in Definition 2.1.13:

- r_-, r_+ are piecewise continuous on Λ , since the trajectories $\tilde{\mathbf{x}}^\lambda$ are continuous w.r.t. λ on the compact Λ , thus the distance between the trajectories and the (fixed) dataset is continuous w.r.t. λ : r_-, r_+ are Lebesgue-integrable on Λ ;
- for any $\lambda \in \Lambda$, $r_-(\lambda), r(\lambda), r_+(\lambda) \in \{0, 1\}$, thus we only need to prove $r_-(\lambda) = 1 \Rightarrow r(\lambda) = 1$ and $r(\lambda) = 1 \Rightarrow r_+(\lambda) = 1$ to ensure the inequality of point 2 in Definition 2.1.13. We have

$$\begin{aligned} r_-(\lambda) = 1 &\Leftrightarrow \|\gamma_{-,1}(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_{-,2}(t), \\ &\Leftrightarrow \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_2(t) - \varepsilon, \\ &\Rightarrow \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_2(t), \\ &\Rightarrow r(\lambda) = 1. \end{aligned} \quad (2.23)$$

Similarly,

$$\begin{aligned} r(\lambda) = 1 &\Leftrightarrow \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_2(t), \\ &\Rightarrow \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_2(t) + \varepsilon, \\ &\Rightarrow \|\gamma_{+,1}(t) - \tilde{\mathbf{x}}^\lambda(t)\|_{\mathbb{R}} \leq \gamma_{+,2}(t), \\ &\Rightarrow r_+(\lambda) = 1. \end{aligned} \quad (2.24)$$

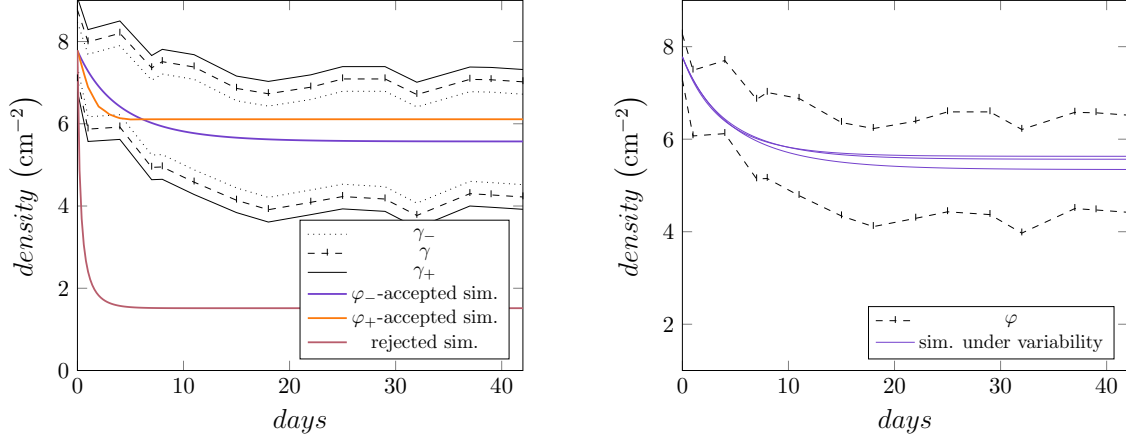
Hence, r_-, r_+ satisfy $r_-(\lambda) \leq r(\lambda) \leq r_+(\lambda)$.

- for any $\lambda \in \Lambda$, we have

$$\begin{aligned} \lim_{\varepsilon \rightarrow 0} \gamma_2(t) - \varepsilon &= \gamma_2(t) = \lim_{\varepsilon \rightarrow 0} \gamma_2(t) + \varepsilon, \\ \Leftrightarrow \lim_{\varepsilon \rightarrow 0} \varphi_-(\lambda) &= \varphi(\lambda) = \lim_{\varepsilon \rightarrow 0} \varphi_+(\lambda), \\ \Leftrightarrow \lim_{\varepsilon \rightarrow 0} r_-(\lambda) &= r(\lambda) = \lim_{\varepsilon \rightarrow 0} r_+(\lambda). \end{aligned} \quad (2.25)$$

□

The setup is depicted in Fig. 2.2. In Fig. 2.2a we depict the general idea behind properties φ, φ_- and φ_+ by depicting trajectories that are either accepted or rejected w.r.t. the properties; in Fig. 2.2b, we depict the setup of the Monte-Carlo procedure, where several values λ are sampled in a neighborhood of a value of interest λ^* before their induced trajectories are computed.



(a) Different curves that are accepted or rejected by properties φ_- , φ_+ .

(b) Curves obtained by repeatedly simulating the model under variability of the parameter λ around a fixed value λ^* .

Figure 2.2: Depictions of experimental and simulation data. The black solid, dashed and dotted lines represent the tunnels corresponding to properties φ_+ , φ and φ_- respectively. The coloured curves are the trajectories induced by different parameter values.

As stated in Section 1.2.3 and in particular in Eq. (1.17), it is possible to compute a value for the integration step h that is sufficiently small to ensure that the resulting global approximation error $\varepsilon(\mathcal{T}, \lambda)$ is smaller than ε for any $\lambda \in \Lambda$, provided the successive derivatives of f can be bounded by M and L . In our case, it may be computed that $M = L = 7$ fit the requirement. Using Eq. (1.17) to compute a suitable value for h , we have

$$\begin{aligned}
 0.01 &\leq T \frac{73}{720} M L^4 h^4 \\
 \Leftrightarrow h &\leq \sqrt[4]{\frac{720}{73 T M L^4}} \\
 \Leftrightarrow h &\leq \sqrt[4]{\frac{720}{73 \times 42 \times 7 \times 7^4}} \\
 \Leftrightarrow h &\leq 0.0611
 \end{aligned} \tag{2.26}$$

We may pick $h = 0.06$ and be assured that the corresponding global approximation error satisfies $\varepsilon(\mathcal{T}, \lambda) \leq 0.01$ for any $\lambda \in \Lambda$. $\blacktriangleleft$

Given a reward function $r(\cdot)$ to evaluate parameters λ^* and a tolerance ε , we are now interested in the auxiliary reward functions $r_-(\cdot)$ and $r_+(\cdot)$ as defined in Definition 2.1.13: we want to establish a statistical guarantee on the expected value $\mathbb{E}[r]$ of $r(\cdot)$ on a neighborhood of λ^* , using the estimators $\hat{r}_-$ and $\hat{r}_+$ of $r_-(\cdot)$ and $r_+(\cdot)$, respectively.

This guarantee is established in Theorem 2.1.15, which constitutes the second result of the thesis. Intuitively, given a parameter λ^* , it provides two statistical guarantees: a confidence interval for the expected value $\mathbb{E}[r(\cdot)]$ of the reward function $r(\cdot)$ on a ball of radius $\rho > 0$ around λ^* , and a statistical guarantee on the size of that interval.

Theorem 2.1.15 (Stability of the approximation w.r.t. parameter λ).

Let $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining a parametric ODE of variable $z(t)$, then let z_0 be an initial condition for z . Let $\rho > 0$ be a radius and $\varepsilon > 0$ be the target integration error for the ODE integration. Let $\lambda^* \in \mathring{\Lambda}$ be a parameter to evaluate, such that $\mathcal{B}_{\lambda^*} = \{\lambda \in \mathbb{R}^m \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\} \subseteq \Lambda$. Let $\theta \in (0, 1)$ be the risk and $\alpha > 0$ be the precision for the Monte-Carlo procedure. Let $r : \Lambda \rightarrow \mathbb{R}$ be a bounded reward function on Λ ; we consider the greatest (resp. lowest) value U (resp. L) of $r(\cdot)$ on $\mathcal{B}_{\lambda^*}$.

Consider the hidden risk $\xi = 1 - \sqrt{1 - \theta}$. The auxiliary rewards $r_{-\varepsilon}$ and $r_{+\varepsilon}$ satisfy

$$\mathbb{P}(r_- \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{-\varepsilon} + \alpha]) \geq 1 - \xi, \quad \mathbb{P}(r_+ \in [\hat{r}_{+\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \xi, \quad (2.27)$$

where the estimators $\hat{r}_{-\varepsilon} = \hat{r}_{-\varepsilon}(\lambda^*)$ and $\hat{r}_{+\varepsilon} = \hat{r}_{+\varepsilon}(\lambda^*)$ are independently computed after performing, for each estimator, a number $N' = \frac{\log(2/\xi)}{2\alpha^2}(U - L)^2$ of simulations induced by f , z_0 and a parameter λ sampled in $\mathcal{B}_{\lambda^*}$.

Furthermore, there exists $\varepsilon^* > 0$ sufficiently small, such that for any tolerance $\varepsilon < \varepsilon^*$ the following statements hold:

- the expected value $r = \mathbb{E}_{\mathcal{B}_{\lambda^*}}[r]$ of reward function $r(\cdot)$ on $\mathcal{B}_{\lambda^*}$ satisfies

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \theta; \quad (2.28)$$

- the distance between $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ satisfies

$$\mathbb{P}(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha) \geq 1 - \theta. \quad (2.29)$$

◁

Theorem 2.1.15 establishes a confidence interval for the expected value $r = \mathbb{E}_{\mathcal{B}_{\lambda^*}}[r]$ by merging the respective intervals for $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r_{-\varepsilon}]$ and $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r_{+\varepsilon}]$. Its proof is given in Section B.2, on page 126; it takes advantage of Lemma 2.1.12. Note that the final interval, that is $[\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]$ in Eq. (2.28), is of size 5α or small with probability $1 - \theta$: α must be tweaked so that the final interval is of the desired size, e.g. $\alpha = 0.02$ if the desired diameter of the interval is 0.1.

2.1.4 Complete study of the running example

We now conduct a complete study of the running example. Recall that the model is defined by the following ODE:

$$\frac{dx}{dt}(t) = ax(t)\left(1 - \frac{x(t)}{b}\right). \quad (2.30)$$

We are interested in finding values for a and b such that the induced model fits the experimental dataset γ provided in the original study [52]. To this end, we will scan a subset Λ of $\mathbb{R}^2$ for values that induce trajectories which stay within the tunnel induced by the dataset under condition of variability, as depicted in Fig. 2.2b. The dataset of interest is extracted from Table 1 in [52]: we consider a set $\mathcal{T} = \{0, 1, 4, \dots, 42\}$ of timepoints, as well as the experimental values x_k and standard error e_k associated to each timepoint. For convenience, we define the function

$$\begin{aligned} \gamma : \mathcal{T} &\rightarrow \mathbb{R} \times \mathbb{R}_{\geq 0} \\ t_k &\mapsto (x_k, e_k). \end{aligned} \quad (2.31)$$

We consider the parameter set $\Lambda = [0, 3] \times [0, 9]$ of parameters for Eq. (2.30). Let $\Delta_\lambda = 0.01$ be the discretization step for Λ . Let $\rho = 0.005$ be the radius of the ball of variability around parameters. For each $\lambda^* \in \Lambda^* = \{\lambda \in \mathring{\Lambda} \mid \exists i, j \in \mathbb{N}_{>0}, \lambda = (i\Delta_\lambda, j\Delta_\lambda)\}$, we define the ball $\mathcal{B}_{\lambda^*} = \{\lambda \in \Lambda \mid \|\lambda^* - \lambda\| \leq \rho\}$ in which the Monte-Carlo procedure for λ^* will be conducted. Let $\theta = \alpha = 0.05$ be the target risk and precision of the Monte-Carlo procedure. Theorem 2.1.15 states that $N = 874$ simulations must be performed for each auxiliary property to ensure these values are met, since in that case the random variables follow a Bernoulli distribution (i.e. $L = 0, U = 1$).

Note here that Theorem 2.1.15 establishes a probabilistic guarantee on the upper

bound of the size of the confidence interval (here, $5\alpha = 0.25$), which does not prevent $\hat{r}_-$ and $\hat{r}_+$ to be closer in practice: in this experiment, the distance between the two values was generally between 0.05 and 0.1. For parameter values inducing trajectories that are closer to the edge of the tunnel, that distance went up to 1 (i.e. r_+ accepted everything, and r_- rejected everything), which is more a sign of a strong model stability w.r.t. λ than anything else, since it shows that the distance between simulations is less than 2ε .

For each parameter λ , we use the Runge-Kutta 4 method described in Eq. (1.12) to compute the trajectory. Let $h = 0.05$ be the integration step of the method, which is less than the value computed in Eq. (2.26), then let $\mathbf{T} = \{t \in [0, 42] \mid \exists i \in \mathbb{N}, t = hi\}$ be a finite discretization of the timeline: as computed in Eq. (2.26), the global approximation error $\varepsilon(\mathbf{T}, \lambda)$ is such that $\varepsilon(\mathbf{T}, \lambda) \leq 0.01$ for any $\lambda \in \Lambda$. For any $\lambda \in \mathcal{B}_{\lambda^*}$, the associated *exact* solution is denoted $\mathbf{x}^\lambda$, and the *approximated* solution is denoted $\tilde{\mathbf{x}}^\lambda$. Finally, given the experimental value $\gamma_1(t)$ and the standard error $\gamma_2(t)$ for each time point t of the experiment (see Eq. (2.31), we consider the property

$$\varphi(\lambda) = \forall t \in \mathcal{T}, \|\gamma_1(t) - \mathbf{x}^\lambda(t)\| \leq \gamma_2(t). \quad (2.32)$$

The reward function associated to φ is

$$r(\lambda) = \begin{cases} 1, & \text{if } \varphi(\lambda) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.33)$$

The auxiliary properties are

$$\begin{aligned} \varphi_-(\lambda) &= \forall t \in \mathcal{T}, \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\| \leq \gamma_2(t) - \varepsilon, \\ \varphi_+(\lambda) &= \forall t \in \mathcal{T}, \|\gamma_1(t) - \tilde{\mathbf{x}}^\lambda(t)\| \leq \gamma_2(t) + \varepsilon, \end{aligned} \quad (2.34)$$

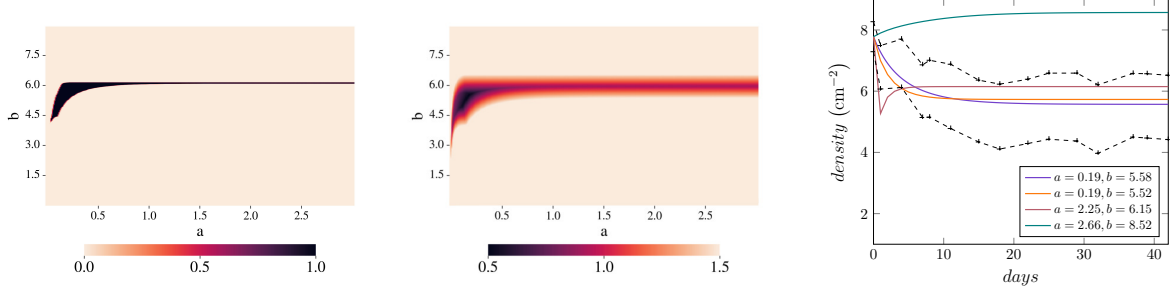
and their respective associated reward functions are

$$r_-(\lambda) = \begin{cases} 1, & \text{if } \varphi_-(\lambda) \text{ is true,} \\ 0, & \text{otherwise,} \end{cases} \quad r_+(\lambda) = \begin{cases} 1, & \text{if } \varphi_+(\lambda) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.35)$$

For each $\lambda^* \in \Lambda^*$, we estimate the expected value $\mathbb{E}[r_-]$ and $\mathbb{E}[r_+]$ of the rewards $r_-(\cdot)$ and $r_+(\cdot)$ on $\mathcal{B}_{\lambda^*}$ by computing the estimators $\hat{r}_-$ and $\hat{r}_+$ through the Monte-Carlo procedure. We will use $\hat{r}_-$ to conduct the study and sort values, as it is the more conservative

indicator.

The simulations were performed on a 2.1 GHz Intel Xeon Silver 4216 processor, running g++ version 7.5.0 on Ubuntu 18.04, and completed in 52 minutes on 56 threads. The results are presented in Fig. 2.3.



(a) Heatmap of the Monte-Carlo estimator $\hat{r}_-$ relative to the expected probability of satisfying φ_- .

(b) Heatmap of the Monte-Carlo estimator $\hat{d}_+$ relative to the expected distance to the dataset.

(c) Trajectories induced by Eq. (2.3) and different parameter values.

Figure 2.3: Results of the study of the parameterization of Eq. (2.3).

In Fig. 2.3a, we represent the score of all parameter values $\lambda^* \in \Lambda^*$, in which the pink zones are zones where $\text{grade}(\lambda^*) = \frac{1}{874} \sum_{i=1}^{874} r_-(\lambda_i) = 0$. As we can see, the gradient is rather strong and doesn't allow discrimination between values in this area: the model is very stable and the parameter either work well w.r.t. φ , or not at all.

In order to refine the result, we then changed our reward function to the distance $d(\lambda)$ between the experimental dataset and the trajectory of the system (see the beginning of Section 2.1.2, especially on page 41). This reward function is defined simply as the supremum distance between the dataset γ and the trajectory $\mathbf{z}^\lambda$, i.e. :

$$d(\lambda) = \max_{t \in \mathcal{T}} \|\gamma_1(t) - \mathbf{z}^\lambda(t)\|_{\mathbb{R}}. \quad (2.36)$$

The auxiliary reward functions $d_-(\lambda)$ and $d_+(\lambda)$ are defined as $d_- : \lambda \mapsto -\varepsilon + \max_{t \in \mathcal{T}} \|\gamma_1(t) - \tilde{\mathbf{z}}^\lambda(t)\|_{\mathbb{R}}$ and $d_+ : \lambda \mapsto \varepsilon + \max_{t \in \mathcal{T}} \|\gamma_1(t) - \tilde{\mathbf{z}}^\lambda(t)\|_{\mathbb{R}}$. The reward functions d, d_-, d_+ obviously respect the conditions presented in Definition 2.1.13, as they are continuous w.r.t. λ (since the trajectories are continuous w.r.t. λ), they satisfy $d_-(\lambda) \leq d(\lambda) \leq d_+(\lambda)$ and they are continuous w.r.t. ε , thus $\lim_{\varepsilon \rightarrow 0} d_-(\lambda) = \lim_{\varepsilon \rightarrow 0} d_+(\lambda) = d(\lambda)$; we focused on d_+ as it is, again, the more conservative function.

The estimation of the expected value of the reward function d_+ is presented in Fig. 2.3b. We piggybacked on the computations of the first study, because the stability of the model

w.r.t. λ implies a low variation of trajectory: experimentally, the maximum distance between trajectories induced by parameters within the same ball $\mathcal{B}_{\lambda^*}$ was lower than 0.5. Figure 2.3b shows a tighter area of values which induce trajectories that are very close to the dataset (down to 0.05 polyps/cm²), plotted in dark purple; this area contains the parameter value ($a = 0.13, b = 5.35$) estimated by the authors of the original study [52]: it comforts us in saying that this method provides tangible results. The pair ($a = 0.19, b = 0.558$) induced the closest trajectories (w.r.t. supremum distance) from the dataset under condition of variability; it produces the purple curve in Fig. 2.3c.

2.2 Stability analysis of a model

In Section 2.1, we were interested in the influence of the parameter λ on the behaviour of the model; we now move to the study of the influence of the initial condition z_0 on the behaviour of the model. More specifically, we are interested in studying the stability of the system under the variability of that initial condition: similarly to what we did in Section 2.1, we want to evaluate the variations on the trajectories of the system that are induced by small variations on the initial condition; to do that, we will perform the Monte-Carlo procedure similarly as in Section 2.1.2, but in balls around initial conditions z_0^* of interest instead of parameters λ^* . As intuited in Remark 2.1.6, the Monte-Carlo procedure is the same whether we study the parameter λ or the initial condition z_0 in the model defined by Eq. (1.6). The study itself, on the other hand, must be adjusted as we are not working with the same concepts: rather than assessing data adequacy we will try and describe the attractors of a system along with their respective basins of attraction, which are defined in Definition 2.2.1.

Definition 2.2.1 (Attractor (see [53], first definition in Section 2 p.179)). Let f be a function defining an ODE for variable $z \in \Omega$, where Ω denotes the phase space of the system.

An attractor is a closed subset A of Ω characterized by the following properties:

- the basin of attraction of A , denoted $\sigma(A)$, which encompasses all points $z_0 \in \Omega$ such that the trajectory $\mathbf{z}(t, z_0) \in A$ when $t \rightarrow \infty$, is of strictly positive measure;
- there is no proper closed subset A' of A whose basin of attraction $\sigma(A')$ coincides with that of A .

◁

Remark 2.2.2. Many definitions of an attractor coexist. The definition of Milnor [53] is convenient in that it is intuitive, which is all we ask for in this work, although the property of minimality is not specifically required here. Other more intricate definitions (see for instance [69] and [27], Section 5.4) involving topology considerations are left to the interest of the reader. ◀

2.2.1 Running example: a study on Duffing's oscillator [33]

Duffing's oscillator, introduced in his book [21], is a nonlinear second order equation used for modeling damped and driven oscillators. It is given by the following equation:

$$\frac{d^2x}{dt^2}(t) + a\frac{dx}{dt}(t) + bx(t) + dx^3(t) = c \cos(kt), \quad (2.37)$$

where a controls the damping, b controls the linear stiffness, d controls the nonlinearity of the restoring force, and c and k are the amplitude and angular frequency of the periodic driving force, respectively. It is a well-studied system [29, 54, 76] and a primary example of chaotic behaviour. Here, we will focus on a slightly modified equation, which was first studied by Holmes and Rand [33]:

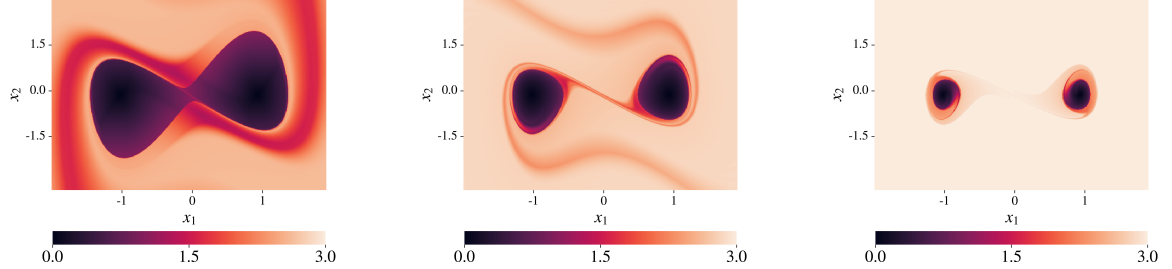
$$\frac{d^2x}{dt^2}(t) + a\frac{dx}{dt}(t) + cx^2\frac{dx}{dt}(t) + bx(t) + dx^3(t) = 0, \quad (2.38)$$

We rewrite it as a first-order system and obtain the following two-dimensional system:

$$\begin{cases} \frac{dx_1}{dt}(t) = x_2(t), \\ \frac{dx_2}{dt}(t) = -bx_1(t) - ax_2(t) - dx_1^3(t) - cx_1^2(t)x_2(t). \end{cases} \quad (2.39)$$

This system exhibits different behaviours depending on the value of (a, b, c, d) . Notably, if $b < 0$ and $c, d > 0$, symbolic study (see [33], Section 2.3) predicts that two equilibria $x_l = (-1, 0)$ and $x_r = (1, 0)$ appear, whose basins of attraction vary in size w.r.t. $|a|$. This is confirmed by Fig. 2.4, which depicts the expected supremum distance between the two equilibria and the forward trajectory induced by each initial condition: the darker the point, the closer the trajectory is expected to be from one of the two equilibria. Note that this figure is only a graphical intuition of the influence of the parameter value on the stability of the model, as providing guarantees for the study of the perturba-

tions of both the parameter and the initial condition at once is beyond the scope of this thesis.



(a) Basin of attraction when $\lambda = (-0.8, -1, 1, 1)$. (b) Basin of attraction when $\lambda = (-1, -1, 1, 1)$. (c) Basin of attraction when $\lambda = (-1.2, -1, 1, 1)$.

Figure 2.4: Basins of attraction for different values of parameter a . The color depicts the expected distance between the two equilibria and the trajectory of duration $T = 10$ seconds induced by each discrete value in $[-2, 2] \times [-3, 3]$: the darker the point, the closer the forward trajectory of the induced model is expected to be from one of the equilibria. Note that this study is merely a graphical intuition of the influence of the parameter on the stability of the model.

For the remaining of the study, we lock the parameter to $\lambda = (-1, -1, 1, 1)$ so that the initial condition of the system is the only variable in the study. The phase portrait of the model induced by λ is depicted in Fig. 2.5; it exhibits a clockwise orbiting motion around two points, in what may be described as a “bow tie”-shaped trajectory, and seems to seek a stationary orbit: the trajectories induced by an initial condition outside the $[-2, 2]$ interval for x_1 go inwards, while the trajectories induced by an initial condition inside the interval go outwards.

2.2.2 Adaptation of mathematical notions

The stability analysis of a system is a fundamental component of the study of dynamical systems. As said in the beginning of this section, the goal is to assess the influence of the initial condition on the trajectories of a system; intuitively, we want to determine if small perturbations of the initial conditions of the ODE induce small differences between trajectories. In the following, given a function f defining an ODE in the form of Eq. (1.5) as well as an initial condition z_0 , the exact (resp. approximate) solution to the induced Cauchy problem is written $\mathbf{z}(t, z_0) = \mathbf{z}^{z_0}(t)$ (resp. $\tilde{\mathbf{z}}(t, z_0) = \tilde{\mathbf{z}}^{z_0}(t)$). Since λ is fixed for the whole study, it does not appear in the writing of f nor $\mathbf{z}$ (resp. $\tilde{\mathbf{z}}$). First, we recall

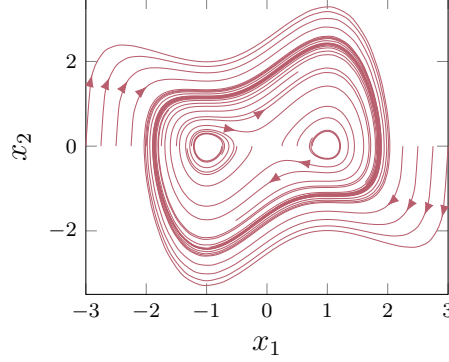


Figure 2.5: Phase space of the model induced by Eq. (2.39) and $\lambda = (-1, -1, 1, 1)$. All initial conditions lie on the line satisfying $x_2 = 0$. The system seems to revolve clockwise around two equilibrium points located at $(-1, 0)$ and $(1, 0)$, seeking a stationary orbit.

the mathematical definition of the two forms of stability which we aim at studying:

Definition 2.2.3 (Local stability of a differential system). Let $n \in \mathbb{N}_{>0}$ be a positive integer. Let $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function defining an ODE for variable $z \in \mathbb{R}^n$. Let z_e be an equilibrium for f , that is $\forall t \in \mathbb{R}, f(t, z_e) = 0$.

- The equilibrium z_e is said to be **Lyapunov stable** [50] if, for every $\rho > 0$, there exists $\rho^* > 0$ such that $\|z_e - z_0\|_{\mathbb{R}^n} \leq \rho^* \Rightarrow \|z_e - \mathbf{z}^{z_0}(t)\|_{\mathbb{R}^n} \leq \rho, \forall t \geq 0$;
- The equilibrium z_e is said to be **asymptotically stable** if it is Lyapunov stable and there exists $\rho^* > 0$ such that $\|z_e - z_0\|_{\mathbb{R}^n} \leq \rho^* \Rightarrow \lim_{t \rightarrow \infty} \|z_e - \mathbf{z}^{z_0}(t)\|_{\mathbb{R}^n} = 0$.

◁

Intuitively, Lyapunov stability means that for every radius $\rho > 0$ there exists a radius ρ^* such that for every initial condition at a distance less than ρ^* from z_e , the distance between the induced trajectory $\mathbf{z}^{z_0}$ and z_e will be less than ρ for any $t \geq 0$; asymptotical stability means that there exists $\rho^* > 0$ such that for every initial condition at a distance less than ρ^* from z_e , the induced trajectory $\mathbf{z}^{z_0}$ will come closer to z_e as t increases. In order to verify such notions on ODE models and take approximations into account (either from the approximation method or the SMC techniques), we propose a weaker notion of stability:

Definition 2.2.4 ((ρ, δ) -stability). Let $d(\cdot, \cdot)$ define a distance between functions in $\mathbb{R}^n \times \mathbb{R}^n$. Let $\rho > 0$ and $\delta > 0$ be the tolerances of the stability. Let $z_0 \in \Omega$ be an initial condition for the model, then let $\mathbf{z}^{z_0}(t)$ be the trajectory of that model.

This solution is called (ρ, δ) -stable if, given another initial condition $z'_0 \in \Omega$, the trajectory $\mathbf{z}^{z'_0}(t)$ satisfies

$$\|z_0 - z'_0\|_{\mathbb{R}^n} \leq \rho \Rightarrow d(\mathbf{z}^{z_0}, \mathbf{z}^{z'_0}) \leq \delta. \quad (2.40)$$

◁

In our case, we will consider stability with respect to a single equilibrium point z_e , which we will compare to trajectories $\mathbf{z}^{z_0}(t)$, for $z_0 \in \Omega$. In this context, (ρ, δ) -stability amounts to verifying whether the stability property w.r.t. z_e , as defined in Definition 2.2.4, holds for initial conditions z_0 such that $\|z_0 - z_e\|_{\mathbb{R}^n} \leq \rho$. Computing this ratio of accepted initial conditions translates to evaluating the probability that a system undergoing small perturbations of its initial condition around z_e satisfies Definition 2.2.4.

The study procedure will be conducted on initial conditions in a way similar to that on ODE parameters: first, a subset $\mathcal{I}$ of initial conditions in the phase space Ω will be discretized; then, for each initial condition $z_0^* \in \mathcal{I}$ to be tested, the Monte-Carlo procedure will be performed by sampling a certain amount of initial conditions in the vicinity of z_0^* , computing the reward associated to each simulation, and averaging these values to estimate the expected reward for a model induced by an initial condition sampled in the vicinity of z_0^* ; finally, these estimators will be used to elect the best (w.r.t. some property) initial values under variability or, as we will see in Section 2.2.3, experimentally depict shapes of interest within the phase space.

Similarly to the study on parameters in Section 2.1, we prove guarantees regarding the stability of the trajectories w.r.t. variations of the initial conditions. First, we have a lemma corresponding to Lemma 2.1.12, i.e. the global approximation error $\varepsilon(\mathcal{T}, z_0)$, defined in Eq. (1.14) on page 21 and relative to the simulation induced by an initial condition z_0 in the phase space Ω , may be arbitrarily bounded on a compact subset of Ω .

Lemma 2.2.5 (Uniform bound for the global approximation error w.r.t. variations of the initial condition).

Let $T > 0$ be a duration for the simulations. Let n be a positive integer. Let $f : \mathbb{R}^+ \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function defining an ODE in the form of Eq. (1.5). Let $\Omega \subseteq \mathbb{R}^n$ be the (compact)

phase space of f , then let $z_0^* \in \dot{\Omega}$ be an initial condition. Let $\rho > 0$ be a radius, such that the ball $\mathcal{B}_{z_0^*} = \{z_0 \in \mathbb{R}^n \mid \|z_0^* - z_0\|_{\mathbb{R}^n} \leq \rho\} \subseteq \Omega$.

Let $(h_i)_{i \in \mathbb{N}} \in \mathbb{R}^+$ be a sequence of integration steps, such that $\lim_{i \rightarrow \infty} h_i = 0$; for each h_i , we denote $\mathcal{T}_i = \{t_k \in [0, T) \mid \exists i \in \mathbb{N}, t_k = k \times h_i\}$ the discretization it induces. For any $\varepsilon > 0$, there exists $i^* \in \mathbb{N}$ such that

$$\forall i > i^*, \forall z_0 \in \mathcal{B}_{z_0^*}, \quad \varepsilon(\mathcal{T}_i, z_0) < \varepsilon. \quad (2.41)$$

◁

The proof of Lemma 2.2.5 is given in Section B.3, on page 127. Similarly to Lemma 2.1.12, Lemma 2.2.5 can be supplemented by an explicit choice of a sufficiently small integration step h so that the approximation error $\varepsilon(\mathcal{T}, \cdot)$ is bounded on $\mathcal{I}$ w.r.t. a discretization $\mathcal{T}$ of the timeline, provided the integration method comes with an appropriate estimation of the global error. It has been discussed in Section 1.2.3 and Eq. (1.17).

Example 2.2.6. Similarly to Eq. (1.16), it is possible to compute a value for integration step h when using the Runge-Kutta 4 method, such that the approximation error is bounded on the compact set of study (see [49]):

$$\varepsilon^1(h) \leq \frac{973}{720} ML^4 h^5. \quad (2.42)$$

Then, the global approximation error introduced in a complete simulation of duration T is

$$\varepsilon(\mathcal{T}, z_0) \leq T \frac{973}{720} ML^4 h^4. \quad (2.43)$$

◀

Second, using the definition of auxiliary reward functions (see Definition 2.1.13), we have a theorem corresponding to Theorem 2.1.15; it provides guarantees on the confidence interval on the expected value $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r(\cdot)]$ of the reward obtained by an initial condition z_0 sampled uniformly at random in a compact neighborhood $\mathcal{B}_{z_0^*}$ of z_0^* .

Remark 2.2.7. Recall that a reward function r is mapping a real value to an input of interest (in our case, an initial condition for a differential model); it may or may not be linked to a specific property, and is used to sort the input values in order to find the best

ones w.r.t. that reward. In this Section 2.2, we want to find the initial conditions that induce the trajectories that stay in the vicinity of a particular equilibrium point. To this end, we will define two auxiliary reward functions r_- and r_+ that satisfy Definition 2.1.13 (see page 48) to circumvent the errors induced by the approximation methods. ◀

Theorem 2.2.8 (Stability of the approximation w.r.t. initial condition z_0).

Let $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function defining an ODE of variable z . Let $\rho > 0$ be a radius and $\varepsilon > 0$ be the target integration error for the ODE integration. Let $z_0^* \in \Omega$ be an initial condition to evaluate, such that $\mathcal{B}_{z_0^*} = \{z_0 \in \mathbb{R}^n \mid \|z_0^* - z_0\|_{\mathbb{R}^n} \leq \rho\} \subseteq \mathring{\Omega}$. Let $\theta \in (0, 1)$ be the risk and $\alpha > 0$ be the precision for the Monte-Carlo procedure. Let $r : \mathcal{I} \rightarrow \mathbb{R}$ be a bounded reward function on $\mathcal{I}$; we consider the greatest (resp. lowest) value U (resp. L) of $r(\cdot)$ on $\mathcal{B}_{z_0^*}$.

Consider the hidden risk $\xi = 1 - \sqrt{1 - \theta}$. The auxiliary rewards $r_{-\varepsilon}$ and $r_{+\varepsilon}$ defined in Definition 2.1.13 satisfy

$$\mathbb{P}(r_- \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{-\varepsilon} + \alpha]) \geq 1 - \xi, \quad \mathbb{P}(r_+ \in [\hat{r}_{+\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \xi, \quad (2.44)$$

where the estimators $\hat{r}_{-\varepsilon} = \hat{r}_{-\varepsilon}(z_0^*)$ and $\hat{r}_{+\varepsilon} = \hat{r}_{+\varepsilon}(z_0^*)$ are indepently computed after performing, for each estimator, a number $N' = \frac{\log(2/\xi)}{2\alpha^2} (U - L)^2$ of simulations induced by f and an initial condition z_0 sampled in $\mathcal{B}_{z_0^*}$.

Furthermore, there exists $\varepsilon^* > 0$ sufficiently small, such that for any tolerance $\varepsilon < \varepsilon^*$ the following statements hold:

- the expected value $r = \mathbb{E}_{\mathcal{B}_{z_0^*}}[r]$ of reward function $r(\cdot)$ on $\mathcal{B}_{z_0^*}$ satisfies the estimation

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \theta \quad (2.45)$$

- the distance between $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ satisfies:

$$\mathbb{P}(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha) \geq 1 - \theta \quad (2.46)$$

◀

Theorem 2.2.8 establishes a confidence interval for the expected value $r = \mathbb{E}_{\mathcal{B}_{z_0^*}}[r]$ by merging the respective intervals for $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r_-]$ and $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r_+]$. Its proof is given in Section B.4, on page 130; similarly to Theorem 2.1.15, it takes advantage of Lemma 2.2.5.

Again, note that the size of the final interval, that is $[\hat{r}_- - \alpha, \hat{r}_+ + \alpha]$ in Eq. (2.45) is of size 5α : α must be tweaked so that the final interval is of the desired size, e.g. $\alpha = 0.02$ if the desired diameter of the interval is 0.1.

Remark 2.2.9. Note that the term “stability” involved in Theorem 2.2.8 (and Theorem 2.1.15) is relative to the integration method (i.e. small perturbations in the settings of the simulation lead to drastic changes in the trajectory) and **not** the system itself (i.e. (ρ, δ) -stability as defined in Definition 2.2.4), which is the topic of this section. ◀

Example 2.2.10 (Running example). We recall that the study of the behaviour of the model under condition of variability for both the parameter and the initial condition is beyond the scope of this work. Hence, for the remaining of this section, the parameter $\lambda = (-1, -1, 1, 1)$ is fixed, to conduct a study on the initial condition alone.

First, we want to experimentally show the (ρ, δ) -stability of the two equilibria x_l and x_r predicted by symbolic analysis. To describe the first part of the study, we will focus on the left equilibrium $x_l = (-1, 0)$. We fix a distance $\delta = 1$ between candidate trajectories and the target point x_l , and we want to find the largest value $\rho > 0$ such that an initial condition $x_0 \in \mathcal{B}(x_l, \rho) = \{x_0 \in \mathbb{R}^2 \mid \|x_l - x_0\| \leq \rho\}$ induces a trajectory $\mathbf{x}^{x_0}(t)$ that maintains a distance below δ w.r.t. x_l , i.e. such that $d(x_l, \mathbf{x}^{x_0}) \leq \delta$. Let $T = 10$ seconds be the duration of the simulations. Given a candidate value for ρ , the property we aim at studying is

$$\varphi(\rho) = \forall x_0 \in \mathbb{R}^2, \|x_l - x_0\| \leq \rho \Rightarrow d(x_l, \mathbf{x}^{x_0}) \leq \delta \quad (2.47)$$

and its associated reward function is

$$r(\rho) = \begin{cases} 1, & \text{if } \varphi(\rho) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases} \quad (2.48)$$

In this study, we will consider several balls $\mathcal{B}(x_l, \rho)$ centered on x_l and of different radii ρ , in which we will perform the Monte-Carlo procedure by repeatedly sampling initial conditions and estimate the associated expected value $\mathbb{E}[r]$ using auxiliary reward functions, in the same fashion as in Section 2.1.4.

Again, the Runge-Kutta 4 method will be used to compute approximate solutions to Eq. (2.39). Since the system is very sensitive (as predicted in the initial study [33]), we need a very low global approximation error value ε ; we set $\varepsilon = 10^{-12}$. To apply Eq. (2.43),

we chose $M = 18$ and $L = 2$, then computed $T^{\frac{973}{720}}ML^4h^4 \leq \varepsilon \Leftrightarrow h \leq 2.25 \times 10^{-4}$, so that $h = 2.0 \times 10^{-4}$ fits the requirement.

Auxiliary properties and their associated reward functions are defined in a similar manner as in Example 2.1.14 on page 49:

$$\begin{aligned}\varphi_-(\rho) &= \forall x_0 \in \mathbb{R}^2, \|x_l - x_0\| \leq \rho \Rightarrow d(x_l, \mathbf{x}^{x_0}) \leq \delta - \varepsilon, \\ \varphi_+(\rho) &= \forall x_0 \in \mathbb{R}^2, \|x_l - x_0\| \leq \rho \Rightarrow d(x_l, \mathbf{x}^{x_0}) \leq \delta + \varepsilon,\end{aligned}\tag{2.49}$$

and

$$r_-(\rho) = \begin{cases} 1, & \text{if } \varphi_-(\rho) \text{ is true,} \\ 0, & \text{otherwise,} \end{cases} \quad r_+(\rho) = \begin{cases} 1, & \text{if } \varphi_+(\rho) \text{ is true,} \\ 0, & \text{otherwise.} \end{cases}\tag{2.50}$$

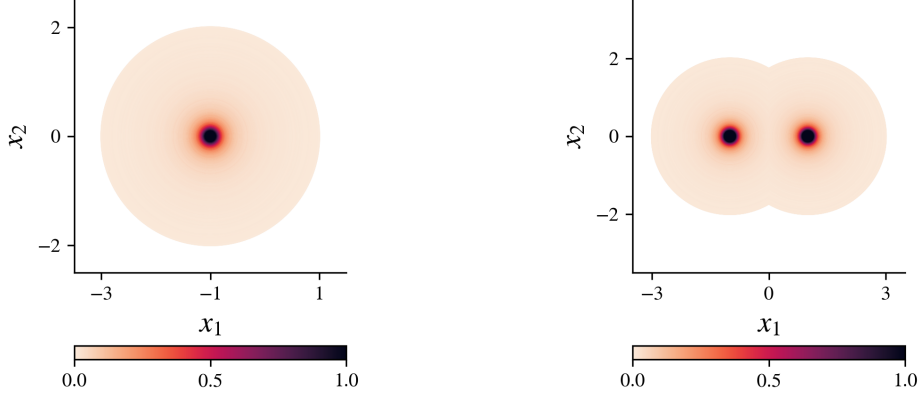
The proof that $r_-(\rho), r_+(\rho)$ are indeed auxiliary reward functions as defined in Definition 2.1.13 is analogous to that in Example 2.1.14 on ??.

We now fix the Monte-Carlo procedure settings, and as in Section 2.1.4 require a confidence interval of radius $\alpha = 0.05$ and a risk $\theta = 0.05$. To meet these requirements, we use Hoeffding's inequality [32] (see Eq. (2.6) for details) and again performed 874 simulations per auxiliary property. Since it is the more conservative property, we focus on φ_- . In order to find the largest value ρ satisfying φ_- , we set a discretization step $\Delta_\rho = 0.005$, and run computations for the candidate values of the set $\mathcal{C} = \{\rho_i \mid \rho_i \leq 2 \wedge \exists i \in \mathbb{N}_{>0}, \rho_i = i\Delta_\rho\}$. The results are depicted in Fig. 2.6a. The simulations were performed on a 2.1 GHz Intel Xeon Silver 4216 processor, running g++ version 7.5.0 on Ubuntu 18.04, and completed in 4 hours and 3 minutes on 48 threads.

Figure 2.6 depicts the probability that equilibria x_l and x_r are $(\rho, 1)$ -stable. Figure 2.6a focuses on the stability of x_l , while Fig. 2.6b depicts the stability of both x_l (on the left) and x_r (on the right). Figure 2.6b shows that both x_l and x_r are indeed attractors as predicted in [33], but also that their stability is precarious: the basin of attraction of either equilibrium does not seem big, as the probability of satisfying $\varphi(\rho)$ quickly decreases as ρ increases, i.e. the probability decreases (on average) as the initial condition departs from x_l and x_r . ◀

2.2.3 Estimating the shape of the basin of attraction

Now that stability has been experimentally exhibited, we turn ourselves to a more complicated problem from an analytical point of view: describing the shape of the basin of attraction. To this end, we will define new reward functions and perform a similar



(a) Heatmap of the expected score of satisfaction of φ_- , relative to $x_l = (-1, 0)$.

(b) Heatmap of the expected score of satisfaction of φ_- , relative to both $x_l = (-1, 0)$ and $x_r = (1, 0)$.

Figure 2.6: Depictions of stability analysis. The color of the circle refers to the expected probability that an initial condition sampled uniformly at random inside the circle stays at a distance below $\delta = 1$ from the equilibria; the darker the circle, the more likely the property is to be satisfied. White zones are zones where simulations were not performed.

study as in Section 2.1.4: discretizing the space of initial conditions, and, for each value, computing the expected distance between an equilibrium and a trajectory induced by an initial condition sampled in its neighborhood.

For the remaining of this section, we fix the function f defining an ODE for variable z with phase space Ω . Let us assume that an equilibrium z_e for f exists, such that $f(z_e) = 0$. Given an initial condition $z_0 \in \mathcal{I}$, we denote by $\mathbf{z}^{z_0}$ the trajectory induced by f and z_0 . Let $d(\cdot, \cdot)$ be a distance between functions in Ω . We want to experimentally describe the set $\{z_0 \in \Omega \mid d(z_e, \mathbf{z}^{z_0}(t)) \xrightarrow{\infty} 0\}$, that is the basin of attraction of z_e ; since we cannot perform infinite simulations, we instead consider the stability of the model for every initial condition of interest, i.e. the stability of fixed-time trajectories w.r.t. variability of the initial condition. To this end, we consider a tolerance $\delta > 0$ and a simulation duration $T > 0$, and are interested in describing the set $A = \{z_0 \in \Omega \mid d(z_e, \mathbf{z}^{z_0}) \leq \delta\}$ of initial conditions that induce an exact solution on $[0, T]$ that is at a distance at most δ from z_e .

Hence, the associated reward function is:

$$r : z_0 \mapsto d(z_e, \mathbf{z}^{z_0}). \quad (2.51)$$

Now, A may be written $A = \{z_0 \in \Omega \mid r(z_0) \leq \delta\}$. Again, we will use auxiliary reward functions to circumvent ODE limitations. First, we define the notion of dataset w.r.t. an equilibrium point:

$$\begin{aligned} \gamma_e : \mathcal{T} &\rightarrow \mathbb{R}^n \\ t &\mapsto z_e \end{aligned} \quad (2.52)$$

We fix the bound $\varepsilon > 0$ for the global approximation errors and compute a suitable integration step $h > 0$ ensuring this bound is met; we set $\mathcal{T} = \{t_k \in [0, T) \mid \exists k \in \mathbb{N}, t_k = k \times h\}$ and define the auxiliary reward functions in the following manner:

$$r_-(z_0) = |d(\gamma_e, \mathbf{z}^{z_0}) - \varepsilon|, \quad r_+(z_0) = |d(\gamma_e, \mathbf{z}^{z_0}) + \varepsilon|. \quad (2.53)$$

This time, the study will focus on r_+ , as it is the more conservative reward function. We define a radius $\rho > 0$ for the balls of variability in which the Monte-Carlo procedure is ran. For each value z_0^* to test, the Monte-Carlo procedure involving a precision $\alpha > 0$ and a risk $\theta \in (0, 1)$ is performed, so that the resulting value $\text{grade}(z_0^*)$ is the empirical mean of the rewards $r_+(z_0)$ obtained for the initial conditions sampled at random in the ball of radius ρ around z_0^* . Intuitively, $\text{grade}(z_0^*)$ is the expected distance between the equilibrium z_e and the trajectory induced by an initial condition sampled randomly in the vicinity of z_0^* .

Example 2.2.11 (Running example). This time, we analyzed a portion $\mathcal{I} = [-2, 2] \times [-1.5, 1.5]$ of the plan, using a discretization step $\Delta_{\mathcal{I}} = 0.01$. Let $\Delta_x = 0.01$ be the discretization step, then let $\rho = 0.005$ be the ball radius. Let $T = 10$ seconds be the duration of the simulations. The upper-bound for the global approximation error is $\varepsilon = 1.0 \times 10^{-12}$, and the integration step h to ensure this bound is met is $h = 2.0 \times 10^{-4}$. Let $\mathcal{T} = \{0, h, 2h, \dots, T\}$ be a discretization of $[0, T]$. Given an initial condition $x_0 \in \mathcal{I}$, we denote by $\mathbf{x}^{x_0}$ the trajectory induced by f and x_0 .

As the system is symmetric w.r.t. point reflection, we will focus on the equilibrium $x_l = (-1, 0)$. We write ν_l the function inducing a dataset w.r.t. x_l and $\mathcal{T}$. Let $z_1(\cdot), z_2(\cdot) :$

$\mathcal{T} \rightarrow \mathbb{R}^n$ be two functions. We consider the following distance:

$$d_{\mathcal{T}}(z_1(t), z_2(t)) = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \|z_1(t) - z_2(t)\|_{\mathbb{R}^n}, \quad (2.54)$$

Given an initial condition $x_0 \in \mathcal{I}$, the reward function associated to this distance and equilibrium x_l is:

$$r(x_0) = d_{\mathcal{T}}(\nu_l, \mathbf{x}^{x_0}) \quad (2.55)$$

Using integration method Runge-Kutta 4 with integration step h defined above, the auxiliary reward function of interest is

$$r_+(x_0) = d_{\mathcal{T}}(\nu_l, \mathbf{x}^{x_0}) + \varepsilon, \quad (2.56)$$

since we want to take into account the error in the worst possible case, i.e. if the actual trajectory is always further from x_l than the computed one.

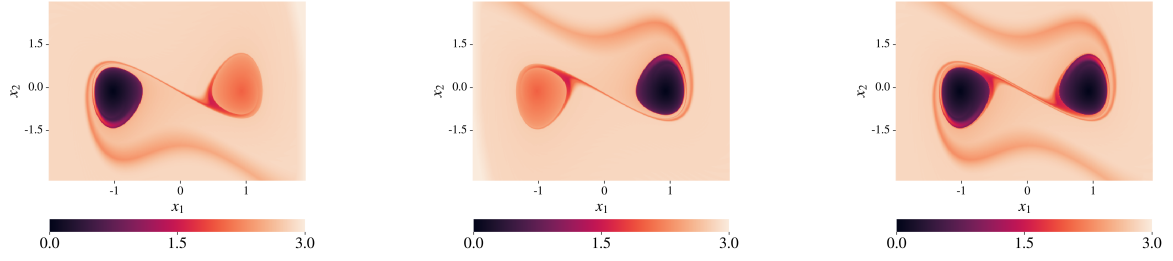
The Monte-Carlo parameters remain the same: $\alpha = \theta = 0.05$; we perform $N = 874$ simulations to meet this values. A first pass on the simulations showed that the distance between trajectories in the same ball $\mathcal{B}_{z_0^*}$ during the Monte-Carlo procedure was no more than 1 (actually below 0.2), hence we do not perform more simulations since the values L and U in Eq. (2.6) are such that $L - U \leq 1$ (see Remark 2.1.7).

Remark 2.2.12. In general, a careful theoretical study is required to ensure that L and U are indeed such that $L - U = 1$. Here, we settled on this approximation since empirical results suggested that $L - U \simeq 0.2$: it seemed extremely unlikely that after 874 simulations, the empirical spread (i.e. the maximum distance between any two trajectories) would be 5 times lower than the theoretical one. ◀

The simulations were performed on the same machine as Section 2.1.4, i.e. a 2.1 GHz Intel Xeon Silver 4216 processor, running g++ version 7.5.0 on Ubuntu 18.04; they completed in 10 hours and 29 minutes on 48 threads.

The results are presented in Fig. 2.7a; a similar study for the right equilibrium is depicted in Fig. 2.7b, and an aggregation of the two studies is depicted in Fig. 2.7c.

Figure 2.7 depicts the estimation of the reward function r_+ defined in Eq. (2.53), that is the expected average distance between equilibria and the trajectory induced by an initial condition sampled uniformly at random in a ball of radius $\rho = 0.005$ around



(a) Basin of attraction of the left equilibrium $x_l = (-1, 0)$.

(b) Basin of attraction of the right equilibrium $x_r = (1, 0)$.

(c) Basin of attraction of the two equilibria x_l and x_r .

Figure 2.7: Basins of attraction for different equilibria. The color represents the expected average distance between an equilibrium and the forward trajectory induced by Eq. (2.39) and an initial condition sampled uniformly at random in the vicinity of the colored point, for a duration of $T = 10$ seconds. The darker the point, the closer this trajectory is expected to be from the equilibrium.

each discrete value in $\mathcal{I}$: the darker the point, the closer this trajectory is expected to be from one of the equilibria. In Fig. 2.7a, we focus on the left equilibrium $x_l = (-1, 0)$; one can see an ovoid shape of dark purple on the left, which is consistent with the expected behaviour of the system in this portion of the space (see the original study [33]): again, this comforts us in saying that our method provides tangible results. The same study regarding the right equilibrium $x_r = (1, 0)$ is depicted in Fig. 2.7b, which is, as expected, a point reflection through $(0, 0)$ of Fig. 2.7a. Both studies show a darker orange zone around the converse equilibrium: this was also expected, since in this region the system is also stable, although w.r.t. the other equilibrium: the system will stay in the vicinity of said equilibrium, effectively maintaining its distance to the equilibrium of interest.

Finally, Fig. 2.7c aggregates the two former studies and exhibits the minimal expected average distance to either one the two equilibria x_l and x_r , effectively defining the shape of stability of the system. Again, this is consistent with the theoretical analysis of the system, which predicts a “bow tie” shape for the basin of attraction. We emphasize that these shapes are discrete: we may only have a finite precision on the border of the basin of attraction; decreasing the discretization step would yield a better precision of that shape, at the cost of increasing the computation time (which is proportional to the square of that discretization step since $\mathcal{I}$ is of dimension 2). ◀

PROBABILISTIC VERIFICATION OF DYNAMICAL SYSTEMS

In Chapter 2, models were described by a single ODE induced by a function f and a parameter λ ; there, the evolution is purely deterministic *a priori*, and the behaviour of the model cannot be modified during its execution; that is, it is not possible to interfere with the dynamics the model is subject to, nor is it possible to reset its position. In this chapter, we will describe and verify more complex models which allow such behaviour through external events, which we defined in Section 1.3 on page 22; these hybrid dynamical systems are based on ODEs to which a “reset mechanism” is added. To be specific, the evolution of these systems is twofold: on the one hand, they *flow*, i.e. they evolve continuously with time according to the dynamics they are subject to, through a parameterized ODE; on the other hand, they may *jump*, i.e. undergo a discrete and instantaneous transformation (according to some probability distribution) of either the ODE they are subject to (through a change of parameter λ) or their current value (through a reset of variable z).

Objective of the chapter

In this chapter, we develop a novel framework for abstracting hybrid dynamical systems as Markov processes¹. The resulting process is fully described according to the dynamics of the system, but does not enclose its history: its graph is finite, as it is constructed over a finite partition of the phase space. The abstraction of the dynamics of the original system by the means of a probability distribution function allows for any ODE to be considered, as long as the usual properties of study are met. In particular, we do not restrict the dynamics to the setting of linear ODEs, or ODEs defined on a decidable structure, and

¹We simplify the framework proposed in [11] to abstract dynamical systems as Markov processes, and enhance it by merging several abstractions (corresponding to several dynamics) together and incorporating a controller.

allow for the dynamics of the system to be changed during the execution of the model.

To give an intuition of this contribution, we briefly summarize the abstraction method detailed in this chapter; a schematics of it is given in Fig. 3.1. Consider a hybrid system $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ (see Definition 1.3.1 on page 27): the main objective is to study the behaviour of $\mathcal{S}$ through its abstraction as a Markov process, i.e. a probabilistic system. To this end, the phase space Ω of $\mathcal{S}$ is partitioned into a finite set $\mathfrak{Q}$ of regions, which will serve as the states of the Markov process; this constitutes a first layer of abstraction, as the exact position z on the system is abstracted as a region R in the partition. Then, for each possible value λ in the (finite) set Λ of parameters of $\mathcal{S}$, we compute the probability that a trajectory of duration τ (the discretization step of the timeline, see Eq. (1.20) on page 26) starting in a region R_i stops in region R_j , and use these values to define the transition probability distribution function of a Markov chain whose states are the regions of the partition. Finally, these Markov chains are aggregated so that the resulting Markov process can change both its position (i.e. the region in Ω) and the transition probability distribution function (identified to the parameter λ). The resulting probabilistic system is then studied to establish properties, which can then be transposed to the setting of the hybrid system.

Again, to ease the comprehension of the study process, we will unfold it on an example from the literature.

3.1 Running example: a stability study on epidemic spread [60]

Following the outbreak of the COVID-19 pandemic in January 2020, numerous studies [5, 56, 70] have been carried out to better understand the dynamics of the spread of the disease. These articles develop and tune mathematical models w.r.t. official health data to better understand the underlying dynamics of the disease. In [60], Ottaviano, Sensi, and Sottile study a so-called *SAIRS* model for the population, which has been developed by Robinson and Stilianakis [67]. In these two articles, the authors explore the stability of the model, that is the existence of equilibrium points in the phase space as well as their stability (similarly to what we assess in Section 2.2). Notably, they investigate several properties, such as the conditions of persistence of the disease, or of its disappearance.

While Robinson and Stilianakis [67] fitted the model to a real-life case of *influenza* epidemics, Ottaviano, Sensi, and Sottile [60] focused more on the theoretical properties

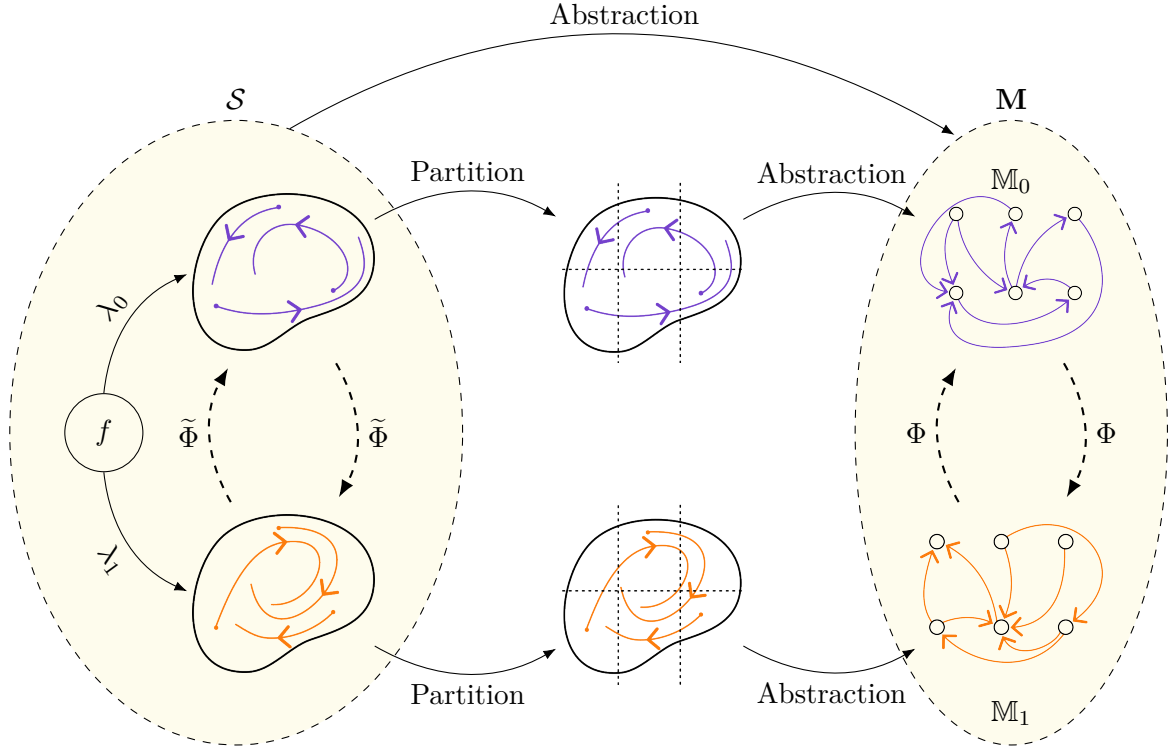


Figure 3.1: Schematics of the abstraction. $\mathcal{S}$ is a hybrid system, represented as a set of dynamics obtained through the combination of function f and parameter values λ_0, λ_1 . $\tilde{\Phi}$ is a switching process for $\mathcal{S}$, which may decide to reset parameter values in the ODE defined by f or the position of the system. Each dynamics induced by f and a parameter value λ is abstracted as a Markov chain; aggregating these Markov chains results in a Markov decision process $\mathbf{M}$ whose nondeterministic choices, represented by process Φ , allow to switch from one virtual Markov chain (i.e. one dynamics) to another.

of the SAIRS model, in particular its stability (as defined in Definition 2.2.3) depending on the value of a particular indicator. The proposed model is defined as follows:

$$\begin{cases} \frac{dS}{dt} = \mu - (\beta_A A(t) + \beta_I I(t)) S(t) - (\mu + \nu) S(t) + \gamma R(t), \\ \frac{dA}{dt} = (\beta_A A(t) + \beta_I I(t)) S(t) - (\alpha + \delta_A + \mu) A(t), \\ \frac{dI}{dt} = \alpha A(t) - (\delta_I + \mu) I(t), \\ \frac{dR}{dt} = \delta_A A(t) + \delta_I I(t) + \nu S(t) - (\gamma + \mu) R(t), \end{cases} \quad (3.1)$$

with initial condition $(S_0, A_0, I_0, R_0) \in \{(S, A, I, R) \in \mathbb{R}_+^4 \mid S + A + I + R = 1\}$. The dynamics of the model are illustrated as a flow diagram in Fig. 3.2.

The parameter of the differential model presented in Eq. (3.1) is denoted $\lambda = (\mu, \beta_A,$

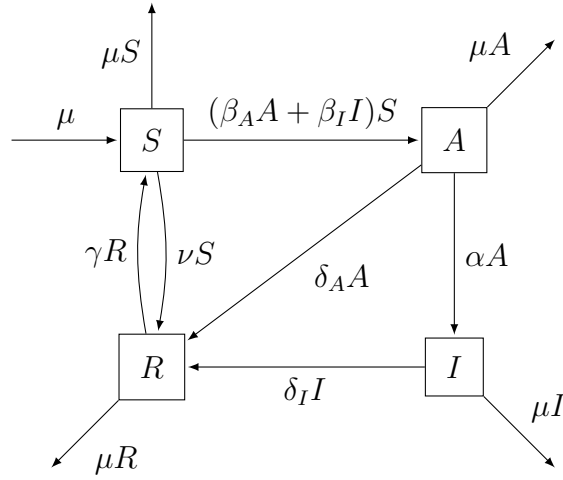


Figure 3.2: Flow diagram for the SAIRS model (3.1) (adapted from [60]).

$\beta_I, \delta_A, \delta_I, \alpha, \nu, \gamma$) with positive real coefficients whose meaning are synthesized in Table 3.1, alongside the values considered for the study.

Parameter	Description	Value(s)
μ	Mortality rate	$\frac{1}{(70 \times 365)}$
β_A	Transmission rate from A to S	$\{0.5, 0.9\}$
β_I	Transmission rate from I to S	$\{0.5, 0.9\}$
δ_A	Recovering rate from A	$\{0.1, 0.5\}$
δ_I	Recovering rate from I	$\{0.1, 0.5\}$
α	Progress rate from A to the symptomatic class I	0.15
ν	Vaccination proportion of the class S	0.01
γ	Loss of immunity rate of the class R	0.01

Table 3.1: Synthesis of the values of the parameter components $\mu, \beta_A, \beta_I, \delta_A, \delta_I, \alpha, \nu, \gamma$.

Since components $\gamma, \alpha, \nu, \gamma$ are fixed, we may lighten the notation and rewrite $\lambda = (\beta_A, \beta_I, \delta_A, \delta_I)$ if the context is sufficiently clear.

In Eq. (3.1), the values S, A, I, R represent fractions of individuals within the total population, thus $S + A + I + R = 1$ at all time. Therefore, the model can be reduced to

the following three-equation system:

$$\begin{cases} \frac{dS}{dt} = \mu - (\beta_A A(t) + \beta_I I(t)) S(t) - (\mu + \nu) S(t) + \gamma R(t), \\ \frac{dA}{dt} = (\beta_A A(t) + \beta_I I(t)) S(t) - (\alpha + \delta_A + \mu) A(t), \\ \frac{dI}{dt} = \alpha A(t) - (\delta_I + \mu) I(t), \end{cases} \quad (3.2)$$

for all $t \geq 0$, and R can be retrieved through $R = 1 - S - A - I$. For convenience, we introduce the notations $z = (S, A, I) \in \mathbb{R}^3$ and

$$\frac{dz}{dt}(t) = f(z, \lambda) \quad (3.3)$$

where $\frac{dz}{dt}$ denotes the derivative of z w.r.t. variable t , and

$$f(z, \lambda) = \begin{pmatrix} \mu - (\beta_A A + \beta_I I) S - (\mu + \nu) S + \gamma(1 - A - I) \\ (\beta_A A + \beta_I I) S - (\alpha + \delta_A + \mu) A \\ \alpha A - (\delta_I + \mu) I \end{pmatrix}. \quad (3.4)$$

Finally, we consider the compact

$$\Omega = \{(S, A, I) \in \mathbb{R}_+^3 \mid S + A + I \leq 1\}. \quad (3.5)$$

The following theorem, extracted from [60], ensures that Ω is positively invariant:

Theorem 3.1.1 (see [60], Theorem 1 p.5).

The region Ω defined by Eq. (3.5) is positively invariant for any $\lambda \in \mathbb{R}_+^4$, i.e. $z(t, z_0, \lambda) \in \Omega$ for any $t \geq 0$ if $z_0 \in \Omega$. $\triangleleft$

Intuitively, Theorem 3.1.1 ensures that any trajectory starting in Ω will remain in Ω , so that it can safely be used as the phase space of the system for the whole study. Then, using Chillingworth' Theorem (see Theorem 1.2.6 on page 19) and the compactness of Ω , we have the following corollary establishing the existence and uniqueness of solutions:

Corollary 3.1.2.

For all $z_0 \in \Omega$, the Cauchy problem defined by Eq. (3.3) and $z(0) = z_0$ admits a unique solution denoted $\mathbf{z}(t, z_0, \lambda)$, defined on $[0, +\infty)$, whose components are non-negative. $\triangleleft$

The proof of Corollary 3.1.2 is given in Section C.1, on page 132.

In [60], Ottaviano, Sensi, and Sottile investigate the (conditions for the) stability of the system. We give a brief summary in Section 3.1.

Summary of the study on stability

Compartmental models may have several equilibrium points that may or may not be stable. The existence and the stability of these equilibria depends on parameter λ , through a critical value called *basic reproduction number*, usually denoted r_0 . In the case of this model (see [60], Lemma 2 p.5), we have:

$$r_0 = \left(\beta_A + \frac{\alpha\beta_I}{\delta_I + \mu} \right) \frac{\gamma + \mu}{(\alpha + \delta_A + \mu)(\nu + \gamma + \mu)}. \quad (3.6)$$

For this model, two equilibria may exist depending on the value of r_0 : the *disease-free equilibrium* (where the disease disappears) DFE , and the *endemic equilibrium* (where the disease persists in the population) EE .

DFE always exists no matter the value of r_0 , and is given by

$$DFE = \left(\frac{\mu + \gamma}{\mu + \nu + \gamma}, 0, 0 \right). \quad (3.7)$$

On the other hand, EE exists only if $r_0 > 1$, and is given by

$$\begin{aligned} S &= \frac{(\alpha + \delta_A + \mu)(\delta_I + \mu)}{\beta_A(\delta_I + \mu) + \beta_I\alpha}, \\ A &= \frac{\delta_I + \mu}{\alpha} I^*, \\ I &= \frac{\alpha(\delta_I + \mu)(\mu + \nu + \gamma)(\alpha + \delta_A + \mu)(r_0 - 1)}{[\beta_A(\delta_I + \mu) + \beta_I\alpha][(\alpha + \delta_A + \mu + \gamma)(\delta_I + \mu) + \alpha\gamma]}. \end{aligned} \quad (3.8)$$

The following theorem, extracted from [60], establishes the global stability of the disease-free equilibrium DFE or the endemic equilibrium EE depending on the value of basic reproduction number r_0 :

Theorem 3.1.3 (see [60], Theorems 5 p.5, 11 p.10, 12 p.12, 14. p13).

If $r_0 < 1$, then the disease-free equilibrium DFE is globally asymptotically stable in Ω .

If $r_0 > 1$, then DFE is unstable, and the endemic equilibrium EE is globally asymptotically stable in the interior $\mathring{\Omega}$ of Ω if either

- $\gamma = 0$;
- $\beta_A = \beta_I$ and $\delta_A = \delta_I$;
- or $\beta_A < \delta_I$.

◁

Note that Conjecture 15 in [60] suggests that the conditions on λ could be removed for EE to be globally asymptotically stable if $r_0 > 1$.

In Fig. 3.3, we present two simulations of the model given by Eq. (3.1) with the following parameters: $\beta_A = 0.15$ (Fig. 3.3a) or $\beta_A = 0.5$ (Fig. 3.3b), $\beta_I = \beta_A$, $\delta_A = \delta_I = 0.1$. In the first simulation (Fig. 3.3a), we have $r_0 \simeq 0.75$, and we observe that the fractions of infected individuals $A(t)$ and $I(t)$ converge to 0, i.e. the system is attracted to the disease-free equilibrium DFE , in accordance with Theorem 3.1.3. In the second simulation (Fig. 3.3b), we have $r_0 \simeq 2.50$, and we observe that $A(t)$ and $I(t)$ converge to positive values, i.e. the epidemic persists and is asymptotically stable, in accordance with Theorem 3.1.3.

3.2 Abstraction procedure

3.2.1 Representation as a Hybrid Dynamical System

We now describe how a system such as Eq. (3.1) in Section 3.1 will be represented in our framework. We start by briefly recalling the definition of Hybrid Dynamical Systems, whose complete definition is available in Section 1.3.2, in particular on page 27:

Definition 1.3.1 (Hybrid Dynamical System). A *hybrid dynamical system* is a tuple $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ where

- $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ is a function defining an ODE as in Eq. (1.18);
- $\Lambda \subseteq \mathbb{R}^m$ is the parameter space of the system;
- $\Omega \subseteq \mathbb{R}^n$ is the phase space of the system, and is compact and positively invariant;
- $\mathcal{T}$ is a discretization of the timeline, as defined in Eq. (1.20);
- $\mathcal{D} = \mathcal{D}^\Lambda \cup \mathcal{D}^\Omega$ is a finite set of discrete distributions on Λ and on Ω ;

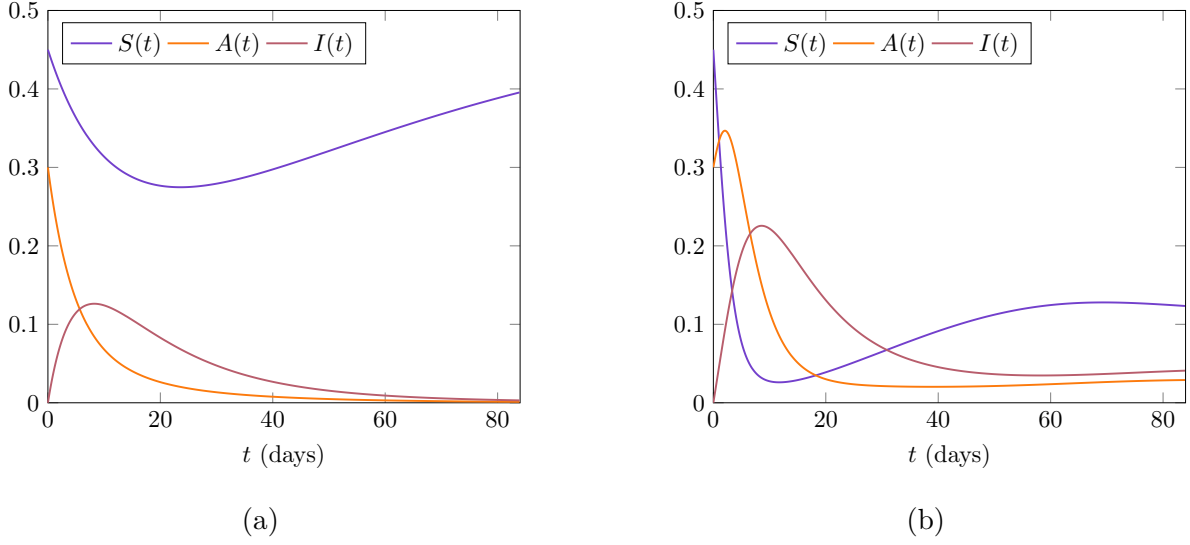


Figure 3.3: Trajectories of the system given by Eq. (3.2) with different parameter values: $\mu = \frac{1}{70 \times 365}$, $\beta_A = 0.15$ (Fig. 3.3a) or $\beta_A = 0.5$ (Fig. 3.3b), $\beta_I = \beta_A$, $\delta_A = \delta_I = 0.1$, $\alpha = 0.15$, $\nu = 0.01$, $\gamma = 0.01$. The purple curves correspond to the fraction $S(t)$ of susceptible individuals, with initial value $S_0 = 0.45$; the orange curves correspond to the fraction $A(t)$ of asymptomatic infected individuals, with initial value $A_0 = 0.3$; the red curves correspond to the fraction $I(t)$ of symptomatic infected individuals, with initial value $I_0 = 0$.

- $\mathcal{A} = \{\emptyset, \zeta_\tau\} \cup \{\zeta_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$ is the set of actions that can be performed by the system.

◁

For the remaining of the chapter we fix the hybrid system $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$. This hybrid system can be controlled by one or several processes to account for the environment of the concrete system, which pick actions that translate to discrete transitions in $\mathcal{S}$, through the realisation of a probability distribution (e.g. $\mathbb{D} \in \mathcal{D}$): this may change the dynamics or the position of the system. These processes, which we identify as controllers (see Eq. (1.24) on page 30), can be of two kinds: i) an artificial change enforced by the modeller of the system, which we denote by $\mathbb{X}$, or ii) a natural change enforced by the environment of the system, which we denote by $\mathbb{Y}$. At each time point $t_k \in \mathcal{T}$, these two processes can pick one action for the hybrid system to perform, which is then translated to a transition $\rightarrow$ (see Eq. (1.23) on page 29). For readability, the i th transition in a behaviour, if it is induced by controller $\mathbb{X}$ or $\mathbb{Y}$, is denoted $\xrightarrow{\mathbb{X}_i}$ or $\xrightarrow{\mathbb{Y}_i}$, respectively; note

that this does not necessarily mean that $\xrightarrow{\mathbb{X}_i}$ or $\xrightarrow{\mathbb{Y}_i}$ occurs at time step $t_i \in \mathcal{T}$, since a transition in $\mathcal{S}$ does not necessarily mean that time has progressed.

Remark 3.2.1. Note that the framework of controllers does not restrict the choices of the controllers *a priori*. Of course, naive approaches to reachability properties (see Definition 1.1.4 on page 14) could simply force resets through discrete transitions, in order to force the system to “move” to the desired state. The controller could also forbid the system to evolve according to its dynamics by only choosing actions that induce discrete transitions (see Eq. (1.23) on page 29), essentially freezing time.

While this is valid from a theoretic point of view, it cannot be reflected into a meaningful scenario for the concrete system; in this work, we choose to constrain when these controllers operate² by enforcing a rotation between them and continuous transitions. ◀

We want to ensure a rotation between the discrete evolution enforced by the controllers and the temporal evolution of the system; to this end, we allow $\mathbb{X}, \mathbb{Y}$ to act only once at each time step $t_k \in \mathcal{T}$, and in that order. This is done by allowing $\mathbb{X}$ to pick an action only if the length of the behaviour π is such that $|\pi| \equiv 1 \pmod{3}$; similarly, $\mathbb{Y}$ is allowed to pick an action only if the length of the behaviour π is such that $|\pi| \equiv 2 \pmod{3}$. If the length of the behaviour is such that $|\pi| \equiv 0 \pmod{3}$, then a continuous transition is performed by the hybrid system. It translates to $\mathbb{X}$ and $\mathbb{Y}$ being able to act once every three transitions each: all behaviours will be of the form

$$\pi = s_0 \xrightarrow{\tau} s_1 \xrightarrow{\mathbb{X}} s_2 \xrightarrow{\mathbb{Y}} s_3 \xrightarrow{\tau} s_4 \xrightarrow{\mathbb{X}} s_5 \xrightarrow{\mathbb{Y}} s_6 \dots \quad (3.9)$$

Note that each of the two controllers can pick the $\emptyset$ action to “pass” their turn. Here, we consider $\mathbb{Y}$ never picks it, so that only $\mathbb{X}$ can wait for its next turn.

The trajectory $\tilde{\mathbf{z}}$ of the hybrid system will be composed of curved lines (corresponding to the continuous evolutions) and discrete points (corresponding the resets induced by the controllers):

- at timepoint $t_k \in \mathcal{T}$, the system is in state (z_i, λ_i) , i.e. it already performed $i = 3k$ transitions; between two successive time points $t_k, t_{k+1} \in \mathcal{T}$, it evolves according to the solution $\mathbf{z}(\cdot, z_i, \lambda_i)$ to the ODE induced by the state (z_i, λ_i) .
- once τ time units have elapsed, the system has performed an $(i + 1)$ th transition

²But one could also constrain *how* they do (e.g. “for every 3 consecutive actions, at least one is a_τ ”).

and is in state (z_{i+1}, λ_{i+1}) ; we have $z_{i+1} = \lim_{t \rightarrow \tau} \mathbf{z}(\tau, z_i, \lambda_i)$, and $\lambda_{i+1} = \lambda_i$;

- then, the controller $\mathbb{X}$ is allowed to pick an action, which is translated to a discrete transition $(z_{i+1}, \lambda_{i+1}) \xrightarrow{\mathbb{D}} (z_{i+2}, \lambda_{i+2})$ for some distribution $\mathbb{D} \in \mathcal{D}$ (if the action is not $\emptyset$) or distribution $\mathbb{D} \in \mathcal{W}$ (if the action is $\emptyset$);
- finally, the controller $\mathbb{Y}$ is allowed to pick an action related to some fixed distribution function $\mathbb{V}$, which is translated to the discrete transition $(z_{i+2}, \lambda_{i+2}) \xrightarrow{\mathbb{V}} (z_{i+3}, \lambda_{i+3})$, such that $z_{i+2} = z_{i+3}$;

As such, the trajectory $\tilde{\mathbf{z}}$ will be discontinuous at each time point where the reset induced by $\mathbb{X}$ changes the position z of the system; it will be right-continuous but not differentiable at each time point where a reset by either $\mathbb{X}$ or $\mathbb{Y}$ changes the dynamics (identified to the value of the parameter Λ) of the system.

A depiction of such trajectory is given in Fig. 3.4.

Mathematically, given two controllers $\mathbb{X}$ and $\mathbb{Y}$, a hybrid dynamical system with two controllers can be seen as the following system $\mathbb{H}$:

$$\mathbb{H} = \begin{cases} \tilde{\mathbf{z}}(t) &= \mathbf{z}(t - k\tau, z_i, \lambda_i), & i = 3k, & k \in \mathbb{N}, & k\tau \leq t < (k+1)\tau, \\ z_{i+1} &= \lim_{t \rightarrow \tau} \mathbf{z}(t, z_i, \lambda_i), & i = 3k, & k \in \mathbb{N}, \\ \lambda_{i+1} &= \lambda_i, & i = 3k, & k \in \mathbb{N}, \\ \mathbb{X}_i &: \pi \xrightarrow{\mathbb{D}} (z_{i+1}, \lambda_{i+1}), & i \equiv 1 \pmod{3}, & \mathbb{D} \in \mathcal{D} \cup \mathcal{W}, \\ \mathbb{Y}_i &: \pi \xrightarrow{\mathbb{V}} (z_i, \lambda_{i+1}), & i \equiv 2 \pmod{3}. \end{cases} \quad (3.10)$$

Both $\mathbb{Y}$ and $\mathbb{X}$ are deterministic controllers for the system, as described in Definition 1.3.5 on page 30: we consider they are deterministic in their choices of action, but these actions induce discrete transitions that are probabilistic in their realisation. $\mathbb{Y}$ will be able to pick a fixed probability distribution function $\mathbb{V}$ corresponding to the natural events that can influence the system, and $\mathbb{X}$ will be able to pick from the set $\mathcal{D}$ of distributions in $\mathcal{S}$. As we will see in the following, these two types of discrete evolutions will be abstracted in two different manners.

For the remaining of the chapter, the global solutions to Eq. (3.10) are called behaviours. These behaviours are probabilistic sequences of interrupted global solutions to the differential model defined in Eq. (1.18) and instanciated with the successive values for z_k and λ_k .

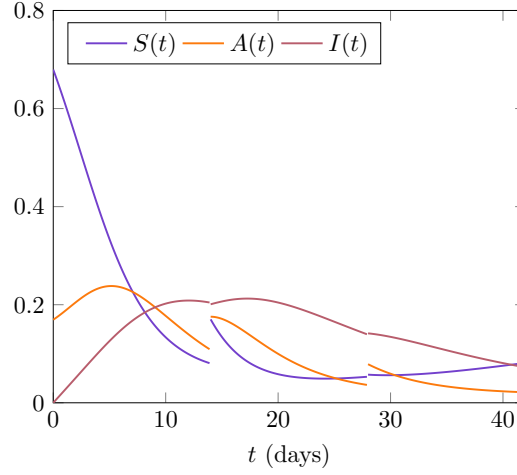


Figure 3.4: An example of trajectory for a Hybrid Dynamical System, based on the model described in Section 3.1. The curves correspond to continuous transitions. Colors have the same meaning as Fig. 3.3.

Example 3.2.2 (Running example). For this example, we consider that two general types of discrete changes are at stake: i) the enforced changes due to public health policies, e.g. lockdown, vaccination campaigns, use of masks. . . ii) the natural changes due to the environment of the disease, e.g. the appearance and spread of a new variant within the population.

These changes will be abstracted as controllers, which can act concurrently at any time step $k\tau$. For the sake of realism, we consider that the enforced process happens before the natural one, so that no controlled change can be conducted instantaneously after a natural change. We will see later on how we enforce such a behaviour.

Enforced evolution of the disease

In this first situation, we consider a modification of the dynamics of the disease (which we model by a change of the value of the parameter λ through a reset of δ_A, δ_I), or a modification of the organization of the population (which we model by a change of the position z of the system, i.e. a change in the value of S, A or I), through the different actions available. This control process is both nondeterministic and probabilistic, as the decision to change the dynamics results from a nondeterministic choice but the outcome of this change follows a (chosen) probability distribution over the possible values; non-determinism is resolved by a controller (see Definition 1.3.5 on page 30), which makes

decisions (i.e. choose actions that induce discrete transitions, see Eq. (1.23) on page 29) at specific time points, based on the history of the system. These choices can be guided by the observation of the epidemic, e.g. apply a change only if the basic reproduction number r_0 is greater than 1, and can take into account some indicators external to the disease itself, e.g. the cost of the various available measures. If the chosen action is $\zeta_{\mathbb{D}}$ for some distribution $\mathbb{D} \in \mathcal{D}$, then the realisation of $\mathbb{D}$ modifies either the recovering rates δ_A, δ_I or the position z of the system; if the chosen action is $\emptyset$, then $\mathcal{S}$ “waits” by undergoing a transition according to $\mathbf{1}_{\lambda_k} \in \mathcal{W}$, where λ_k is the current value of parameter λ .

This whole control process is denoted $\mathbb{X}$, such that given a history $\pi = s_0 \dots s_k$ of the system (see Eq. (1.25) on page 30) we have $\mathbb{X} : \pi \rightarrow a$, where $a = \emptyset$ or $a = \zeta_{\mathbb{D}}$ for some distribution $\mathbb{D} \in \mathcal{D}$.

Designing a controller $\mathbb{X}$ that ensures the satisfaction of some property φ w.r.t. $\mathcal{S}$ is the end goal of this chapter, so we do not yet give its definition, as it does not differ from the definition of a controller (see Definition 1.3.5 on page 30); its description will be given later on in Section 3.2.4.

Natural evolution of the disease

In this second situation, we consider a modification of the dynamics of the disease by its natural environment, which we model by a change in the values of β_A, β_I . This process is considered purely probabilistic, as it does not seem realistic to consider that natural changes would be dictated by an actual entity. It may act once at each timestep $t_k = k\tau$ of the execution of the system, and it is twofold:

1. a variable $\mathbb{B}$ following a Bernoulli distribution of parameter $b \in (0, 1)$ is sampled;
2. depending on the value of $\mathbb{B}$:
 - if $\mathbb{B} = 0$, then nothing happens and the values β_A, β_I are not modified;
 - if $\mathbb{B} = 1$, a second random variable $\mathbb{V}$ is sampled and the values β_A, β_I are updated according to its probability distribution.

We now identify the variables to their respective distributions.

The distribution $\mathbb{V}$ corresponds to several distributions in the hybrid system: since $\mathbb{V}$ only modifies β_A, β_I , there exists one distribution $\mathbb{V}_\lambda$ for each possible value of $\lambda \in \Lambda$, simply defined as $\mathbb{V}_\lambda(\lambda') = \mathbb{V}(\lambda, \lambda')$.

Unlike other actions $a \in \mathcal{A}$, the actions picked by $\mathbb{Y}$ may induce two different transitions in $\mathcal{S}$, based on the realisation of $\mathbb{B}$: if $\mathbb{B} = 0$, then the state of the system is not changed, and $\mathcal{S}$ “waits” (similarly as when $\mathbb{X}$ chooses $\emptyset$) by performing a self-loop through the Dirac distribution $\mathbf{1}_\lambda$ concentrated on the current parameter value λ ; if $\mathbb{B} = 1$, a discrete transition is induced by the distribution $\mathbb{V}_\lambda$.

It is possible to merge $\mathbb{B}$ and $\mathbb{V}_\lambda$ as a single probability distribution; formally, given a behaviour $\pi = s_0 \dots s_k$ such that $s_k = (z_k, \lambda_k)$, the probability distribution $\tilde{\mathbb{V}}_{\lambda_k}$ associated to the transition enforced by $\mathbb{Y}$ is defined as follows:

$$\begin{aligned} \tilde{\mathbb{V}}_{\lambda_k} : \Omega \times \Lambda &\rightarrow [0, 1] \\ (z, \lambda) &\mapsto \begin{cases} b\mathbb{V}(\lambda_k, \lambda) + (1 - b)\mathbf{1}_{\lambda_k}(\lambda), & \text{if } z = z_k, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.11)$$

These distributions form the set $\mathcal{V} = \{\tilde{\mathbb{V}}_\lambda \mid \lambda \in \Lambda\}$; at this point, we augment the hybrid system $\mathcal{S}$ with new corresponding actions by appending $\mathcal{A}$ with the set $\mathcal{A}_\mathcal{V} = \{\zeta_{\tilde{\mathbb{V}}_\lambda} \mid \lambda \in \Lambda\}$. To precisely mimic the distribution $\mathbb{V}$, we consider that $\zeta_{\tilde{\mathbb{V}}_\lambda}$ is enabled for state $s \in (\Omega \times \Lambda)$ only if $s = (z, \lambda)$ for some $z \in \Omega$.

The whole process is denoted $\mathbb{Y}$ and associates the action $\zeta_{\tilde{\mathbb{V}}_{\lambda_k}}$ to all input behaviours $\pi = s_0 \dots s_k$, where $s_k = (z_k, \lambda_k)$, and the action $\zeta_{\tilde{\mathbb{V}}_{\lambda_k}}$ induces a discrete transition according to distribution $\tilde{\mathbb{V}}_{\lambda_k}$; it is defined as:

$$\begin{aligned} \mathbb{Y} : (\Omega \times \Lambda)^+ &\rightarrow \mathbf{A} \\ \pi &\mapsto \zeta_{\tilde{\mathbb{V}}_{\lambda_k}}. \end{aligned} \quad (3.12)$$

Joint effect of $\mathbb{X}$ and $\mathbb{Y}$

In this chapter, these two controllers $\mathbb{X}$ and $\mathbb{Y}$ will be executed one after the other, starting with $\mathbb{X}$, and each pair of choices will be preceded by a continuous evolution in $\mathcal{S}$. Intuitively, if we abuse notations and denote by $\xrightarrow{\mathbb{X}}$ and $\xrightarrow{\mathbb{Y}}$ the transitions induced by the actions picked by controllers $\mathbb{X}$ and $\mathbb{Y}$, a run in this setting will look like this:

$$\pi = s_0 \xrightarrow{\tau} s_1 \xrightarrow{\mathbb{X}} s_2 \xrightarrow{\mathbb{Y}} s_3 \xrightarrow{\tau} s_4 \dots \quad (3.13)$$

To finish this example, we focus on the natural process $\mathbb{Y}$, and consider it alone in the control of $\mathcal{S}$. Recall that given a discretization $\mathcal{T} = \{t_0 < t_1 < \dots\}$ of the timeline, $\mathbb{Y}$ may choose one action at each timepoint $t_k \in \mathcal{T} \setminus \{t_0\}$, which is then translated to a

discrete transition in $\mathcal{S}$.

In Fig. 3.5, we provide two sets of trajectories for the different fractions $S(t)$, $A(t)$, $I(t)$ of the population; the parameter set is the same as in Fig. 3.3, except for β_A, β_I which are assumed equal and randomly chosen, at each timepoint in the discretization of the timeline, in the set $\{n \times 0.05 \mid 1 \leq n \leq 20\}$. This corresponds to a scenario where the dynamics are modified at each time point $k\tau$ by the environment, i.e. the parameter b of the Bernoulli distribution $\mathbb{B}$ is equal to 1, and $\mathbb{V}$ is a discrete distribution function on $\Lambda \times ([0.05, 1]^2 \times (0.1, 0.1))$, such that for any $\lambda \in \Lambda$, $\mathbb{V}(\lambda, \cdot)$ is a uniform distribution on $([0.05, 1]^2 \times (0.1, 0.1))$ with discretization step 0.05. In Fig. 3.5a, $\tau = 7$ days; in Fig. 3.5b, $\tau = 21$ days. As expected, a coarser time discretization step τ leads to less resets and a smoother behaviour, as seen in Fig. 3.5b.

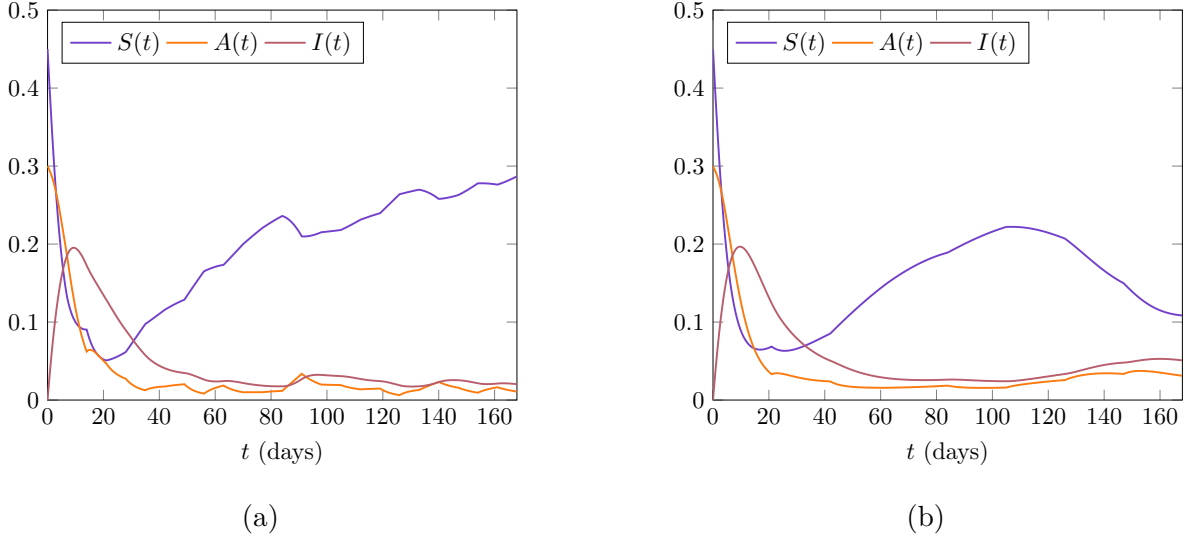


Figure 3.5: Two trajectories for the system with different values for τ . In Fig. 3.5a, $\tau = 7$; in Fig. 3.5b, $\tau = 21$. Colors have the same meaning as in Fig. 3.3.

◀

In this section, we present the framework to abstract a hybrid dynamical system $\mathcal{S}$ as a Markov Decision Process. This framework can be divided into three main steps:

1. in Section 3.2.2, we partition the phase space of the system, then abstract each possible dynamics (corresponding to each parameter λ) of the hybrid dynamical system as a Markov chain;

2. in Section 3.2.3, we introduce a first layer of abstraction, corresponding to the natural evolution process $\mathbb{Y}$ of the system, and produce a Markov decision process encompassing every dynamics, which are linked according to the behaviour of $\mathbb{Y}$ through a fixed controller with 2 actions (see Definition 1.4.4 on page 34);
3. in Section 3.2.4, we add a second layer of abstraction, corresponding to an enforced evolution process $\mathbb{X}$ for the system, and produce a Markov decision process for which a controller (see Definition 1.4.4) can be designed to switch between dynamics or position, according to the different distributions available in $\mathcal{S}$.

3.2.2 Abstraction of the dynamics as a Markov chain

Partition of the phase space

We now unfold the first step of our framework, that is the partitioning of the phase space Ω . Recall that the resulting partition $\mathfrak{Q}$ will be used as states of the various Markov processes we will consider later on.

We start by considering only the phase space Ω of the system, which we will partition into a finite set of region. Then, considering the time discretization $\mathcal{T}$ defined in Eq. (1.20) (see page 26), we will evaluate the ratio of trajectories of duration τ that start in region $R_i \in \mathfrak{Q}$ and stops in region $R_j \in \mathfrak{Q}$; these ratios are interpreted as probabilities, and identified as a probability distribution function over the set $\mathfrak{Q}$, in order to build a Markov chain. Note that this probabilities are not natural, as the continuous evolution of a hybrid dynamical system is deterministic; it results from the abstraction of the exact position z of the system as the region which contains it within Ω : since this information is lost, the exact position of the system cannot be tracked and one needs to work at the region-level, in which case the transitions are not deterministic.

The partition can take any form as long as all regions have a non-zero measure, do not show particularities (such as cracks), and the possible reset values $z^* \in \mathcal{Z}^*$ relative to the distributions in $\mathcal{D}^\Omega$ are not on the border of the regions; the reason for the first requirement is given after Eq. (3.15) on page 88, and the reason for the second and third requirements is given after Lemma 3.2.9 on page 90 and discussed in Section C.3 on page 133.

It should be noted that this partition is tightly related to the property that needs to be verified, as well as the dynamics of the system: a system that evolves quickly w.r.t. t

requires a more granular partition than a system whose dynamics is slower, in order to prevent binary behaviour as explained in Example 3.2.3.

Example 3.2.3. Consider a value $\lambda = (0.5, 0.9, 0.1, 0.1)$ and a two region-partition $\Omega = \{R_{df}, R_e\}$ for the system given in Fig. 3.3a, with $R_{df} = \{(S, A, I) \in \mathbb{R}^3 \mid A + I \leq 0.05\}$ and $R_e = \Omega \setminus R_{df}$: R_{df} contains only the disease-free equilibrium DFE and is considered to be *almost* releaved from the disease, while R_e contains the endemic equilibrium EE . This value of λ is such that the endemic equilibrium EE (see Eq. (3.8) on page 76) exists and is globally asymptotically stable: if let alone, the system will be attracted by EE no matter its starting point.

Depending on the value of τ , different Markov chains can be built. Three of them are depicted in Fig. 3.6.

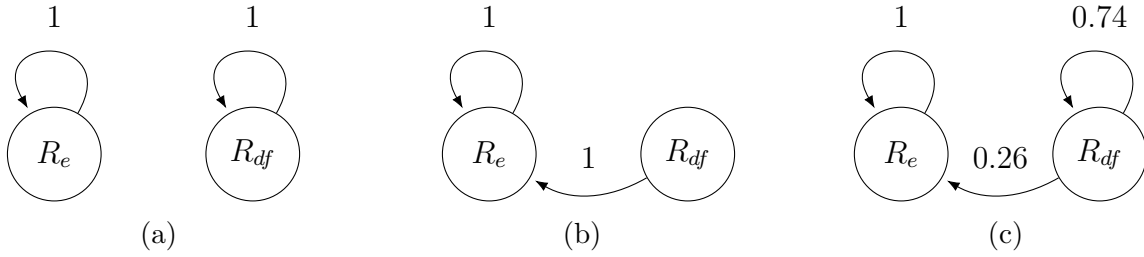


Figure 3.6: Three Markov chain abstractions of a system and a two-region partition, but with different time discretization steps τ ; all probabilities are identified to the ratio of trajectories of duration τ starting in one region and stopping in another, as explained in Eq. (3.15) (see page 88), and are computed through SMC techniques. Figure 3.6a has been constructed with too small a value $\tau = 0.25$ day compared to the velocity of the system, hence it is impossible to leave R_{df} in the Markov chain (note that in reality the probability of switching from R_{df} to R_e is not exactly 0, but it is extremely small); Figure 3.6b has been constructed with too big a value $\tau = 60$ days compared to the velocity of the system, hence the system necessarily transitions towards R_e in one step and we lack behaviour granularity; Figure 3.6c has been constructed with a more adequate value $\tau = 7$ days and we have all possible behaviours for the system in R_{df} , i.e. self-loops and transition to another region.

As we can see in Fig. 3.6a, a small value for τ (w.r.t. the speed of evolution of the system) leads to a disconnected graph with no possibility for the system to switch from one state to the other: the trajectories are so small w.r.t. the speed of evolution that it is almost improbable to switch from one region to another as the initial condition requires to be very close to their interface; since the SMC technique used to compute the probabilities is based on a uniform sampling of the initial region, sampling an initial condition close

enough to said interface is very unlikely, and it did not happen when performing our simulations (note that it could happen if we were to redo the simulations). On the contrary, a large value for τ leads to a completely deterministic evolution, as we can see in Fig. 3.6b where the system will stay in R_{df} after one transition. The abstraction process requires to find a middle ground between these two extreme behaviours, so that a system similar to Fig. 3.6c can be built: the value $\tau = 7$ days is satisfactory in that regard. ◀

Example 3.2.4 (Running example). Here, we focus on a reachability property ensuring the disappearance of the disease in a finite time. Therefore, we set a time bound T_{max} and are interested in the the property $\varphi =$ “the region around DFE is always reachable before T_{max} units of time have elapsed.” This naturally leads to a two-region partition: a region R_{df} containing DFE (recall that DFE is unique and characterized by μ, ν and γ which are not affected by the changes of dynamics) but no equilibrium EE , and another region R_e which complements R_{df} in Ω . Since we have a finite amount of dynamics, this partition can be constructed manually by finding the minimal distance ρ^* between DFE and EE , and setting a tolerance $\rho < \rho^*$ to define $R_{df} = \{(S, A, I) \in \Omega \mid A + I \leq \rho\}$, i.e. R_{df} is the closed region of Ω where at most a fraction ρ of the population is infected (either with or without symptoms); the resulting partition is denoted

$$\Omega = \{R_{df}, R_e\}. \quad (3.14)$$

Note that for the sake of simplicity, this partition is uniform w.r.t. λ . More complex partitions (e.g. λ -dependent partition) could be considered. ◀

Abstraction of the dynamics as a Markov chain

The second step of the abstraction procedure is to abstract each potential dynamics as a Markov chain. A *discrete time Markov chain* (MC for short, see [9], Section 10.1 p.747) is a transition system in which transitions are probabilistic. A more detailed description is given in Section 1.4.1, on page 31.

The goal here is to construct a transition probability distribution over the regions of the partition Ω . We assume a fixed value of λ for the remainder of the section, which we identify as the dynamics which the system is subject to during a continuous transition. Consider the time step τ defined in Eq. (1.20) to discretize the timeline. For each pair (R_i, R_j) of regions in Ω , we are interested in the ratio $p_{i,j}^{\tau,\lambda}$ of initial conditions z_0 in R_i

whose forward trajectory $\mathbf{z}(\cdot, z_0, \lambda)$ of duration τ stops in R_j . To describe it formally, consider a measure $m(\cdot)$ on Ω ³; we define these ratios in the following manner:

$$p_{i,j}^{\tau,\lambda} = \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\})}{m(R_i)}. \quad (3.15)$$

Note that Eq. (3.15) requires each region $R_i \in \mathfrak{Q}$ to have a non-zero measure, which is assumed here. Note also that since the initial ODE defined in Eq. (1.18) is autonomous, the ratio $p_{i,j}^{\tau,\lambda}$ does not depend on the starting time point of the trajectories. As expected, these ratios define probability distribution, and will be used as transition probabilities within the Markov process:

Lemma 3.2.5.

Let $m(\cdot)$ be a measure on phase space Ω . Let $\mathfrak{Q}$ be a partition of Ω , such that all region $R_i \in \mathfrak{Q}$ have a non-zero measure $m(R_i)$. For any $R_i, R_j \in \mathfrak{Q}$, we consider the value

$$p_{i,j}^{\tau,\lambda} = \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\})}{m(R_i)}. \quad (3.16)$$

For each region $R_i \in \mathfrak{Q}$, the values $p_{i,j}^{\tau,\lambda}$ define a probability distribution over the regions R_j of $\mathfrak{Q}$. ◁

The proof of Lemma 3.2.5 is given in Section C.2, on page 132.

We aggregate the values $p_{i,j}^{\tau,\lambda}$ as the function $\mathbb{P}_\lambda^\tau : \mathfrak{Q} \times \mathfrak{Q} \rightarrow [0, 1]$ defined by $\mathbb{P}_\lambda^\tau(R_i, R_j) = p_{i,j}^{\tau,\lambda}$; it follows that we have constructed a Markov chain:

Proposition 3.2.6.

Given a probability distribution $\mathbb{I}$ over $\mathfrak{Q}$, the tuple $\mathbb{M}_\lambda^\tau = (\mathfrak{Q}, \mathbb{P}_\lambda^\tau, \mathbb{I})$ is a Markov chain. ◁

Example 3.2.7 (Running example). Let $\lambda \in \Lambda$ be a parameter for the model, then let $\mathbb{S}_\lambda$ be the model induced by the ODE defined in Eq. (3.2) (on page 75) and λ .

Following Example 3.2.4 on page 87, we assume a value $\rho = 0.05$ that fits the requirements for the separation of the equilibria; then, we fix $R_{df} = \{(S, A, I) \in [0, 1] \mid A + I \leq 0.05\}$ and $R_e = \{(S, A, I) \in [0, 1]^3 \mid A + I > 0.05\}$. Let $z_0 \in R_e$ be an initial condition for the system, then let $\mathbb{I}$ be the Dirac distribution concentrated on R_e . The abstraction of $\mathbb{S}_\lambda$ on a symbolic level as a *parametric* Markov chain, whose transition probabilities are

³In our setting, $m(\cdot)$ is the Lebesgue measure on Ω but any other measure could be considered.

not yet expressed as numbers, is depicted in Fig. 3.7; concrete values for $p_{df,df}^\lambda, p_{df,e}^\lambda, p_{e,df}^\lambda$ and $p_{e,e}^\lambda$ can be computed using different methods depending on the context.

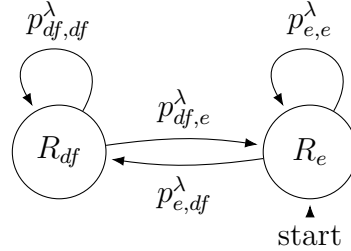


Figure 3.7: A parametric abstraction of the system defined by Eq. (3.2) and an initial condition $z_0 \in R_e$. The initial distribution is a Dirac distribution concentrated on R_e , as shown by the “start” label below the right node.

Here, we used SMC techniques similar to those used in Section 2.2; when instantiating λ to e.g. $(0.9, 0.9, 0.5, 0.5)$ we obtain Fig. 3.8.

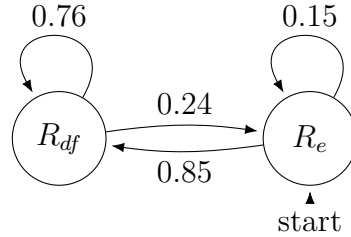


Figure 3.8: A concrete MC obtained by abstracting $\mathbb{S}_\lambda$ for $\lambda = (0.9, 0.9, 0.5, 0.5)$.

◀

In order to prove that the produced Markov chain is faithful to the dynamics of the differential system, i.e. that the behaviours of the differential system are indeed abstracted as behaviours of the Markov chain, we adapt the usual notion of simulation (see [73], Definition 4.7 p.34) to our setting. Intuitively, a simulation relation between two systems A and B establishes a relation between their states, such that any transition found in A can be emulated in B . Using the notion of transition in a hybrid dynamical system (defined in Eq. (1.22) on page 29), we can define the t -simulation of a differential system by another transition system of any kind, and then use this notion to establish a link of *almost- τ -simulation* of a differential system by a Markov chain:

Definition 3.2.8 ((almost-) t -simulation of a differential system). Let f be a function defining a parameterized ODE, then let λ be a parameter for f and Ω_λ be the phase space induced by f and λ . Let $t > 0$ be a fixed time duration. We say that a transition system B with state set Q_B t -simulates differential system $\mathbb{S}_\lambda = (f, \lambda, \Omega_\lambda)$ through a relation $\mathcal{R} \subseteq \Omega_\lambda \times Q_B$ iff:

1. for any initial state $z_0 \in \Omega_\lambda$, there exists an initial state $q_0 \in Q_B$ such that $(z_0, q_0) \in \mathcal{R}$;
2. for any pair $(z, q) \in \mathcal{R}$ and any t -successor z' of z , there exists a successor q' of q such that $(z', q') \in \mathcal{R}$.

We call **almost- t -simulation** a relation $\mathcal{R} \subseteq \Omega_\lambda \times Q_B$ that satisfies item 2. almost everywhere, i.e. such that the set $\check{\mathcal{R}}$ of pairs $(z, q) \in \mathcal{R}$ such that a transition $z \rightarrow z'$ in $\mathbb{S}_\lambda$ cannot be matched by a transition $q \rightarrow q'$ in B is such that $m(\check{\mathcal{R}}) = 0$. If there exists an almost- t -simulation relation between $\mathbb{S}_\lambda$ and B , we say that B **almost- t -simulates** $\mathbb{S}_\lambda$. $\triangleleft$

Using Definition 3.2.8, we establish a link of almost- τ -simulation of a differential system by its abstraction as a Markov chain, as presented in this section; the following lemma constitutes the first result of this chapter:

Lemma 3.2.9 (Almost- τ -simulation).

Let $\mathbb{S}_\lambda = (f, \lambda, \Omega_\lambda)$ be a differential system, then let z_0 be an initial state for $\mathbb{S}_\lambda$. Let $\tau > 0$ be a time step. Consider the following initial distribution over the states of $\mathfrak{Q}$:

$$\mathbb{I}(R) = \begin{cases} 1, & \text{if } z_0 \in R, \\ 0, & \text{otherwise.} \end{cases} \quad (3.17)$$

The MC $\mathbb{M}_\lambda^\tau = (\mathfrak{Q}, \mathbb{P}_\lambda^\tau, \mathbb{I})$ almost- τ -simulates the behaviour of differential system $\mathbb{S}_\lambda$. $\triangleleft$

The proof of Lemma 3.2.9 is given in Section C.3, on page 133. The impossibility to simulate all fixed-time behaviours of $\mathbb{S}_\lambda$ by its abstraction as a Markov chain $\mathbb{M}_\lambda^\tau$ comes from the difficulties encountered when trajectories stop on the border of a region. Indeed, since the probabilities in $\mathbb{M}_\lambda^\tau$ are identified as a ratio of measures (see Eq. (3.16) on page 88), it can happen that a point z lying on the border of R_i induces a trajectory that stops on the border of a region R_j : in that case, continuity of trajectories w.r.t. initial

condition cannot be leveraged to ensure that a neighborhood of z can be found, such that any initial condition in that neighborhood induces a trajectory that stops in R_j ; it results that we did not manage to prove, in that case, that the measure of the set of accepting initial condition is non-zero, i.e. that the probability $p_{i,j}^{\lambda,\tau}$ of switching from R_i to R_j is non-zero in $\mathbb{S}_\lambda$, and that there exists a transition from R_i to R_j in $\mathbb{M}_\lambda^\tau$. It is discussed in Section C.3 and only concerns a set of measure 0, hence the notion of *almost- τ -simulation*.

In the following, since the time duration τ is fixed for the whole study, we lighten the notation by dropping the “ τ ” superscript and we write $p_{i,j}^\lambda$, $\mathbb{P}_\lambda$ and $\mathbb{M}_\lambda$. A direct consequence of Lemma 3.2.9 is Corollary 3.2.10.

Corollary 3.2.10 (Reachability).

Let $k \in \mathbb{N}$ be a positive integer. Let $R \in \mathfrak{Q}$ be a target region. If $\mathbb{M}_\lambda$ can reach R within k transitions with probability 1, then the abstracted system $\mathbb{S}_\lambda$ will reach R within $k\tau$ time units with probability 1. $\triangleleft$

The proof of Corollary 3.2.10 is given in Section C.4 on page 135.

3.2.3 Aggregation of the Markov chains

Each Markov chain constructed in Section 3.2.2 is related to a single dynamics induced by a given parameter value λ , without any switch. Here, we introduce a first layer of abstraction corresponding to the natural changes in environmental factors through the controller $\mathbb{Y}$ (see Section 3.2.1 on page 77). To this end, we aggregate the Markov chains produced in Section 3.2.2 as a bigger system $\mathbf{M}$, and define a new probability distribution function $\mathbf{P}$ that can be parameterized by the means of actions, thus creating a Markov Decision Process (MDP, see Definition 1.4.3 on page 32). This MDP will be able to mimic both the continuous and the discrete (induced by natural process $\mathbb{Y}$) transitions of hybrid system $\mathcal{S}$ through a controller Ψ (see Definition 1.4.4 on page 34): at each point in a run of the MDP, Ψ will choose actions based on the past behaviour π of $\mathbf{M}$, which will translate to transition probability distribution functions among the states of $\mathbf{M}$ (see Eq. (1.29) on page 33). While the actions chosen by this controller could, in general, affect both the position z and the parameter λ of the system, here we consider they may only change the value of λ as explained in Section 3.2; in Section 3.2.4, we will see a change of both position and parameter can be adapted to the setting of Markov processes.

The goal in this section is to aggregate the Markov chains produced earlier as a larger

MDP, encapsulating both the continuous dynamics related to each parameter λ in $\mathcal{S}$ and the discrete jumps between them, caused by the controller $\mathbb{Y}$; it can be leveraged to verify reachability properties such as property φ in Example 3.2.4, in the case where only a natural control is applied.

This will be encoded through actions, of which we consider three types:

- an action $\emptyset$ which forces $\mathbf{M}$ to transition to its current state, i.e. perform a self-loop;
- an action a_τ that lets the MDP evolve according to the distribution $\mathbb{P}_\lambda$ induced by the value of parameter λ within the hybrid system $\mathcal{S}$, as defined in Eq. (3.15);
- an action $a_\mathbb{V}$ that mimics (in $\mathbf{M}$) the change (in $\mathcal{S}$) induced by natural controller $\mathbb{Y}$ through the probability distribution $\mathbb{V}$ over the parameter set Λ .

In order to have meaningful results, especially in the choice of distributions, we will assume the set Λ is finite. Indeed, as explained in Section 1.2.2 on page 27, the set $\mathcal{D}$ must be a finite set of discrete distributions; without loss of generality, we only need to study the possible output values of these distributions, which we identify to Λ .

Now, we describe the Markov process $\mathbf{M}$ which we will build. In order to track both the position (at a region level) and the dynamics of the system at once, we consider the product set of pairs of regions and parameters which we define as follows:

$$\mathbf{Q} = \mathbf{\Omega} \times \Lambda \quad (3.18)$$

In this context, a transition $(z, \lambda) \rightarrow (z', \lambda')$ in the hybrid system will be abstracted as a transition $(R, \lambda) \rightarrow (R', \lambda')$ in the Markov process, where R, R' are such that $z \in R$ and $z' \in R'$.

Intuitively, the controller $\mathbb{Y}$ for $\mathcal{S}$ may enforce a change of the dynamics of $\mathcal{S}$ by realising a random variable following a probability distribution $\tilde{\mathbb{V}}_\lambda$ over the parameter set Λ (formally, this will happen through an action $\zeta_{\tilde{\mathbb{V}}_\lambda}$, as explained in Eq. (1.24) on page 30). Similarly to what we presented at the beginning of Section 3.2, we consider this discrete change happens in two steps: i) a variable $\mathbb{B}$ following a Bernoulli distribution of parameter $b \in [0, 1]$ is realised; ii) if $\mathbb{B} = 1$, then a variable $\tilde{\mathbb{V}}_\lambda$ is realised and a new value is picked for parameter λ .

First, we may need to define several new distributions based on $\mathbb{V}$, for the hybrid system. Indeed, as explained in Example 3.2.2, $\mathbb{V}$ might not affect all the components of a parameter value (in that example, $\mathbb{V}$ only affects 2 out of the 4 that compose the

parameter λ), and is as such a probability distribution *function* over Λ , i.e. $\mathbb{V} : \Lambda^2 \rightarrow [0, 1]$ takes two parameter values (one is the current parameter, the other is the next one) to map to a value in $[0, 1]$; in that case, since the distributions in a hybrid system (see Section 1.3, in particular above Definition 1.3.1 on page 27) are defined over the states in $\Omega \times \Lambda$, we need to construct one distribution $\mathbb{V}_\lambda$ for each possible value of the parameter $\lambda \in \Lambda$. Formally, given $\lambda \in \Lambda$ we construct the following split distribution:

$$\begin{aligned} \mathbb{V}_\lambda : \Lambda &\rightarrow [0, 1] \\ \lambda' &\mapsto \mathbb{V}(\lambda, \lambda') \end{aligned} \tag{3.19}$$

Thus, $\mathbb{V}_\lambda$ may only target new parameters $\lambda' \in \Lambda$ if $\mathbb{V}(\lambda, \lambda') > 0$.

Remark 3.2.11. Note that the split distributions $\mathbb{V}_\lambda$ are only required for the hybrid system: in the setting of Markov processes, since the actions induce probability distribution functions between the states (rather than a distribution *on* the states, as it is the case in hybrid systems), $\mathbb{V}$ can be transparently translated to that setting (see Eq. (3.26) on page 95). ◀

The action picked by $\mathbb{Y}$ translates to either a discrete change according to $\mathbb{V}_\lambda$ (similarly to other actions $\zeta_{\mathbb{D}}$ for some $\mathbb{D} \in \mathcal{D}$) or a self-loop (similarly to the action $\emptyset$), depending on the realisation of a random variable $\mathbb{B}$ following a Bernoulli distribution of parameter $b \in [0, 1]$. To induce both possibilities in a single distribution, we define the following distribution $\tilde{\mathbb{V}}_\lambda$:

$$\begin{aligned} \tilde{\mathbb{V}}_\lambda : \Lambda &\rightarrow [0, 1] \\ \lambda &\mapsto b\mathbb{V}_\lambda(\lambda') + (1 - b)\mathbf{1}_\lambda(\lambda') \end{aligned} \tag{3.20}$$

Thus, we have the set $\mathcal{V} = \{\tilde{\mathbb{V}}_\lambda \mid \lambda \in \Lambda\}$ of distributions; since Λ is finite, $\mathcal{V}$ is necessarily finite too. It induces the (finite) set $\mathcal{A}_{\mathcal{V}} = \{\zeta_{\tilde{\mathbb{V}}_\lambda} \mid \tilde{\mathbb{V}}_\lambda \in \mathcal{V}\}$ of actions. Finally, given a behaviour $\pi = s_0 s_1 \dots s_k$ for $\mathcal{S}$ such that $s_k = (z_k, \lambda_k)$ with λ_k , the controller $\mathbb{Y}$ is defined as follows:

$$\begin{aligned} \mathbb{Y} : (\Omega \times \Lambda)^+ &\rightarrow \mathbf{A} \\ \pi &\mapsto \zeta_{\tilde{\mathbb{V}}_{\lambda_k}}. \end{aligned} \tag{3.21}$$

Example 3.2.12. Let $\Delta = 0.1$ be a discretization step, then let $\Lambda = \{\lambda \in [0, 1]^4 \mid \exists i, j, k, l \in \mathbb{N}, \lambda = (i\Delta, j\Delta, k\Delta, l\Delta)\}$ be a parameter set. The components of the pa-

parameter values $\lambda \in \Lambda$ are denoted $(\beta_A, \beta_I, \delta_A, \delta_I)$. Let $\mathbb{B}$ be a Bernoulli distribution of parameter $b \in [0, 1]$. Let $\mathbb{V}$ be a Dirac distribution on $\mathbb{R}^2$, concentrated on $(0.5, 0.5)$.

If this distribution is used by the natural process to change the value of (β_A, β_I) it must be adapted to Λ . Consider the following function $\mathbb{V}'$:

$$\begin{aligned} \mathbb{V}' : \quad \Lambda &\rightarrow [0, 1] \\ (\beta_A, \beta_I, \delta_A, \delta_I) &\mapsto \begin{cases} 1, & \text{if } \beta_A = \beta_I = 0.5, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.22)$$

$\mathbb{V}'$ is clearly not a probability distribution over Λ , as both $\mathbb{V}'(0.5, 0.5, 0, 0) = 1$ and $\mathbb{V}'(0.5, 0.5, 1, 1) = 1$. Hence, given two parameter values $\lambda = (\beta_A, \beta_I, \delta_A, \delta_I)$ and $\lambda' = (\beta'_A, \beta'_I, \delta'_A, \delta'_I)$, the distribution function $\mathbb{V}_{\mathbb{Y}}$ used by $\mathbb{Y}$ is defined as follows:

$$\begin{aligned} \mathbb{V}_{\mathbb{Y}} : \quad \Lambda^2 &\rightarrow [0, 1] \\ (\lambda, \lambda') &\mapsto \begin{cases} 1, & \text{if } \beta'_A = \beta'_I = 0.5 \text{ and } (\delta_A, \delta_I) = (\delta'_A, \delta'_I), \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.23)$$

Since we must use distributions in $\mathcal{S}$, we split $\mathbb{V}_{\mathbb{Y}}$ into several distributions $\mathbb{V}_{\lambda}$, one for each parameter value $\lambda \in \Lambda$; moreover, we take the Bernoulli distribution $\mathbb{B}$ into account, and build a distribution $\tilde{\mathbb{V}}_{\lambda}$ analogous to Eq. (3.20), where $\mathbf{1}_{\lambda}(\cdot)$ denotes the Dirac distribution on Λ , concentrated on λ :

$$\begin{aligned} \tilde{\mathbb{V}}_{\lambda} : \quad \Lambda &\rightarrow [0, 1] \\ \lambda' &\mapsto b\mathbb{V}_{\mathbb{Y}}(\lambda, \lambda') + (1 - b)\mathbf{1}_{\lambda}(\lambda') \end{aligned} \quad (3.24)$$

Finally, we have:

$$\mathcal{V} = \{\tilde{\mathbb{V}}_{\lambda} \mid \lambda \in \Lambda\}. \quad (3.25)$$

Note that because Λ is finite, $\mathcal{V}$ is finite as well. ◀

For the remaining of this chapter, we fix a parameter $b \in (0, 1)$ and consider the Bernoulli distribution of parameter b , denoted by $\mathbb{B}$; we also fix the discrete distribution function $\mathbb{V}$ over Λ . Denoting by $\mathbf{1}_{\lambda_i}$ the Dirac distribution on Λ concentrated on λ_i , the discrete evolutions (induced by probabilistic controller $\mathbb{Y}$) of $\mathcal{S}$ are abstracted (in $\mathbf{M}$) as

the following transition probability distribution function:

$$\begin{aligned} \mathbf{P}_{\mathbb{V}} : \quad \mathbf{Q} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), (R_j, \lambda_j)) &\mapsto \begin{cases} b\mathbb{V}(\lambda_i, \lambda_j) + (1 - b)\mathbf{1}_{\lambda_i}(\lambda_j), & \text{if } R_i = R_j \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.26)$$

The continuous evolutions of $\mathcal{S}$ are abstracted as a transition probability distribution function $\mathbf{P}_{\tau}$ over the set $\mathbf{Q}$, analogous to Eq. (3.16) on page 88, and defined as follows:

$$\begin{aligned} \mathbf{P}_{\tau} : \quad \mathbf{Q} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), (R_j, \lambda_j)) &\mapsto \begin{cases} p_{i,j}^{\lambda_i}, & \text{if } \lambda_i = \lambda_j, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.27)$$

Finally, given a state $(R_i, \lambda_i) \in \mathbf{Q}$, we denote by $\mathbf{1}_{(R_i, \lambda_i)}$ the Dirac distribution concentrated on (R_i, λ_i) . Then, we consider the set $\mathbf{A} = \{a_{\tau}, a_{\mathbb{V}}, \emptyset\}$ of actions (see Definition 1.4.3 on page 32) for $\mathbf{M}$, and define the probability distribution function $\mathbf{P}$ of $\mathbf{M}$ as follows:

$$\begin{aligned} \mathbf{P} : \quad \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), a, (R_j, \lambda_j)) &\mapsto \begin{cases} \mathbf{P}_{\tau}((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_{\tau} \\ \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_{\mathbb{V}} \\ \mathbf{1}_{(R_i, \lambda_i)}((R_j, \lambda_j)), & \text{if } a = \emptyset \end{cases} \end{aligned} \quad (3.28)$$

Intuitively, the action a_{τ} lets $\mathbf{M}$ evolve according to the probability of the underlying Markov chain abstracting the ODE instantiated with the current value λ_i , i.e. according to the dynamics (in the hybrid system) induced by λ_i ; the action $a_{\mathbb{V}}$ enforces a transition which encapsulates both the Bernoulli process $\mathbb{B}$ and the distribution $\mathbb{V}$ corresponding to the natural changes in $\mathcal{S}$; the action $\emptyset$ forces a self-loop on the current state of $\mathbf{M}$.

As expected, $\mathbf{P}$ defines a probability distribution:

Lemma 3.2.13.

Let $\mathbf{P}_{\mathbb{V}}$ and $\mathbf{P}_{\tau}$ be the functions defined in Eqs. (3.26) and (3.27) on page 95, respectively.

The function

$$\begin{aligned}
 \mathbf{P} : \quad \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} &\rightarrow [0, 1] \\
 ((R_i, \lambda_i), a, (R_j, \lambda_j)) &\mapsto \begin{cases} \mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\tau \\ \mathbf{P}_\forall((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\forall \\ \mathbf{1}_{(R_i, \lambda_i)}((R_j, \lambda_j)), & \text{if } a = \emptyset \end{cases} \quad (3.29)
 \end{aligned}$$

defines a transition probability distribution over the states of $\mathbf{Q}$. $\triangleleft$

The proof of Lemma 3.2.13 is given in Section C.5, on page 135.

It follows from Lemma 3.2.13 that we have constructed a Markov decision process:

Proposition 3.2.14.

Given an initial distribution $\mathbf{I}$ over $\mathbf{Q}$, the tuple $\mathbb{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ is a Markov decision process. $\triangleleft$

A trimmed example of the Markov decision process $\mathbf{M}$ corresponding to the running example (without accounting for control policies) is presented in Fig. 3.9, in which the initial distribution $\mathbf{I}$ is a Dirac Distribution concentrated on (R_e, λ_2) . Again, it is trivial to define an initial distribution corresponding to the initial state (z_0, λ_0) of a hybrid system by considering the Dirac distribution concentrated on (R_0, λ_0) , where $z_0 \in R_0$.

The goal of this section is to establish a link between the MDP $\mathbf{M}$ produced in Proposition 3.2.14 on one side, and the continuous behaviour of $\mathcal{S}$ and the probabilistic controller $\mathbb{Y}$ on the other; this link is established in Lemma 3.2.16. It is built upon the notion of τ -simulation given in Definition 3.2.8 on page 89, adapted to the setting of hybrid dynamical systems (i.e. by considering any transition in $\mathcal{S}$, in particular discrete transitions induced by a distribution):

Definition 3.2.15 ((almost-) τ -simulation of a Hybrid Dynamical System). Let $\tau > 0$ be a time discretization. Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system.

We say that a transition system B with state set Q_B τ -simulates $\mathcal{S}$ through a relation $\mathcal{R} \subseteq (\Omega \times \Lambda) \times Q_B$ iff:

1. for any initial state $s_0 \in \Omega \times \Lambda$, there exists an initial state $q_0 \in Q_B$ such that $(s_0, q_0) \in \mathcal{R}$;

2. for any pair $(s, q) \in \mathcal{R}$, any action $a \in \mathcal{A}$ and any a -successor s' of s , there exists a successor q' of q such that $(s', q') \in \mathcal{R}$.

We call **almost- τ -simulation** a relation $\mathcal{R} \subseteq (\Omega \times \Lambda) \times Q_B$ that satisfies item 2. almost everywhere, i.e. such that the set $\check{\mathcal{R}}$ of pairs $(s, q) \in \mathcal{R}$ such that a transition $s \rightarrow_a s'$ in $\mathcal{S}$ cannot be matched by a transition $q \rightarrow q'$ in B is such that $m(\check{\mathcal{R}}) = 0$. If there exists an almost- τ -simulation relation between $\mathcal{S}$ and B , we say that B **almost- τ -simulates** $\mathcal{S}$. $\triangleleft$

In the setting of MDPs, a state $q' \in \mathbf{Q}$ is a successor of q iff there exists an action $a \in \mathbf{A}$, such that $\mathbf{P}(q, a, q') > 0$.

Lemma 3.2.16 (Almost- τ -simulation of the hybrid system and $\mathbb{Y}$).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a controller, whose action concatenates a Bernoulli distribution $\mathbb{B}$ of parameter $b \in [0, 1]$ and a discrete distribution function $\mathbb{V}$ over the parameter set Λ . Let (z_0, λ_0) be an initial state for $\mathcal{S}$. Consider the action set $\mathbf{A} = \{a_\tau, a_\mathbb{V}\}$, as well as the probability distribution function $\mathbf{P}$ defined in Eq. (3.28). Consider the following initial distribution over the states of $\mathbf{Q}$:

$$\mathbf{I}(R, \lambda) = \begin{cases} 1, & \text{if } z_0 \in R_0 \text{ and } \lambda = \lambda_0, \\ 0, & \text{otherwise.} \end{cases} \quad (3.30)$$

The MDP $\mathbb{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ almost- τ -simulates $\mathcal{S}$ under controller $\mathbb{Y}$. $\triangleleft$

The proof of Lemma 3.2.16 is given in Section C.6, on page 137. For the same reasons as for Markov chains abstracting a single dynamics, the impossibility to simulate all fixed-time behaviours of duration τ comes from the difficulties encountered when trajectories in $\mathcal{S}$ stop on the border of a region; again, this only concerns a set of measure 0, hence we have an **almost- τ -simulation** of the hybrid system $\mathcal{S}$ by its the abstraction $\mathbb{M}$.

In Example 3.2.7, we presented Fig. 3.7 on page 89 in which the probabilities are unknown variables. These Markov chains are called *parametric*, and have been the topic of extensive studies in the past years [35, 64]; model checkers such as STORM [30] incorporate tools to analyze these models and compute symbolic functions of these unknowns representing the parametric probability of satisfying a given property, such as φ introduced in Example 3.2.4. These functions can then be evaluated on the true values (or intervals) of the transition probabilities so that the probability of satisfaction of φ can effectively be computed. The example below illustrates how such a formula can be found.

Example 3.2.17. Consider a two-parameter set $\Lambda = \{\lambda_1, \lambda_2\}$. Consider $\mathcal{S}$ is given a controller $\mathbb{Y}$ with Bernoulli parameter $b \in [0, 1]$ and a distribution function $\mathbb{V}$ over the pairs of parameters in Λ . We assume that the action of the controller $\mathbb{Y}$ is always preceded by a continuous transition $\xrightarrow{\tau}$ in $\mathcal{S}$, such that every other transition in an execution of $\mathcal{S}$ is continuous. To enforce this constraint, $\mathbb{Y}$ may only act if the length of the input behaviour is even, i.e. if $|\pi| \equiv 0 \pmod{2}$. Formally, given a behaviour $\pi = s_0 \dots s_k$ such that $s_k = (z_k, \lambda_k)$, we have

$$\begin{aligned} \mathbb{Y} : (\Omega \times \Lambda)^+ &\rightarrow \mathcal{A} \\ \pi &\mapsto \zeta_{\tilde{\mathbb{V}}_{\lambda_k}}, \end{aligned} \tag{3.31}$$

where $\zeta_{\tilde{\mathbb{V}}_{\lambda_k}}$ is the action inducing a change according to $\tilde{\mathbb{V}}_{\lambda_k}$, a distribution that encompasses both the Bernoulli distribution $\mathbb{B}$ and the split distribution $\mathbb{V}_{\lambda_k}$ over Λ corresponding to λ_k , i.e. $\tilde{\mathbb{V}}_{\lambda_k}(\lambda) = b\mathbb{V}_{\lambda_k}(\lambda) + (1 - b)\mathbf{1}_{\lambda_k}(\lambda)$.

Let $R_{df} \subset \Omega$ be a region containing only the disease-free equilibrium DFE , then $R_e = \Omega \setminus R_{df}$ be its complement. We assume that the initial distribution $\mathbf{I}$ is a Dirac distribution concentrated on (R_e, λ_2) . Given a time step τ , we can build the parametric Markov decision process depicted in Fig. 3.9, where some transitions have been removed to limit cluttering.

Now, consider a time step value $\tau = 7$ days.

Let $\mathbf{M}$ be the Markov decision process abstracting $\mathcal{S}$, as obtained in this section. We consider a deterministic controller Ψ for $\mathbf{M}$, in which the action corresponding to the effect of $\mathbb{V}$ in $\mathcal{S}$ is denoted $a_{\mathbb{V}}$, and which is alternate:

$$\begin{aligned} \Psi : \mathbf{Q}^+ &\rightarrow \mathbf{A} \\ \pi &\mapsto \begin{cases} a_{\tau}, & \text{if } |\pi| \text{ is odd,} \\ a_{\mathbb{V}}, & \text{if } |\pi| \text{ is even.} \end{cases} \end{aligned} \tag{3.32}$$

Note that this controller mixes actions (in $\mathbf{M}$) corresponding to continuous or discrete (in $\mathcal{S}$) transitions. This is valid from a theoretical point of view, but a modeller may find this strange; we do this in order to have a simple example, and we will see in Section 3.2.4 how a MDP with multiple controllers can be produced, so that it can only induce one type of transition.

As explained after Definition 1.4.4 (see page 34), the controller Ψ induces an (infinite) Markov chain whose states are the possible behaviours of $\mathbf{M}$ under Ψ . In our case, this

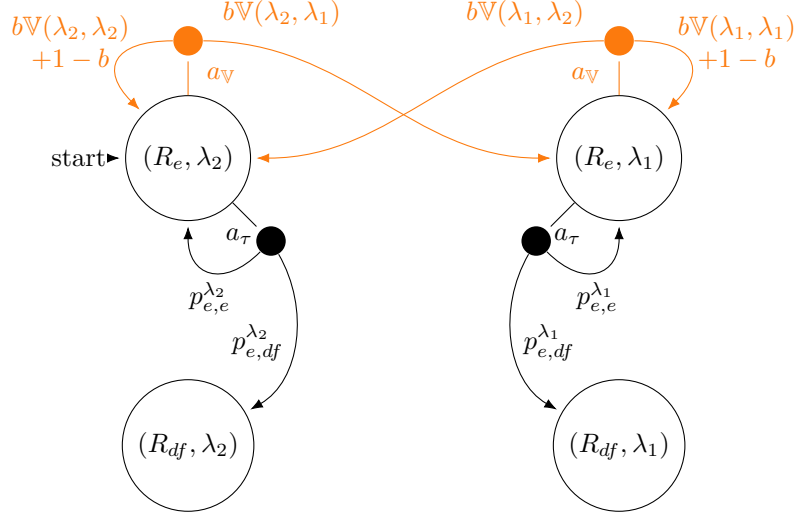


Figure 3.9: Markov decision process resulting from the coupling of the chains obtained in Section 3.2.2, where the initial distribution $\mathbf{I}$ is a Dirac distribution concentrated on (R_e, λ_2) . Black transitions correspond to the a_τ action, i.e. the natural, continuous evolution described in Section 3.2. Orange transitions correspond to the a_v action, i.e. the influence of probabilistic process $\mathbb{Y}$ in $\mathcal{S}$; for this action, a Bernoulli variable of parameter $b \in [0, 1]$ (corresponding to the parameter in $\mathbb{Y}$) is first realised, and induces a transition back to the current state if $\mathbb{B} = 0$ (i.e. with probability $1 - b$), and induces a transition towards the new state according to the realisation of variable $\mathbb{V}$ if $\mathbb{B} = 1$ (i.e. with probability b). The black- and orange-filled circles are not actual states in the MDP, but virtual ones to help understanding the evolution. Some transitions have been removed to limit cluttering.

induced Markov chain $\mathbb{M}_\Psi = (\mathbb{S}, \mathbb{P}_\Psi, \mathbb{I})$ is defined as follows:

- $\mathbb{S} = \mathbf{Q}^+$;
- given a behaviour $\pi = s_0 s_1 \dots s_k$ in $\mathbb{S}$ such that $s_k = (R_k, \lambda_k)$ and a state $s_{k+1} = (R_{k+1}, \lambda_{k+1}) \in \mathbf{Q}$, the transition probability distribution function $\mathbb{P}_\Psi$ is:

$$\begin{aligned} \mathbb{P}_\Psi : \quad \mathbb{S} \times \mathbb{S} &\rightarrow [0, 1] \\ (\pi, \pi s_{k+1}) &\mapsto \begin{cases} \mathbf{P}_\tau(s_k, s_{k+1}), & \text{if } |\pi| \text{ is odd,} \\ \mathbf{P}_\mathbb{V}(s_k, s_{k+1}), & \text{if } |\pi| \text{ is even.} \end{cases} \end{aligned} \quad (3.33)$$

- given a behaviour $\pi = s_0 \dots s_k \in \mathbf{S}^+$, the initial probability distribution $\mathbb{I}_\Psi$ is:

$$\begin{aligned} \mathbb{I}_\Psi : \mathbb{S} &\rightarrow [0, 1] \\ \pi &\mapsto \begin{cases} \mathbf{I}(s_0), & \text{if } \pi = s_0 \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.34)$$

Assume we are interested in the property $\varphi =$ “the population reaches R_{df} within two weeks.” Using parametric model checking techniques, or tools such as STORM [30], we can compute a formula for the probability of reaching R_{df} within two transitions in the parametric Markov decision process $\mathbf{M}$, which corresponds to the probability $P(\varphi)$ of satisfying φ . This formula is parametric in the values of the transition probabilities between the regions, as well as the distribution followed by random process $\mathbb{V}$, and is given by Eq. (3.35).

$$\begin{aligned} P(\varphi) &= c_0(\lambda_2) + c_1(\lambda_1, \lambda_2, \mathbb{V}) b + c_2(\lambda_1, \lambda_2, \mathbb{V}) b^2, \\ &\text{with} \\ c_0(\lambda_2) &= - \left[(p_{e,df}^{\lambda_2})^2 - 2p_{e,df}^{\lambda_2} \right], \\ c_1(\lambda_1, \lambda_2, \mathbb{V}) &= - \left[\mathbb{V}(\lambda_2, \lambda_1)(p_{e,df}^{\lambda_1})^2 - 3\mathbb{V}(\lambda_2, \lambda_1)p_{e,df}^{\lambda_1} + \mathbb{V}(\lambda_2, \lambda_1)p_{e,df}^{\lambda_2}p_{e,df}^{\lambda_1} \right. \\ &\quad \left. + 3p_{e,df}^{\lambda_2} - 2(p_{e,df}^{\lambda_2})^2 - 3\mathbb{V}(\lambda_2, \lambda_2)p_{e,df}^{\lambda_2} + 2\mathbb{V}(\lambda_2, \lambda_2)(p_{e,df}^{\lambda_2})^2 \right], \\ c_2(\lambda_1, \lambda_2, \mathbb{V}) &= - \left[\mathbb{V}(\lambda_2, \lambda_2)^2(p_{e,df}^{\lambda_2})^2 - \mathbb{V}(\lambda_2, \lambda_1)(p_{e,df}^{\lambda_1})^2 \right. \\ &\quad + \mathbb{V}(\lambda_2, \lambda_1)\mathbb{V}(\lambda_1, \lambda_2)p_{e,df}^{\lambda_1}p_{e,df}^{\lambda_1} - \mathbb{V}(\lambda_2, \lambda_1)\mathbb{V}(\lambda_1, \lambda_1)p_{e,df}^{\lambda_1} \\ &\quad - \mathbb{V}(\lambda_2, \lambda_1)\mathbb{V}(\lambda_1, \lambda_2)p_{e,df}^{\lambda_2} - p_{e,df}^{\lambda_2} + 2\mathbb{V}(\lambda_2, \lambda_1)p_{e,df}^{\lambda_1} + (p_{e,df}^{\lambda_2})^2 \\ &\quad - \mathbb{V}(\lambda_2, \lambda_1)p_{e,df}^{\lambda_2}p_{e,df}^{\lambda_1} + 2\mathbb{V}(\lambda_2, \lambda_2)p_{e,df}^{\lambda_2} - \mathbb{V}(\lambda_2, \lambda_1)\mathbb{V}(\lambda_1, \lambda_2)p_{e,df}^{\lambda_1} \\ &\quad - 2\mathbb{V}(\lambda_2, \lambda_2)(p_{e,df}^{\lambda_2})^2 + \mathbb{V}(\lambda_2, \lambda_1)\mathbb{V}(\lambda_2, \lambda_2)p_{e,df}^{\lambda_1}p_{e,df}^{\lambda_2} \\ &\quad \left. - \mathbb{V}(\lambda_2, \lambda_2)^2(p_{e,df}^{\lambda_2})^2 + \mathbb{V}(\lambda_2, \lambda_1)^2(p_{e,df}^{\lambda_1})^2 \right]. \end{aligned} \quad (3.35)$$

Remark that Eq. (3.35) corresponds to the probability of satisfying φ in the *abstraction* of the hybrid system, not the system itself. Still, it gives an indication of what the probability could be in the original system (without guarantees). Moreover, instantiating (parts of) the different variables of this formula, for instance b or $p_{e,df}^\lambda$, allows to compute

the probability of satisfying φ under these values: if they are evaluated using the original system, for instance through SMC techniques, then it gives an indication of the probability of satisfaction of φ . In this example, in the case where an environmental change *cannot* occur, then $b = 0$ and we have:

$$\begin{aligned}
 P(\varphi) &= c_0(\lambda_2) \\
 &= -\left[(p_{e,df}^{\lambda_2})^2 - 2p_{e,df}^{\lambda_2}\right] \\
 &= 2p_{e,df}^{\lambda_2} - (p_{e,df}^{\lambda_2})^2 \\
 &= p_{e,df}^{\lambda_2} + (1 - p_{e,df}^{\lambda_2})p_{e,df}^{\lambda_2} \\
 &= p_{e,df}^{\lambda_2} + (p_{e,e}^{\lambda_2} \cdot p_{e,df}^{\lambda_2}).
 \end{aligned} \tag{3.36}$$

In that case, the MDP $\mathbf{M}$ degenerates into the Markov chain induced by λ_2 (since the parameter cannot be changed); Eq. (3.36) corresponds to the probability of reaching (R_{df}, λ_2) from (R_e, λ_2) with at most two (black) transitions in the Markov process presented in Fig. 3.9. ◀

3.2.4 Enhancement of the Markov decision process

We now move to the second layer of hybridation of the Markov process, which corresponds to the introduction of a second controller.

To this end, the Markov decision process $\mathbf{M}$ produced in Section 3.2.3 is augmented with a second controller Φ , whose aim is to mimic the action of the controller $\mathbb{X}$ in $\mathcal{S}$; at each time point in a run of the MDP where it is allowed to, Φ will choose an action that translates to a transition probability distribution among the states of the MDP. Intuitively, the effect of such an action in the MDP corresponds to an enforced change of dynamics or position in the hybrid system, e.g. the realisation of control process $\mathbb{X}$ defined in Example 3.2.2 on page 81. Because this controller is deterministic w.r.t. the behaviour π of the system and has multiple choices available at each timepoint, it is possible to devise a strategy to determine how this controller operates, i.e. determining the choice(s) it may make based on π .

Recall that, in Eq. (1.21) on page 29, we added a set $\mathcal{W}$ of Dirac distributions to emulate a waiting action in $\mathcal{S}$. This will be used in this section, as Φ may want to pass its turn. Consider the set $\mathcal{D} \cup \mathcal{W}$ of distributions for the hybrid system: this set corresponds to the collection of changes that can be enforced by the controller $\mathbb{X}$, through the set

$\mathcal{A}_{\mathbb{X}} = \{\emptyset\} \cup \{\zeta_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$ of actions (see Eq. (1.23) on page 29). When an action $a \in \mathcal{A}_{\mathbb{X}}$ is picked by $\mathbb{X}$ in the state (z_i, λ_i) , the system transitions to a new state (z_j, λ_j) according to a distribution $\mathbb{D} \in \mathcal{D} \cup \mathcal{W}$, i.e. with a probability $\mathbb{D}((z_j, \lambda_j))^4$.

To encode this controller in the setting of a MDP, we define a set of actions that the controller Φ can pick from. Again, two types of actions are considered:

- an action $\emptyset$ that forces the MDP to stay in its current state, which is done by considering a set $\tilde{\mathcal{W}}$ of Dirac distributions concentrated on each state of $\mathbf{M}$;
- a set of actions $a_{\mathbb{D}}$ that enforces a probabilistic reset of state according to the distribution $\mathbb{D} \in \mathcal{D}$; for convenience, we might write $\mathbb{D}^{\Lambda}$ or $\mathbb{D}^{\Omega}$ whether $\mathbb{D} \in \mathcal{D}^{\Lambda}$ or $\mathbb{D} \in \mathcal{D}^{\Omega}$ (see Definition 1.3.1 on page 27).

With these notations, we consider the following set of actions:

$$\mathbf{A} = \{\emptyset, a_{\tau}, a_{\mathbb{V}}\} \cup \{a_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}. \quad (3.37)$$

We consider again the product set $\mathbf{Q} = \mathbf{\Omega} \times \mathbf{\Lambda}$. Given a state $(R_i, \lambda_i) \in \mathbf{Q}$, the Dirac distribution concentrated on (R_i, λ_i) is denoted by $\mathbf{1}_{(R_i, \lambda_i)}$. Using the transition probability functions $\mathbf{P}_{\mathbb{V}}$ (see Eq. (3.26) on page 95) and $\mathbf{P}_{\tau}$ (see Eq. (3.27) on page 95) of the Markov decision process aggregating all dynamics defined in Section 3.2.3, we define the transition probability distribution function $\mathbf{P}$ of $\mathbf{M}$ as follows:

$$\begin{aligned} \mathbf{P} : \quad \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), a, (R_j, \lambda_j)) &\mapsto \begin{cases} \mathbf{P}_{\tau}((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_{\tau}, \\ \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_{\mathbb{V}}, \\ \mathbf{1}_{(R_i, \lambda_i)}(R_j, \lambda_j), & \text{if } a = \emptyset, \\ \mathbb{D}^{\Lambda}(\lambda_j), & \text{if } a = a_{\mathbb{D}^{\Lambda}} \text{ and } R_i = R_j, \\ \sum_{\substack{z_j \in R_j \\ \mathbb{D}^{\Omega}(z_j) \neq 0}} \mathbb{D}^{\Omega}(z_j), & \text{if } a = a_{\mathbb{D}^{\Omega}} \text{ and } \lambda_i = \lambda_j, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.38)$$

⁴For convenience, we abuse notations and write $\mathbb{D}$ indifferently whether it affects the position z or the parameter λ of $\mathcal{S}$

Proposition 3.2.18.

The function $\mathbf{P}$ defined in Eq. (3.38) defines a transition probability distribution function.

◁

It follows that we built a new Markov decision process:

Proposition 3.2.19.

Given an initial distribution $\mathbf{I}$ over $\mathbf{Q}$, the tuple $\mathbf{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ is a Markov decision process.

◁

Note that the MDP framework allows any controller to pick any action; the specific behaviours (e.g. probabilistic behaviours, restrictions on the actions, etc.) of a controller may be defined within its description, or as it is the case here, by restricting when it can act.

Example 3.2.20. We build upon Example 3.2.17, in which we have two parameter values λ_1, λ_2 and a two-region partition $\Omega = \{R_{df}, R_e\}$. We add two parameter values λ_3, λ_4 , such that $\mathbb{V}(\lambda_i, \lambda_j) > 0 \Leftrightarrow \lambda_i, \lambda_j \in \{\lambda_1, \lambda_2\}$ or $\lambda_i, \lambda_j \in \{\lambda_3, \lambda_4\}$: the natural controller $\mathbb{Y}$ cannot force a switch from e.g. λ_1 to λ_3 .

First, we only consider the MDP encompassing $\Omega, \lambda_1, \lambda_2, \lambda_3, \lambda_4$ and two actions a_τ and $a_\mathbb{V}$; it is depicted in Fig. 3.10.

As we can see, since the system starts in (R_e, λ_2) and there is no transition between the states $(R_e, \lambda_1), (R_{df}, \lambda_1), (R_e, \lambda_2)$ or (R_{df}, λ_2) and the states $(R_e, \lambda_3), (R_{df}, \lambda_3), (R_e, \lambda_4)$ or (R_{df}, λ_4) , the latter states are unreachable. In that case, the evolution of the MDP heavily depends on its starting state, since its graph presents two disconnected subgraphs.

We now consider the set $\mathcal{D}$ of distributions in $\mathcal{S}$, and we augment the MDP with the second set of actions $\{a_\mathbb{D} \mid \mathbb{D} \in \mathcal{D}\}$. We assume the existence of two distributions $\mathbb{D}_1$ and $\mathbb{D}_{1,3}$ which follow uniform discrete distribution over $\{\lambda_1\}$ and $\{\lambda_1, \lambda_3\}$ respectively, and consider their associated actions $a_{\mathbb{D}_1}, a_{\mathbb{D}_{1,3}}$. We assume $a_{\mathbb{D}_1}$ is enabled in (R_{df}, λ_3) and $a_{\mathbb{D}_{1,3}}$ is enabled in both (R_e, λ_1) and (R_e, λ_3) . These distributions allow to probabilistically switch between parameter values λ in $\mathcal{S}$, in particular to enforce a change that could not happen naturally (through the process $\mathbb{Y}$ defined in Example 3.2.2 on page 82). The resulting (parametric) MDP is depicted in Fig. 3.11, in which we also consider the action $\emptyset$ for completeness.

As we can see, it is now possible to switch between the left and right parts of the MDP,

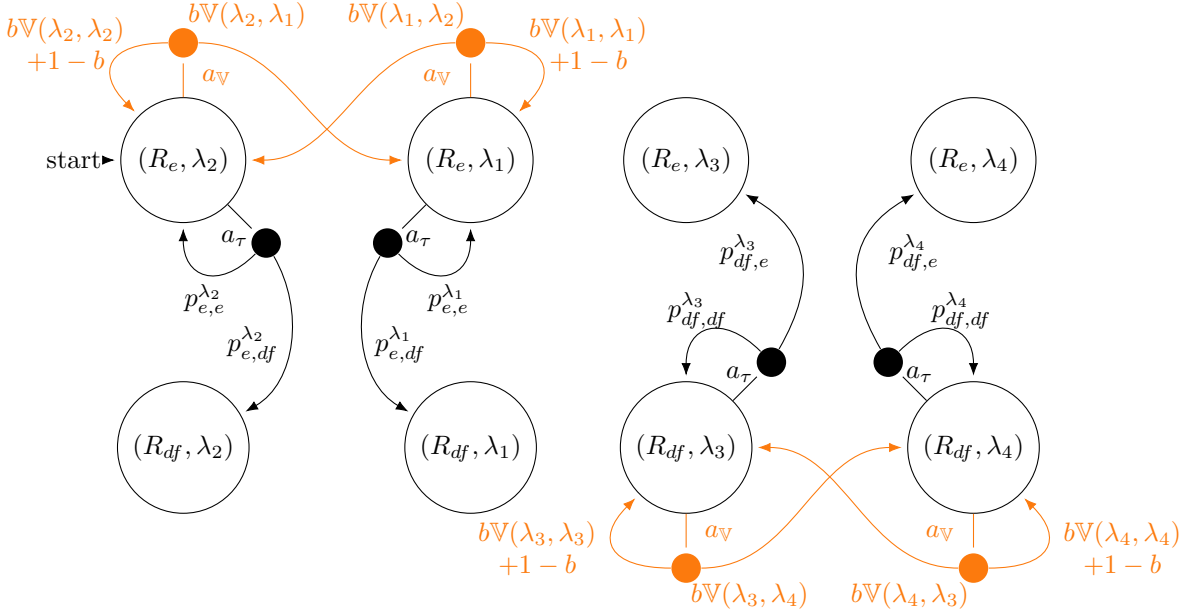


Figure 3.10: The process resulting from the addition of λ_3, λ_4 and a controller to the Markov chain produced in Example 3.2.17. Colors have the same meaning as in Fig. 3.9. The black- and orange-filled circles are not actual states in the MDP, but virtual ones to help understanding the evolution. Remark that the right side of the figure is disconnected, thus unreachable, from the left side.

i.e. between the sets $\{\lambda_1, \lambda_2\}$ and $\{\lambda_3, \lambda_4\}$ of parameters through actions $a_{\mathbb{D}_1}$ and $a_{\mathbb{D}_{1,3}}$. Moreover, there are now states $((R_e, \lambda_1), (R_e, \lambda_3))$ and (R_{df}, λ_3) in which the controller Φ can make actual choices between a_τ - and $a_{\mathbb{D}}$ -actions. Note that adding $a_{\mathbb{D}}$ -actions does not introduce new behaviours w.r.t. the hybrid system, as they are the direct equivalent (relative to the MDP) of distributions $\mathbb{D} \in \mathcal{D}$ in the hybrid system. ◀

The MDP produced in Proposition 3.2.19 can now be used for devising strategies that ensure the satisfaction of properties of interest. Indeed, MDPs are easy to work with and there exist various checkers, such as STORM [30] or PRISM [41], which allow the formal verification of properties, and the elaboration of strategies in this setting.

The goal of this chapter is to formally characterize a link between the behaviours of the hybrid system and those of the MDP, so that the MDP can be used as a proxy of the hybrid system; this link is established in Theorem 3.2.21, which is an extension of Lemma 3.2.16 (see page 97) and is also built upon the notion of τ -simulation of hybrid systems (see Definition 3.2.15 on page 96).

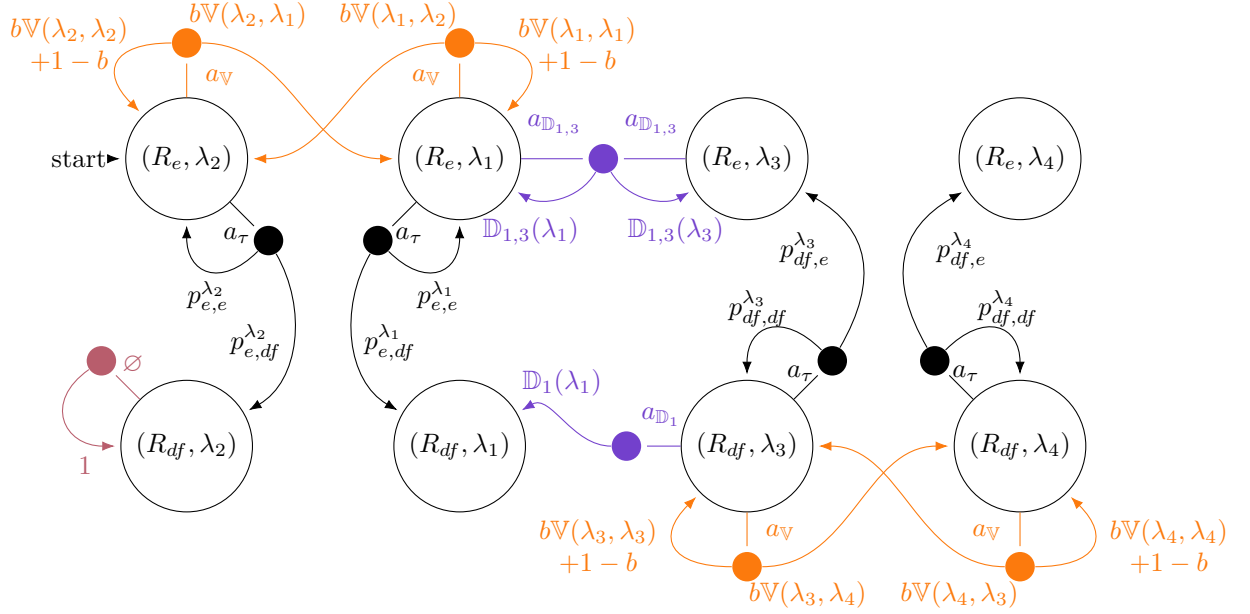


Figure 3.11: The process resulting from the addition of distributions $\mathbb{D}_1$ and $\mathbb{D}_{1,3}$ to the MDP pictured in Fig. 3.10. Orange and black have the same meaning. Purple transitions correspond to actions $a_{\mathbb{D}}$ for some distributions $\mathbb{D} \in \{\mathbb{D}_1, \mathbb{D}_{1,3}\}$. The right side of the figure is now connected, thus reachable, from the left side through these transitions. The dark red transition corresponds to the waiting action $\emptyset$, which allows the system to stay in its current state. Again, many actions and transitions have been removed to improve readability.

Theorem 3.2.21 (Almost- τ -simulation of the hybrid system).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a controller, whose action concatenates a Bernoulli distribution $\mathbb{B}$ of parameter $b \in [0, 1]$ and a discrete distribution function $\mathbb{V}$ over the parameter set Λ . Let (z_0, λ_0) be an initial state for $\mathcal{S}$. Consider the action set $\mathbf{A} = \{a_\tau, a_\nabla, \emptyset\} \cup \{a_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$, as well as the probability distribution function $\mathbf{P}$ defined in Eq. (3.38). Consider the following initial distribution over the states of $\mathbf{Q}$:

$$\mathbf{I}(R, \lambda) = \begin{cases} 1, & \text{if } z_0 \in R_0 \text{ and } \lambda = \lambda_0, \\ 0, & \text{otherwise.} \end{cases} \quad (3.39)$$

The MDP $\mathbf{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ almost τ -simulates $\mathcal{S}$. ◁

The proof of Theorem 3.2.21 is given in Section C.7, on page 139. For the same reasons

as for Lemma 3.2.16, the impossibility to simulate all fixed-time behaviours of duration τ comes from the difficulties encountered when trajectories in $\mathcal{S}$ stop on the border of a region; again, it only concerns a set of measure 0, hence we have an **almost**- τ -simulation of the hybrid system $\mathcal{S}$ by its MDP abstraction $\mathbf{M}$. Theorem 3.2.21 enables the transfer of some properties of $\mathbf{M}$ to $\mathcal{S}$, through Corollary 3.2.22.

Corollary 3.2.22 (Bounded reachability).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a natural process controller related to a distribution function $\mathbb{V}$ over the parameter set Λ . Given a second controller $\mathbb{X}$ for $\mathcal{S}$, we assume that $\mathbb{X}$ and $\mathbb{Y}$ can pick one action each, in that order, and that an action ζ_τ is always picked once before $\mathbb{X}$ can make a choice. Let $\mathbf{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ be the MDP abstraction of $\mathcal{S}$, then let Ψ be the controller mimicking $\mathbb{Y}$ by being able to pick the corresponding action $a_\mathbb{V}$ in $\mathbf{A}$.

Let $i \in \mathbb{N}_{>0}$ be a positive integer, such that $i = 3k$ for some $k \in \mathbb{N}$. Let $R^ \in \mathfrak{Q}$ be a target region in Ω . Let Φ be a second controller for $\mathbf{M}$. We assume that Φ and Ψ can pick one action each, in that order, and that an action a_τ is always picked once before Φ can make a choice.*

If Φ is such that $\mathbf{M}$ reaches R^ in i steps or less with probability 1, then there exists a corresponding controller $\mathbb{X}$ for $\mathcal{S}$ such that $\mathcal{S}$ reaches R^* in at most i transitions, and therefore in at most $k\tau$ time units, with probability 1.*

In this case, $\mathbb{X}$ consists in a sequence of choices for $\mathcal{S}$ corresponding to the actions chosen in $\mathbf{M}$ under controller Φ , adapted to its setting:

- $a_\mathbb{D}$ translates to having process $\mathbb{X}$ choose action $\zeta_\mathbb{D}, \mathbb{D} \in \mathcal{D}$ in order to influence the evolution of $\mathcal{S}$ in a similar manner as $\zeta_\mathbb{D}$ influences $\mathbf{M}$;
- $\emptyset$ translates to having $\mathbb{X}$ pick the Dirac distribution $\mathbf{1}_\lambda \in \mathcal{W}$, where λ is the current parameter value.

◁

By construction, the controller $\mathbb{X}$ obtained through Corollary 3.2.22 has the same constraints as Φ .

The proof of Corollary 3.2.22 is given on Section C.8 on page 141. It must be noted that Corollary 3.2.22 is relative to bounded time properties. Indeed, Theorem 3.2.21 states that the MDP $\mathbf{M}$ τ -simulates $\mathcal{S}$ except for a set $\check{\mathcal{R}}$ of pairs $(z, R) \in \Omega \times \mathfrak{Q}$, whose

measure is 0. While all finite bounds induce a set of non-simulated trajectories whose measure is 0, the infinite sum of all sets is not guaranteed to have a measure 0. Hence, we leave unbounded time properties for future work.

An example of derived controller $\mathbb{X}$ is given in Table 3.6 of Section 3.3.

3.3 Complete study of the running example

3.3.1 Setting of the study

We first recall the setting of the study, i.e. the function f defining our model in Eq. (3.4) on page 75, as well as the meaning and values of the different components of parameter λ given in Table 3.1 on page 74; the corresponding (compact) phase space Ω is defined in Eq. (3.5) on page 75.

Our model is given by the following system of ordinary differential equations:

$$\begin{cases} \frac{dS}{dt} = \mu - (\beta_A A(t) + \beta_I I(t)) S(t) - (\mu + \nu) S(t) + \gamma(1 - A - I), \\ \frac{dA}{dt} = (\beta_A A(t) + \beta_I I(t)) S(t) - (\alpha + \delta_A + \mu) A(t), \\ \frac{dI}{dt} = \alpha A(t) - (\delta_I + \mu) I(t), \end{cases} \quad (3.40)$$

For convenience, we introduce the notations $z = (S, A, I) \in \mathbb{R}^3$ and we write

$$f(z, \lambda) = \begin{pmatrix} \mu - (\beta_A A + \beta_I I) S - (\mu + \nu + \gamma) S + \gamma(1 - A - I) \\ (\beta_A A + \beta_I I) S - (\alpha + \delta_A + \mu) A \\ \alpha A - (\delta_I + \mu) I \end{pmatrix}. \quad (3.41)$$

We consider the parameter set Λ given in Table 3.2 on page 108, where we recall the value r_0 associated to each value, as well the stability of the equilibria DFE and EE .

For each $\lambda \in \Lambda$, we consider the Dirac distribution $\mathbf{1}_\lambda$ concentrated on λ ; these distributions form the set $\mathcal{W}$ of “waiting” distributions in $\mathcal{S}$.

The phase space Ω is the compact

$$\Omega = \{(S, A, I) \in \mathbb{R}_+^3 \mid S + A + I \leq 1\}. \quad (3.42)$$

Theorem 3.1.1 (see page 75) ensures Ω is a positively invariant set, no matter the value of parameter λ .

	β_A	β_I	δ_A	δ_I	r_0	DFE	EE
λ_1	0.5	0.5	0.1	0.1	2.5	$\times$	$\checkmark$
λ_2	0.5	0.9	0.1	0.1	3.71	$\times$	$\checkmark$
λ_3	0.9	0.5	0.1	0.1	3.31	$\times$	$\checkmark$
λ_4	0.9	0.9	0.1	0.1	4.51	$\times$	$\checkmark$
λ_5	0.5	0.5	0.5	0.1	0.96	$\checkmark$	$\cdot$
λ_6	0.5	0.9	0.5	0.1	1.43	$\times$	$?$
λ_7	0.9	0.5	0.5	0.1	1.27	$\times$	$?$
λ_8	0.9	0.9	0.5	0.1	1.73	$\times$	$\cdot$
λ_9	0.5	0.5	0.1	0.5	1.30	$\times$	$\cdot$
λ_{10}	0.5	0.9	0.1	0.5	1.54	$\times$	$?$
λ_{11}	0.9	0.5	0.1	0.5	2.10	$\times$	$?$
λ_{12}	0.9	0.9	0.1	0.5	2.34	$\times$	$\checkmark$
λ_{13}	0.5	0.5	0.5	0.5	0.50	$\checkmark$	$\cdot$
λ_{14}	0.5	0.9	0.5	0.5	0.59	$\checkmark$	$\cdot$
λ_{15}	0.9	0.5	0.5	0.5	0.81	$\checkmark$	$\cdot$
λ_{16}	0.9	0.9	0.5	0.5	0.90	$\checkmark$	$\cdot$

Table 3.2: Synthesis of the parameter domain Λ . For each parameter, the basic reproduction number r_0 is computed. The columns DFE and EE indicate the asymptotic stability of the equilibrium: the symbol $\checkmark$ means that the equilibrium exists and is Globally Asymptotically Stable (GAS); the symbol $\times$ means that the equilibrium exists, but is not GAS; the symbol $?$ means that the equilibrium exists but the stability is unknown (though Ottaviano, Sensi, and Sottile suggest that EE is stable if $r_0 > 1$, see [60], Conjecture 15 p.15); the symbol $\cdot$ means that the endemic equilibrium does not exist.

We are interested in studying the property φ ="the population is never fully infected and reaches the disease-free equilibrium in at most 56 time units." To see if this property can be satisfied, we will produce a Markov Decision Process $\mathbf{M}$ as described in Section 3.2, which abstracts the behaviors of $\mathcal{S}$: if two controllers Ψ (abstracting the natural control process $\mathbb{Y}$) and Φ (which we can specify) can be designed so that φ is satisfied with probability 1, then we can design a controller $\mathbb{X}$ for $\mathcal{S}$, derived from Φ , so that $\mathcal{S}$ satisfies φ with probability 1.

To this end, we set a tolerance $\rho = 0.05$ and consider the following partition of Ω :

$$\Omega = \begin{cases} R_{e,i} = \{(S, A, I) \in \Omega \mid A + I \in [\frac{5+19(i-1)}{100}, \frac{5+19i}{100})\}, i \in [1..5], \\ R_{df} = \{(S, A, I) \in \Omega \mid A + I < 0.05\}. \end{cases} \quad (3.43)$$

Then, we fix $\tau = 7$ days and consider the timeline discretization $\mathcal{T} = \{k\tau \mid k \in [0..7]\}$,

so that the system may evolve during 8 weeks, and we identify φ as i) the system never enters the most infected region $R_{e,5}$, and ii) the system enters the (almost) disease-free region R_{df} after at most 8 continuous transitions.

To describe the set $\mathcal{D}$ of distributions in $\mathcal{S}$, we consider 21 reset values for λ and $z = (S, A, I)$:

- the set Λ defined in Table 3.2;
- a set $\mathcal{Z} = \{z_{df}, z_{e,1}, z_{e,2}, z_{e,3}, z_{e,4}\}$, where

$$\begin{aligned} - z_{df} &= (\frac{49}{100}, \frac{1}{100}, \frac{1}{100}); \\ - z_{e,i} &= (\frac{86-19(i-1)}{200}, \frac{7+10(i-1)}{100}, \frac{7+9(i-1)}{100}), i \in [1..4]. \end{aligned}$$

Note that for any $i \in [1..4]$, $z_{e,i} \in R_{e,i}$; we purposefully leave $R_{e,5}$ without a reset position since this is the region we want to avoid. Moreover, given two parameter values $\lambda_i = (\beta_A, \beta_I, \delta_A, \delta_I)$ and $\lambda_j = (\beta_A, \beta_I, \delta'_A, \delta'_I)$ which share the same component values β_A, β_I , we denote by $\mathbb{D}_{i,j}$ the uniform distribution on $\{\lambda_i, \lambda_j\}$. Finally, we consider the following set $\mathcal{D}$ of distributions:

$$\mathcal{D} = \mathcal{D}^\Lambda \cup \mathcal{D}^\Omega, \text{ where } \begin{cases} \mathcal{D}^\Lambda = \{\mathbb{D}_{i,j} \mid (\beta_A, \beta_I, \delta_A, \delta_I) = \lambda_i \neq \lambda_j = (\beta_A, \beta_I, \delta'_A, \delta'_I) \in \Lambda\}, \\ \mathcal{D}^\Omega = \{\mathbf{1}_z \mid z \in \mathcal{Z}\}. \end{cases} \quad (3.44)$$

At each time point $t_k \in \mathcal{T}$, a natural controller $\mathbb{Y}$ picks an action which, according to some distribution function $\mathbb{V}$, may change the value of parameter $\lambda \in \Lambda$. We assume $\mathbb{V}$ only affects the components β_A, β_I of λ and mimics a uniform distribution on the set $\{0.1, 0.5\}^2$. Given two parameter values $\lambda, \lambda' \in \Lambda$ such that $\lambda = (\beta_A, \beta_I, \delta_A, \delta_I)$ and $\lambda' = (\beta'_A, \beta'_I, \delta'_A, \delta'_I)$, we have:

$$\begin{aligned} \mathbb{V} : \quad \Lambda^2 &\rightarrow [0, 1] \\ (\lambda, \lambda') &\mapsto \begin{cases} 1, & \text{if } \beta'_A = \beta'_I = 0.5 \text{ and } (\delta_A, \delta_I) = (\delta'_A, \delta'_I) \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.45)$$

First, we split $\mathbb{V}$ into 16 distributions (one for each parameter $\lambda = (\beta_A, \beta_I, \delta_A, \delta_I) \in \Lambda$, see Remark 3.2.11 on page 93):

$$\begin{aligned} \mathbb{V}_\lambda : \Lambda &\rightarrow [0, 1] \\ \lambda' &\mapsto \begin{cases} \frac{1}{4}, & \text{if } \lambda' = (\beta'_A, \beta'_I, \delta_A, \delta_I), \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.46)$$

Unlike other actions $\zeta_{\mathbb{D}}, \mathbb{D} \in \mathcal{D}$ which we will define below, the action of $\mathbb{Y}$ may induce two different transitions in $\mathcal{S}$: given the realisation of a variable $\mathbb{B}$ following a Bernoulli distribution of parameter $b \in [0, 1]$, this action enforces either i) a discrete transition according to $\mathbb{V}_\lambda$ if $\mathbb{B} = 1$, or ii) a loop on the same state if $\mathbb{B} = 0$. The self-loop of case ii) is emulated as a discrete transition on the same parameter λ : indeed, $\mathbb{Y}$ has picked an action, thus the system *must* undergo a transition; this is achieved by “changing” the value of parameter λ to the same value, through the distribution $\mathbf{1}_\lambda \in \mathcal{W}$. Formally, the probability distribution $\tilde{\mathbb{V}}_\lambda$ associated to the discrete transition enforced by $\mathbb{Y}$ is defined as follows:

$$\begin{aligned} \tilde{\mathbb{V}}_\lambda : \Lambda &\rightarrow [0, 1] \\ \lambda &\mapsto b\mathbb{V}_\lambda(\lambda') + (1 - b)\mathbf{1}_\lambda(\lambda') \end{aligned} \quad (3.47)$$

Consequently, we have $\mathcal{V} = \{\tilde{\mathbb{V}}_\lambda \mid \lambda \in \Lambda\}$ to account for the changes enforced by natural controller $\mathbb{Y}$. For each distribution $\tilde{\mathbb{V}}_\lambda \in \mathcal{V}$, we consider a corresponding action $\zeta_{\tilde{\mathbb{V}}_\lambda}$ that may induce a discrete transition according to distribution $\tilde{\mathbb{V}}_\lambda$, and define the set $\mathcal{A}_{\mathcal{V}} = \{\zeta_{\tilde{\mathbb{V}}_\lambda} \mid \tilde{\mathbb{V}}_\lambda \in \mathcal{V}\}$ of actions.

Given a behaviour $\pi \in (\Omega \times \Lambda)^+$ whose last state is (z, λ) , the controller $\mathbb{Y}$ is defined as follows:

$$\begin{aligned} \mathbb{Y} : (\Omega \times \Lambda)^+ &\rightarrow \mathcal{A}_{\mathcal{V}} \\ \pi &\mapsto \zeta_{\tilde{\mathbb{V}}_\lambda}. \end{aligned} \quad (3.48)$$

Recall that $\mathbb{X}$ may pick actions which induce discrete transitions according to some distribution $\mathbb{D}$ in $\mathcal{D}$ or $\mathcal{W}$ (if it chooses $\emptyset$). Each distribution $\mathbb{D} \in \mathcal{D}$ is associated to an action $\zeta_{\mathbb{D}}$; we restrict which action $\zeta_{\mathbb{D}}, \mathbb{D} \in \mathcal{D}^\Lambda$ can be picked by $\mathbb{X}$ based on the current value of parameter λ , and which action $\zeta_{\mathbb{D}}, \mathbb{D} \in \mathcal{D}^\Omega$ can be picked based on the current value of position z ; given a behavior $\pi = s_0 \dots s_k \in (\Omega \times \Lambda)^+$ whose last state is $s_k = (z_k, \lambda_k)$:

- $\forall \mathbb{D}_{i,j} \in \mathcal{D}^\Lambda, \zeta_{\mathbb{D}_{i,j}} \in \mathcal{A}(s_k) \Leftrightarrow \lambda_k \in \{\lambda_i, \lambda_j\}$;
- $\forall \mathbf{1}_z \in \mathcal{D}^\Omega, \zeta_{\mathbf{1}_z} \in \mathcal{A}(s_k) \Leftrightarrow$ the regions R_k and R such that $z_k \in R_k$ and $z \in R$ are neighbors, or such that $R = R_k$.

Finally, we consider the set $\mathcal{A} = \{\zeta_\tau, \emptyset\} \cup \{a_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\} \cup \{\zeta_{\tilde{\mathbb{V}}} \mid \tilde{\mathbb{V}} \in \mathcal{V}\}$. These actions induce a transition in $\mathcal{S}$, as defined in Eq. (1.23) on page 29. The actions ζ_τ , $\zeta_{\tilde{\mathbb{V}}_\lambda}$ and $\emptyset$ are always enabled.

Proposition 3.3.1.

The tuple $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ is a Hybrid Dynamical System, and $\mathbb{Y}$ is a controller for $\mathcal{S}$. $\triangleleft$

We assume the initial state of the system is (z_0, λ_2) , where $z_0 = (0.48, 0.27, 0)$. The initial region $R_0 \in \mathfrak{Q}$ such that $z_0 \in R_0$ is $R_{e,2}$.

3.3.2 Abstraction as a MDP

We now unfold the abstraction procedure on the system we just described in Section 3.3.1.

First, we consider the partition Ω defined in Eq. (3.43) on page 108. This partition is denoted $\mathfrak{Q} = \{R_{df}\} \cup \{R_{e,i} \mid i \in [1..5]\}$, and we denote by $\mathbf{Q} = \mathfrak{Q} \times \Lambda$ the state set of the future MDP $\mathbf{M}$.

Second, we abstract the continuous transition $\xrightarrow{\tau}$ in $\mathcal{S}$ to the setting of MDPs:

$$\begin{aligned} \mathbf{P}_\tau : \quad \mathbf{Q} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), (R_j, \lambda_j)) &\mapsto \begin{cases} p_{i,j}^{\lambda_i}, & \text{if } \lambda_i = \lambda_j, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.49)$$

We also adapt the distributions $\tilde{\mathbb{V}}_{\lambda_i} \in \mathcal{V}$ to the setting of MDPs, by defining the following transition probability distribution function $\mathbb{P}_{\mathbb{V}}$ which encompasses all of them:

$$\begin{aligned} \mathbf{P}_{\mathbb{V}} : \quad \mathbf{Q} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), (R_j, \lambda_j)) &\mapsto \begin{cases} \tilde{\mathbb{V}}_{\lambda_i}(\lambda_j), & \text{if } R_i = R_j \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.50)$$

The distributions $\mathbf{1}_z \in \mathcal{D}^\Omega$, $z \in \Omega$ are adapted as the Dirac distributions $\mathbf{1}_R$, $R \in \mathfrak{Q}$ concentrated on R , where R is such that $z \in R$. Note that there is a one-to-one association between $\mathbf{1}_z$ and $\mathbf{1}_R$, so that we can know which distribution $\mathbf{1}_z$ to pick when deriving the controller $\mathbb{X}$ from Φ , based on $\mathbf{1}_R$.

Third, for each action $a \in \mathcal{A}$ (see Proposition 3.3.1) we define a corresponding action

for $\mathbf{M}$, and we have $\mathbf{A} = \{a_\tau, a_\forall, \emptyset\} \cup \{a_{\mathbb{D}} \mid \mathbb{D} \in \mathcal{D}\}$; the following function $\mathbf{P}$ defines a probability distribution function:

$$\begin{aligned} \mathbf{P} : \quad \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), a, (R_j, \lambda_j)) &\mapsto \begin{cases} \mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\tau, \\ \mathbf{P}_\forall((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\forall, \\ \mathbf{1}_{(R_i, \lambda_i)}(R_j, \lambda_j), & \text{if } a = \emptyset, \\ \mathbb{D}_{i,k}(\lambda_j), & \text{if } a = a_{\mathbb{D}_{i,k}} \text{ and } R_i = R_j, \\ \mathbf{1}_{R_k}(R_j) & \text{if } a = a_{\mathbf{1}_{R_k}} \text{ and } \lambda_i = \lambda_j, \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (3.51)$$

We assume that the actions $a_{\mathbb{D}_{i,k}}$, $a_{\mathbf{1}_{R_k}}$ follow the same rules of availability as their corresponding actions in $\mathcal{S}$ (see Section 3.3.1 on page 110). The others actions a_τ , $a_\forall$ and $\emptyset$ are always enabled.

Finally, given the initial state $(z_0, \lambda_2) \in \Omega \times \Lambda$ and the region $R_{e,2} \in \mathfrak{Q}$, we denote by $\mathbf{I}$ the Dirac distribution concentrated on $(R_{e,2}, \lambda_2)$. As established by Theorem 3.2.21 (see page 104), the MDP $\mathbf{M} = (\mathbf{Q}, \mathbf{P}, \mathbf{A}, \mathbf{I})$ almost- τ -simulates hybrid system $\mathcal{S}$.

3.3.3 Devising controllers

Now that we have built a MDP, we can use it as a proxy of the hybrid system $\mathcal{S}$, thanks to the almost- τ -simulation established by Theorem 3.2.21.

To have a similar setting as hybrid system $\mathcal{S}$, the MDP $\mathbf{M}$ will be augmented with two controllers Ψ (associated to natural controller $\mathbb{Y}$ in $\mathcal{S}$) and Φ (which we will design in order to satisfy φ , and will be used to design $\mathbb{X}$ in $\mathcal{S}$). Similarly to $\mathcal{S}$, we enforce a rotation between these controllers: each controller will be able to pick an action if the length of the input behaviour π is such that $\pi \equiv 1 \pmod{3}$ (in the case of Φ) or $\pi \equiv 2 \pmod{3}$ (in case of Ψ); if the length of π is such that $|\pi| \equiv 0 \pmod{3}$, then the action a_τ is selected for the MDP, thus simulating a continuous transition in $\mathcal{S}$.

First, we define Ψ transparently from the definition of $\mathbb{Y}$, given in Eq. (3.48) on page 110:

$$\begin{aligned} \Psi : \quad \mathbf{Q}^+ &\rightarrow \mathbf{A} \\ \pi &\mapsto a_\forall. \end{aligned} \quad (3.52)$$

We can now design a controller $\Phi : \mathbf{S} \rightarrow \mathbf{A}$ such that the property φ is satisfied with

probability 1 in $\mathbb{M}_\Phi$, the Markov chain induced by Φ and Ψ . This can be done using tools, such as PRISM [41], or by hand if the context is sufficiently simple.

Here, it is relatively easy to design a controller by hand. To simplify the tables and since one action $a_{\mathbb{D}}$ is tied to exactly one distribution $\mathbb{D} \in \mathcal{D}$, we abuse notations and write the distribution $\mathbb{D}$ instead of the action $a_{\mathbb{D}}$. A first controller is given in Table 3.3.

	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$
$R_{e,3}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$
$R_{e,2}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$
$R_{e,1}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$
R_{df}	$\mathbb{D}_{1,13}$	$\mathbb{D}_{2,14}$	$\mathbb{D}_{3,15}$	$\mathbb{D}_{4,16}$	$\mathbb{D}_{5,13}$	$\mathbb{D}_{6,14}$	$\mathbb{D}_{7,15}$	$\mathbb{D}_{8,16}$
	λ_9	λ_{10}	λ_{11}	λ_{12}	λ_{13}	λ_{14}	λ_{15}	λ_{16}
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$	$1_{R_{e,3}}$
$R_{e,3}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$	$1_{R_{e,2}}$
$R_{e,2}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$	$1_{R_{e,1}}$
$R_{e,1}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$	$1_{R_{df}}$
R_{df}	$\mathbb{D}_{9,13}$	$\mathbb{D}_{10,14}$	$\mathbb{D}_{11,15}$	$\mathbb{D}_{12,16}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 3.3: A first controller Φ for $\mathbf{M}$. Since the game is lost if $\mathbf{M}$ enters $(R_{e,5}, \lambda)$ for any $\lambda \in \Lambda$, we denote by $\times$ the fact that no action is picked in those states.

The controller proposed in Table 3.3 is naive, and takes advantage of the fact that actions are costless: the simplest strategy to guide $\mathbf{M}$ towards R_{df} is to sequentially reset the position to the nearest region between the current one and R_{df} .

A major issue of our setting is that one cannot ensure that $\mathbf{M}$ will progress towards R_{df} between two timepoints: indeed, in the worst case, $\mathbb{X}$ can change the position of $\mathbf{M}$ to its neighbor region (e.g. $R_{e,2}$ if the current region is $R_{e,3}$) through a discrete transition, but the following continuous transition, if $\lambda \notin \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$, may move $\mathbf{M}$ back to the current region, or even to a region further from R_{df} than $R_{e,2}$ ⁵.

This can be solved by adding new actions that allow $\mathbf{M}$ to jump over a neighbor region toward the next-after (e.g. from $R_{e,3}$ to $R_{e,1}$) or by forcing a reset of dynamics

⁵Though it is easy to prove that R_{df} is always reachable from any state (R_i, λ_i) in $\mathbf{M}$, notably because Φ can always select $a_{\mathbb{D}_{i,j}}$ for some $\lambda_j \in \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$. Yet, even though R_{df} can always be reached within a *finite* amount of steps, this length cannot be upper-bounded.

to one that will make $\mathbf{M}$ move towards R_{df} during the next continuous transition, no matter the change induced by Ψ . Given a parameter $\lambda \in \Lambda$, we denote by $\mathbf{1}_\lambda$ the Dirac distribution concentrated on λ , and denote by $a_{\mathbf{1}_\lambda}$ the associated action. Since Φ may only change the values of δ_A, δ_I , the action $a_{\mathbf{1}_\lambda}$ is enabled in state (R, λ) (resp. (z, λ)) iff $\lambda = (\beta_A, \beta_I, \delta_A, \delta_I)$ and $\lambda' = (\beta_A, \beta_I, \delta'_A, \delta'_I)$. Then, we can build the controller given by Table 3.4.

	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$
$R_{e,3}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$
$R_{e,2}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$
$R_{e,1}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$
R_{df}	$\mathbb{D}_{1,13}$	$\mathbb{D}_{2,14}$	$\mathbb{D}_{3,15}$	$\mathbb{D}_{4,16}$	$\mathbb{D}_{5,13}$	$\mathbb{D}_{6,14}$	$\mathbb{D}_{7,15}$	$\mathbb{D}_{8,16}$
	λ_9	λ_{10}	λ_{11}	λ_{12}	λ_{13}	λ_{14}	λ_{15}	λ_{16}
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{R_{e,3}}$	$\mathbf{1}_{R_{e,3}}$	$\mathbf{1}_{R_{e,3}}$	$\mathbf{1}_{R_{e,3}}$
$R_{e,3}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{R_{e,2}}$	$\mathbf{1}_{R_{e,2}}$	$\mathbf{1}_{R_{e,2}}$	$\mathbf{1}_{R_{e,2}}$
$R_{e,2}$	$\mathbf{1}_{\lambda_{13}}$	$\mathbf{1}_{\lambda_{14}}$	$\mathbf{1}_{\lambda_{15}}$	$\mathbf{1}_{\lambda_{16}}$	$\mathbf{1}_{R_{e,1}}$	$\mathbf{1}_{R_{e,1}}$	$\mathbf{1}_{R_{e,1}}$	$\mathbf{1}_{R_{e,1}}$
$R_{e,1}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$	$\mathbf{1}_{R_{df}}$
R_{df}	$\mathbb{D}_{9,13}$	$\mathbb{D}_{10,14}$	$\mathbb{D}_{11,15}$	$\mathbb{D}_{12,16}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 3.4: Controller Φ ensuring $\mathbf{M}$ will reach R_{df} within 14 transitions, with probability 1, if the first transition in $\mathbf{M}$ does not dissatisfies φ . Again, since the game is lost if $\mathbf{M}$ enters $(R_{e,5}, \lambda)$ for any $\lambda \in \Lambda$, we denote by $\times$ the fact that no action is picked in those states.

Using the controller defined by Table 3.4, $\mathbf{M}$ will reach R_{df} within 14 transitions if the first transition (corresponding to a continuous transition in $\mathcal{S}$) does not move it to $(R_{e,5}, \lambda_2)$. Indeed, the starting region of all behaviours in $\mathbf{M}$ will be $(R_{e,2}, \lambda_2)$, and the first action picked for $\mathbf{M}$ is a_τ ; in Table 3.5, we present the computed (through SMC techniques) transition probabilities between regions in $\mathfrak{Q}$, for $\lambda = \lambda_2 = (0.5, 0.9, 0.1, 0.1)$. This table shows that during the simulations, no trajectory of duration τ starting in $R_{e,2}$ stopped in $R_{e,4}$ nor $R_{e,5}$: we can deduce that, even if the ratio of trajectories starting in $R_{e,2}$ and stopping in $R_{e,4}$ or $R_{e,5}$ may not actually be 0 (since we cannot estimate the exact forward trajectories induced by f), it is at least very small (especially regarding $R_{e,5}$), thus the probability of moving from $(R_{e,2}, \lambda_2)$ to $(R_{e,5}, \lambda_2)$ is also very small (but

it may be possible).

	R_{df}	$R_{e,1}$	$R_{e,2}$	$R_{e,3}$	$R_{e,4}$	$R_{e,5}$
R_{df}	0.67	0.33	0	0	0	0
$R_{e,1}$	0.01	0.74	0.25	0	0	0
$R_{e,2}$	0	0.01	0.76	0.23	0	0
$R_{e,3}$	0	0	0.03	0.84	0.13	0
$R_{e,4}$	0	0	0	0.09	0.90	0.01
$R_{e,5}$	0	0	0	0	0.42	0.58

Table 3.5: Transition probabilities between regions if $\lambda = (0.5, 0.9, 0.1, 0.1)$. These values are obtained through SMC techniques by computing the ratio of trajectories that start and end in the various regions, with $N = 874$ tries per pair of regions; as such, they are not the true ratios of accepting initial conditions (as defined in Eq. (3.15) on page 88), but estimated values.

Assuming this first transition did not move $\mathbf{M}$ to $(R_{e,5}, \lambda_2)$, Table 3.5 shows that in the worst case, $\mathbf{M}$ will move towards $(R_{e,4}, \lambda_2)$ (with probability ~ 0), towards $(R_{e,3}, \lambda_2)$ in some cases (with probability ~ 0.23), towards $(R_{e,2}, \lambda_2)$ in most cases (with probability ~ 0.76) and towards $(R_{e,1}, \lambda_2)$ in the best cases (with probability ~ 0.01); of course, we assume the worst, i.e. that the state of $\mathbf{M}$ changes to $(R_{e,4}, \lambda_2)$. Then, Φ selects the action $a_{\lambda_{14}}$, which changes the state of $\mathbf{M}$ to $(R_{e,4}, \lambda_{14})$. The action $a_{\mathbb{V}}$ is then selected by Ψ , and changes again the value of the parameter λ , so that the value is now $\lambda \in \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$ after the third transition in any behaviour.

Since $\forall \lambda \in \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$, the basic reproduction number $r_0 < 0$, we know that $\forall R_{e,i}, R_j \in \mathfrak{Q}, \mathbf{P}_\tau((R_{e,i}, \lambda), (R_j, \lambda)) > 0 \Leftrightarrow R_j = R_{e,j}$ and $j \leq i$, or $R_j = R_{df}$. Intuitively, it means that during the transition induced by a_τ , $\mathbf{M}$ will either stay in the same region, or drift towards R_{df} through the regions $R_{e,j}$ between the current region $R_{e,i}$ and R_{df} .

The fourth transition is induced by a_τ , and the state of $\mathbf{M}$ changes to $(R_{e,4}, \lambda)$ at worst (but could be e.g. $(R_{e,2}, \lambda)$). The fifth transition is controlled by Φ , which selects $a_{1_{R_{e,3}}}$, then Ψ chooses $a_{\mathbb{V}}$ and the state of $\mathbf{M}$ after the sixth transition is $(R_{e,3}, \lambda')$, with $\lambda' \in \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$. The seventh transition is induced by a_τ , and the state of $\mathbf{M}$ changes to $(R_{e,3}, \lambda')$ at worst (but could be e.g. $(R_{e,1}, \lambda')$). Then Φ picks $a_{1_{R_{e,2}}}$, then Ψ selects $a_{\mathbb{V}}$ and the state of $\mathbf{M}$ after the ninth transition is $(R_{e,2}, \lambda'')$, with $\lambda'' \in \{\lambda_{13}, \lambda_{14}, \lambda_{15}, \lambda_{16}\}$. This rotation then happens once more, and after the twelfth transition, $\mathbf{M}$ is in state $(R_{e,1}, \lambda''')$; the thirteenth transition is induced by a_τ , and in the worst cases the state of $\mathbf{M}$ is now $(R_{e,1}, \lambda''')$: Φ now picks $a_{1_{R_{df}}}$ and the system moves towards (R_{df}, λ''') , and

property φ has been satisfied.

Since the actions in $\mathbf{M}$ are transparently defined based on the actions in $\mathcal{S}$, we can completely define the controller $\mathbb{X}$ in $\mathcal{S}$ using the definition of Φ in Table 3.4 by identifying $\zeta_{1_{z_{e,i}}}$ to $a_{1_{R_{e,i}}}$; this controller is defined in Table 3.6. We can now use this controller to

	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$
$R_{e,3}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$
$R_{e,2}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$
$R_{e,1}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$
R_{df}	$\mathbb{D}_{1,13}$	$\mathbb{D}_{2,14}$	$\mathbb{D}_{3,15}$	$\mathbb{D}_{4,16}$	$\mathbb{D}_{5,13}$	$\mathbb{D}_{6,14}$	$\mathbb{D}_{7,15}$	$\mathbb{D}_{8,16}$
	λ_9	λ_{10}	λ_{11}	λ_{12}	λ_{13}	λ_{14}	λ_{15}	λ_{16}
$R_{e,5}$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$R_{e,4}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{z_{e,3}}$	$1_{z_{e,3}}$	$1_{z_{e,3}}$	$1_{z_{e,3}}$
$R_{e,3}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{z_{e,2}}$	$1_{z_{e,2}}$	$1_{z_{e,2}}$	$1_{z_{e,2}}$
$R_{e,2}$	$1_{\lambda_{13}}$	$1_{\lambda_{14}}$	$1_{\lambda_{15}}$	$1_{\lambda_{16}}$	$1_{z_{e,1}}$	$1_{z_{e,1}}$	$1_{z_{e,1}}$	$1_{z_{e,1}}$
$R_{e,1}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$	$1_{z_{df}}$
R_{df}	$\mathbb{D}_{9,13}$	$\mathbb{D}_{10,14}$	$\mathbb{D}_{11,15}$	$\mathbb{D}_{12,16}$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

Table 3.6: Controller $\mathbb{X}$ ensuring $\mathcal{S}$ will reach R_{df} within 14 transitions, with probability 1. Again, since the game is lost if $\mathcal{S}$ enters $(R_{e,5}, \lambda)$ for any $\lambda \in \Lambda$, we denote by $\times$ the fact that no action is picked in those states.

run simulations of $\mathcal{S}$, two of which are depicted in Fig. 3.12.

In Fig. 3.12a, the trajectory for system $\mathcal{S}$ under controllers $\mathbb{X}$ and $\mathbb{Y}$ is depicted with a solid line; the trajectory for $\mathcal{S}$ under controller $\mathbb{Y}$ only is depicted with a dashed line. In this trajectory, the system undergoes a change of parameter at time point $t = 7$ days, as we see the two (solid and dashed) sets of trajectories (for S , A , I respectively) split at this point, showing a change of dynamics between the run without $\mathbb{X}$ (dashed line) and a run with $\mathbb{X}$ (solid line). Then, at time point $t = 14$ days, a reset of position occurs, hence the discontinuities in the solid trajectories. Finally, as the system $\mathcal{S}$ is already in region R_{df} , $\mathbb{X}$ just waits and no changes occurs anymore: property φ is indeed satisfied. In Fig. 3.12b, we show another controller with which changes are more progressive, with a 0.1-increment on δ_A, δ_I at each step, and no position reset can occur; this controller induces a similar, though slower to achieve, result as Fig. 3.12a. While this controller does not fit our initial setting (as it uses distributions outside of $\mathcal{D}$), it shows that depending on the policy,

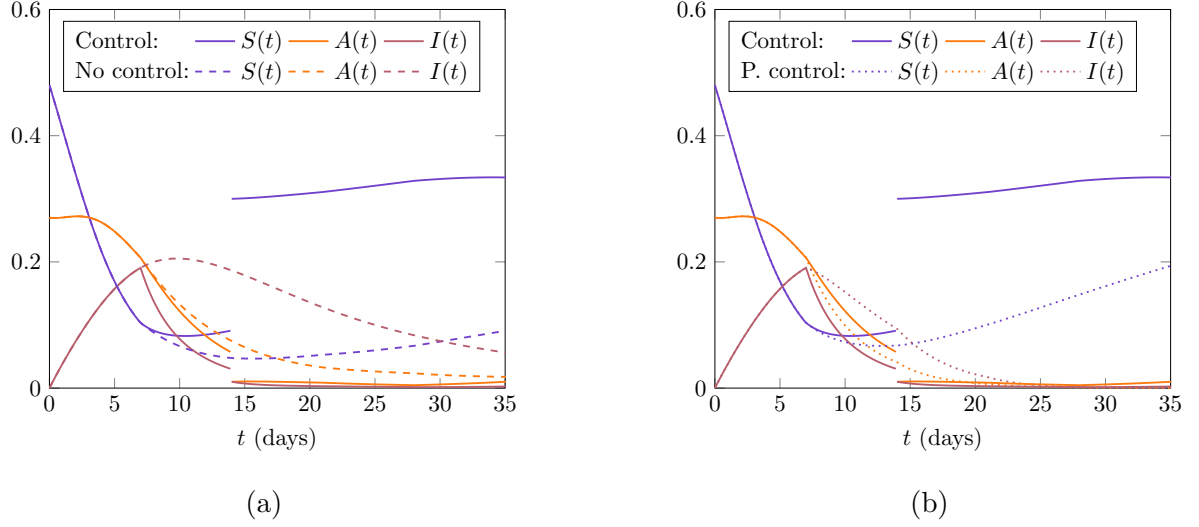


Figure 3.12: Two runs for $\mathcal{S}$ under controllers $\mathbb{Y}$ and two different versions of $\mathbb{X}$. The purple lines denote the trajectory of variable S ; the orange lines denote the trajectory of variable A ; the red lines denote the trajectory of variable I . A trajectory for which only the controller $\mathbb{Y}$ operates is depicted with a dashed line. The run in which the controller $\mathbb{X}$ (defined in Table 3.6) is added to perform other discrete changes is depicted with a solid line, with the exact same changes induced by $\mathbb{Y}$. In Fig. 3.12a, a second, more progressive controller is used, which may only increment the values of δ_A, δ_I by a step 0.1, leading to more progressive changes, and cannot reset the position of the system; its trajectory is depicted with a dotted line.

several strategies may lead to similar results.

CONCLUSION

In this thesis, we have presented new methods for the probabilistic and statistical study of dynamical systems described by Ordinary Differential Equations.

Our first main contribution was the adaptation of Statistical Model Checking [79] to the setting of ODEs, in order to estimate the value of a reward obtained by a given model. We started by a careful study of the integration method used to approximate the solutions to an ODE, and then showed that the produced error, which can be arbitrarily bounded for any given simulation, can also be bounded on the whole parameter space as well as the whole state space of the model, provided the set of interest is compact; finally, we use these bounds to construct auxiliary reward functions, which are then used to estimate the (initial) reward that would be obtained by a given model. This result is of clear importance since, to the best of our knowledge, this formal proof was left to be done even though SMC itself was already in use in the context of ODEs [47]. We provided a step-by-step explanation of this technique along with the proof, but it is worth mentioning that this method is method- and model-agnostic: it can be leveraged in the case of any integration method for which the approximation error can be bounded w.r.t. the variable of interest, and for any ODE which meets the requirements of that method.

Still, ODEs by themselves proved not powerful enough to represent complex systems, as one cannot interact with it: once a parameter and an initial condition have been set, the model is isolated. This motivates the formalization of more complex models, whose dynamics can be changed during a simulation. While Hybrid Automata [4, 73] proved powerful enough to meet this criterion, they rapidly proved *too* powerful to be of use for concrete application, since reachability is undecidable as soon as e.g. variable reset is relaxed, or if the ODEs at use are nonlinear [31]. Hence, one must find a proper abstraction, i.e. one for which properties can be decided while retaining enough information about the hybrid system to transfer these properties back to it, in order to study hybrid systems. In this regard, the concept of simulation relation is crucial as it allows the study of a simpler, finite-state system for which properties such as reachability or safety can be assessed, while capturing all behaviors of the hybrid system.

In this thesis, we abstracted hybrid systems as Markov processes, and used them as a proxy of the original hybrid systems, so as to derive specific properties and design controllers that ensure the satisfaction of properties. Again, this result is of importance since Hybrid Automata are a topic of great interest in the modelization community, as they provide the power of ODEs in terms of representation of real phenomena while keeping the possibility of interacting with it and have several dynamics to choose from at any given time. While this abstraction does not produce an equivalence w.r.t. original system, hence restricting the properties that can be verified, we believe this is a first step in an interesting direction.

The main difficulty of the simulation relation approach is twofold: i) proving said relation of the original model by its abstraction; ii) ensuring the abstraction does not introduce (too many) new behaviors, so that the approximation error of the abstraction can be estimated. While i) may be readily proved by the construction of the abstraction, ii) is more challenging to obtain, and is the main challenge faced by computer scientists nowadays. A major direction of improvement would then be to refine the abstraction process in order to properly restrict the behaviours of the Markov process, so that it matches more closely that of the hybrid system. Although bisimulation is likely not achievable without allowing arbitrary resets in the hybrid system, we believe that a proper refinement of the abstraction process may be achieved. The works of Banse et al. [11] are quite encouraging from this point of view, even though their abstraction may appear difficult to grasp from someone unfamiliar with formal methods.

PROOFS FOR CHAPTER 1

A.1 Proof of Lemma 1.3.2

Lemma 1.3.2.

Let $\mathcal{S} = (f, \Omega, \Lambda, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid dynamical system, as defined in Definition 1.3.1.

For any $(z_0, \lambda) \in \Omega \times \Lambda$, the ODE induced by Eq. (1.18) is well-formed and the Cauchy problem induced by f , λ and z_0 admits a single continuous solution on $[t_i, t_j]$ for any $t_i < t_j \in \mathcal{T}$. ◁

Proof of Lemma 1.3.2. By definition of Λ , the ODE induced by f and any parameter $\lambda \in \Lambda$ is well-written. Moreover, Ω is a compact subset of $\mathbb{R}^n$, hence a manifold.

Then, considering $f : \Omega \times \mathbb{R} \rightarrow \Omega$ is $\mathbf{C}^1$ w.r.t. function $z(\cdot)$, Chillingworth' Theorem (see [61], Theorem 7.6 p.100) states that the Cauchy problem induced by f , λ and z_0 admits a unique solution on Ω , that is defined for any $t \in \mathbb{R}$. ◻

PROOFS FOR CHAPTER 2

B.1 Proof of Lemma 2.1.12

Lemma 2.1.12 (Uniform bound for the global approximation error w.r.t. λ).

Let $T > 0$ be a duration for the simulations. Let n be a positive integer. Let $f : \mathbb{R}^+ \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining an ODE in the form of Eq. (1.6). Let $\Lambda \subset \mathbb{R}^m$ be a compact parameter space for f , then let $\lambda^* \in \mathring{\Lambda}$ be a parameter. Let $\Omega \subseteq \mathbb{R}^n$ be the phase space of f and parameter space Λ , then let $z_0 \in \Omega$ be an initial condition. Let $\rho > 0$ be a radius, such that the ball $\mathcal{B}_{\lambda^*} = \{\lambda \in \mathbb{R}^m \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\} \subseteq \Lambda$.

Let $(h_i)_{i \in \mathbb{N}} \in \mathbb{R}^+$ be a sequence of integration steps, such that $\lim_{i \rightarrow \infty} h_i = 0$; for each h_i , we denote $\mathcal{T}_i = \{t_k \in [0, T) \mid \exists k \in \mathbb{N}, t_k = k \times h_i\}$ the discretization it induces. For any $\varepsilon > 0$, there exists $i^* \in \mathbb{N}$ such that

$$\forall i > i^*, \forall \lambda \in \mathcal{B}_{\lambda^*}, \quad \varepsilon(\mathcal{T}_i, \lambda) < \varepsilon. \quad (2.13)$$

◁

Proof of Lemma 2.1.12. First, we fix $\lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}$, and we consider the approximate solutions $\tilde{\mathbf{z}}^{\lambda_1}, \tilde{\mathbf{z}}^{\lambda_2}$ to the ODEs induced by f and λ_1, λ_2 respectively and starting from z_0 :

$$\begin{cases} z_0^{\lambda_1} &= z_0, \\ \tilde{\mathbf{z}}_{k+1}^{\lambda_1} &= \tilde{\mathbf{z}}_k^{\lambda_1} + h \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_1}, \lambda_1, h), \end{cases} \quad \begin{cases} z_0^{\lambda_2} &= z_0, \\ \tilde{\mathbf{z}}_{k+1}^{\lambda_2} &= \tilde{\mathbf{z}}_k^{\lambda_2} + h \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_2}, \lambda_2, h). \end{cases} \quad (\text{B.1})$$

We now recall the notion of (local) Lipschitz continuity of a function:

Definition B.1.1 (Lipschitz continuity (see [61], Definition 2 p.71)). Let E be an open subset of $\mathbb{R}^n$. A function $f : \mathbb{R} \times E \rightarrow \mathbb{R}^n$ is said to satisfy a Lipschitz condition on E if

there exists a constant $\kappa > 0$ such that

$$\forall x, y \in E, \|f(x) - f(y)\|_{\mathbb{R}^n} \leq \kappa \|x - y\|_{\mathbb{R}^n}. \quad (\text{B.2})$$

In that case, we say that f is Lipschitz continuous.

Furthermore, we say that f is *locally* Lipschitz continuous if for every point $x_0 \in E$, there exists a neighborhood $U \subset E$ of x such that f restricted to U is Lipschitz continuous. $\triangleleft$

The proof of Lemma 2.1.12 can be derived from the following theorem:

Theorem B.1.2 (Stability with respect to consistency error).

Let J be a positive integer, then let $\mathcal{T}_J$ be the set of $J + 1$ first timesteps in $\mathcal{T}$. Let $\mathcal{F}$ define an approximation method (see Eq. (1.11)).

If $\mathcal{F}$ is κ_1 -Lipschitz w.r.t. λ and κ_2 -Lipschitz continuous w.r.t. z , then the approximation method is stable w.r.t. the consistency error, i.e. there exists $\mathcal{K} > 0$ such that

$$\forall \lambda_1, \lambda_2 \in \mathcal{B}(\lambda^*, \rho), \max_{0 \leq k \leq J} |\varepsilon^1(\lambda_1, t_k) - \varepsilon^1(\lambda_2, t_k)| \leq \mathcal{K} \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m}. \quad (\text{B.3})$$

$\triangleleft$

Note that this notion of stability is relative to the method: it does not imply stability of the studied system itself.

Remark B.1.3. In the case of a Runge-Kutta method, Lipschitz continuity of $\mathcal{F}$ is ensured when f is Lipschitz continuous. In our case, since f is $\mathbf{C}^1$ on the compact $\Omega \times \Lambda$, its first derivative is continuous, thus it is bounded and f is Lipschitz continuous on $\Omega \times \Lambda$. Then, Lemma 1 p.241 in [19] ensures that $\mathcal{F}$ is Lipschitz-continuous w.r.t. Λ . $\blacktriangleleft$

Remark B.1.4. In the case of other approximation methods whose integration step is not constant, other ways may be leveraged to find a condition for the stability of the method (see [19], Chapter IX p.251). $\blacktriangleleft$

Proof of Theorem B.1.2. By assumption, $\mathcal{F}$ is κ_1 -Lipschitz continuous w.r.t. λ :

$$\forall t, z, h \in \mathbb{R}, \forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, \|\mathcal{F}(t, z, \lambda_1, h) - \mathcal{F}(t, z, \lambda_2, h)\|_{\mathbb{R}^n} \leq \kappa_1 \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m}. \quad (\text{B.4})$$

It follows that

$$\begin{aligned}
\|\tilde{\mathbf{z}}_{k+1}^{\lambda_1} - \tilde{\mathbf{z}}_{k+1}^{\lambda_2}\|_{\mathbb{R}^n} &\leq \|\tilde{\mathbf{z}}_k^{\lambda_1} - \tilde{\mathbf{z}}_k^{\lambda_2}\|_{\mathbb{R}^n} + h\|\mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_1}, \lambda_1, h) - \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_2}, \lambda_2, h)\|_{\mathbb{R}^n} \\
&\leq \|\tilde{\mathbf{z}}_k^{\lambda_1} - \tilde{\mathbf{z}}_k^{\lambda_2}\|_{\mathbb{R}^n} + h\|\mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_1}, \lambda_1, h) - \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_1}, \lambda_2, h)\|_{\mathbb{R}^n} \\
&\quad + h\|\mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_1}, \lambda_2, h) - \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{\lambda_2}, \lambda_2, h)\|_{\mathbb{R}^n} \\
&\leq (1 + h\kappa_2)\|\tilde{\mathbf{z}}_k^{\lambda_1} - \tilde{\mathbf{z}}_k^{\lambda_2}\|_{\mathbb{R}^n} + h\kappa_1\|\lambda_1 - \lambda_2\|_{\mathbb{R}^m},
\end{aligned} \tag{B.5}$$

for $0 \leq k \leq J$. We write $\|\tilde{\mathbf{z}}_k^{\lambda_1} - \tilde{\mathbf{z}}_k^{\lambda_2}\|_{\mathbb{R}^n} = \Delta_{\tilde{\mathbf{z}},k}$ and $\|\lambda_1 - \lambda_2\|_{\mathbb{R}^m} = \Delta_\lambda$, and we get

$$\Delta_{\tilde{\mathbf{z}},k+1} \leq (1 + h\kappa_2)\Delta_{\tilde{\mathbf{z}},k} + h\kappa_1\Delta_\lambda. \tag{B.6}$$

Applying the discrete Gronwall Lemma (see for instance [19], Lemma 1 p229), we deduce

$$\max_{0 \leq k \leq J} \Delta_{\tilde{\mathbf{z}},k} \leq e^{\kappa_2 T} \left(\Delta_{\tilde{\mathbf{z}},0} + \sum_{0 \leq j \leq k-1} h\kappa_1 \Delta_\lambda \right)$$

which leads to

$$\max_{0 \leq k \leq J} \|\tilde{\mathbf{z}}_k^{\lambda_1} - \tilde{\mathbf{z}}_k^{\lambda_2}\|_{\mathbb{R}^n} \leq e^{\kappa_2 T} T \kappa_1 \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m},$$

since $z_0^{\lambda_1} = z_0^{\lambda_2} = x_0$ and $hJ = T$.

Furthermore, it has been proved (see [61], Section 2.3 p.79) that if $\mathcal{F}$ is Lipschitz continuous w.r.t. λ , then the exact solution $\mathbf{z}^\lambda$ is also Lipschitz continuous w.r.t. λ that is, there exists $\kappa_3 > 0$ such that

$$\forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, \forall t \in [0, T], \|\mathbf{z}^{\lambda_1}(t) - \mathbf{z}^{\lambda_2}(t)\|_{\mathbb{R}^n} \leq \kappa_3 \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m}. \tag{B.7}$$

Finally, we have

$$\begin{aligned}
|\varepsilon^1(t_k, \lambda_1) - \varepsilon^1(t_k, \lambda_2)| &\leq \|\mathbf{z}^{\lambda_1}(\tau_k) - \tilde{\mathbf{z}}^{\lambda_1}(\tau_k) - \mathbf{z}^{\lambda_2}(\tau_k) - \tilde{\mathbf{z}}^{\lambda_2}(\tau_k)\|_{\mathbb{R}^n} \\
&\leq \|\mathbf{z}^{\lambda_1}(\tau_k) - \mathbf{z}^{\lambda_2}(\tau_k)\|_{\mathbb{R}^n} + \|\tilde{\mathbf{z}}^{\lambda_1}(\tau_k) - \tilde{\mathbf{z}}^{\lambda_2}(\tau_k)\|_{\mathbb{R}^n} \\
&\leq \mathcal{K} \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m},
\end{aligned} \tag{B.8}$$

with $\mathcal{K} = \kappa_3 + T\kappa_1 e^{\kappa_2 T}$, which completes the proof of Theorem B.1.2. $\square$

It remains to show that Theorem B.1.2 implies Lemma 2.1.12.

Let $(h_i)_{i \geq 0}$ be a sequence of discretization steps such that $\lim_{i \rightarrow \infty} h_i = 0$. For each h_i , we denote the discretization of the interval $[0, T]$ by $\mathcal{T}_i = \{t_k \in [0, T] \mid \exists k, t_k = k \times h_i\}$.

Since the approximation method given by Eq. (1.11) is assumed to be convergent, each function $\varepsilon(\mathcal{T}_i, \lambda)$ of global approximation error relative to λ and defined in Eq. (1.14) is pointwise convergent to 0.

Furthermore, we recall that $\mathcal{F}$ is Lipschitz continuous w.r.t. $\lambda \in \Lambda$. Hence, Theorem B.1.2 implies that the functions $(\varepsilon(h_i \cdot))_{i \geq 0}$ of global approximation error $\varepsilon(h_i, \lambda)$ relative to λ are also Lipschitz continuous, with uniform Lipschitz constant $\mathcal{K}$:

$$|\varepsilon(\mathcal{T}_i, \lambda_1) - \varepsilon(\mathcal{T}_i, \lambda_2)| \leq \mathcal{K} \|\lambda_1 - \lambda_2\|_{\mathbb{R}^m}, \quad \forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, \quad \forall i \in \mathbb{N}. \quad (\text{B.9})$$

Consequently, the functions $(\varepsilon(\mathcal{T}_i, \cdot))_{i \geq 0}$ are uniformly equicontinuous. Hence, Arzelà–Ascoli Theorem [22] implies that the sequence $(\varepsilon(\mathcal{T}_i, \cdot))_{i \geq 0}$ converges uniformly to 0 on $\mathcal{B}_{\lambda^*}$, and for any $\varepsilon > 0$, there exists $i^* \in \mathbb{N}$, such that $\forall i \geq i^*, \forall \lambda \in \mathcal{B}_{\lambda^*}, \varepsilon(\mathcal{T}_i, \lambda) < \varepsilon$, and Lemma 2.1.12 is proved. $\square$

B.2 Proof of Theorem 2.1.15

Theorem 2.1.15 (Stability of the approximation w.r.t. parameter λ).

Let $f : \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ be a function defining a parametric ODE of variable $z(t)$, then let z_0 be an initial condition for z . Let $\rho > 0$ be a radius and $\varepsilon > 0$ be the target integration error for the ODE integration. Let $\lambda^* \in \mathring{\Lambda}$ be a parameter to evaluate, such that $\mathcal{B}_{\lambda^*} = \{\lambda \in \mathbb{R}^m \mid \|\lambda^* - \lambda\|_{\mathbb{R}^m} \leq \rho\} \subseteq \Lambda$. Let $\theta \in (0, 1)$ be the risk and $\alpha > 0$ be the precision for the Monte-Carlo procedure. Let $r : \Lambda \rightarrow \mathbb{R}$ be a bounded reward function on Λ ; we consider the greatest (resp. lowest) value U (resp. L) of $r(\cdot)$ on $\mathcal{B}_{\lambda^*}$.

Consider the hidden risk $\xi = 1 - \sqrt{1 - \theta}$. The auxiliary rewards $r_{-\varepsilon}$ and $r_{+\varepsilon}$ satisfy

$$\mathbb{P}(r_- \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{-\varepsilon} + \alpha]) \geq 1 - \xi, \quad \mathbb{P}(r_+ \in [\hat{r}_{+\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \xi, \quad (2.27)$$

where the estimators $\hat{r}_{-\varepsilon} = \hat{r}_{-\varepsilon}(\lambda^*)$ and $\hat{r}_{+\varepsilon} = \hat{r}_{+\varepsilon}(\lambda^*)$ are independently computed after performing, for each estimator, a number $N' = \frac{\log(2/\xi)}{2\alpha^2} (U - L)^2$ of simulations induced by f, z_0 and a parameter λ sampled in $\mathcal{B}_{\lambda^*}$.

Furthermore, there exists $\varepsilon^* > 0$ sufficiently small, such that for any tolerance $\varepsilon < \varepsilon^*$ the following statements hold:

- the expected value $r = \mathbb{E}_{\mathcal{B}_{\lambda^*}}[r]$ of reward function $r(\cdot)$ on $\mathcal{B}_{\lambda^*}$ satisfies

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \theta; \quad (2.28)$$

-
- the distance between $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ satisfies

$$\mathbb{P}(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha) \geq 1 - \theta. \quad (2.29)$$

◁

Proof of Theorem 2.1.15. The first result of this theorem, that is Eq. (2.27) which establishes a confidence interval for the expected value of a reward function $r(\lambda)$ in a neighborhood of λ^* , is implied by Eq. (2.6). Hence, the proof here will focus on Eq. (2.28) and Eq. (2.29), that is establishing a confidence interval for $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r(\cdot)]$ using the Monte-Carlo estimators $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ of $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r_{-\varepsilon}]$ and $\mathbb{E}_{\mathcal{B}_{\lambda^*}}[r_{+\varepsilon}]$, respectively.

The independence of the samples used to determine the estimators $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ guarantees that

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) = \mathbb{P}(r \geq \hat{r}_{-\varepsilon} - \alpha) \times \mathbb{P}(r \leq \hat{r}_{+\varepsilon} + \alpha).$$

Using Eq. (2.28), we have $\mathbb{P}(r_{-\varepsilon} \geq \hat{r}_{-\varepsilon} - \alpha) \geq 1 - \xi$. Next, Definition 2.1.13 (see page 48) implies $\mathbb{P}(r \geq \hat{r}_{-\varepsilon} - \alpha) \geq \mathbb{P}(r_{-\varepsilon} \geq \hat{r}_{-\varepsilon} - \alpha) \geq 1 - \xi$. Similarly, we have $\mathbb{P}(r \leq \hat{r}_{+\varepsilon} + \alpha) \geq 1 - \xi$, and finally $\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq (1 - \xi)^2 = 1 - \theta$, by independence of $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ and since $\xi = 1 - \sqrt{1 - \theta}$.

Since $r_{-\varepsilon}$ and $r_{+\varepsilon}$ are continuous w.r.t. ε by construction (see Definition 2.1.13), there exists $\varepsilon^* > 0$ such that $|r_{-\varepsilon} - r_{+\varepsilon}| \leq \alpha$, for $\varepsilon < \varepsilon^*$.

Next, we write

$$|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq |\hat{r}_{-\varepsilon} - r_{-\varepsilon}| + |\hat{r}_{+\varepsilon} - r_{+\varepsilon}| + |r_{-\varepsilon} - r_{+\varepsilon}|, \quad (\text{B.10})$$

hence we have, for $\varepsilon < \varepsilon^*$:

$$\begin{aligned} \mathbb{P}(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha) &\geq \mathbb{P}(|\hat{r}_{-\varepsilon} - r_{-\varepsilon}| \leq \alpha) \times \mathbb{P}(|\hat{r}_{+\varepsilon} - r_{+\varepsilon}| \leq \alpha) \times \mathbb{P}(|r_{-\varepsilon} - r_{+\varepsilon}| \leq \alpha) \\ &\geq (1 - \xi)^2 \times 1 = 1 - \theta. \end{aligned} \quad (\text{B.11})$$

In parallel, Lemma 2.1.12 guarantees that a sufficiently small value exists for h such that the global stability error can be uniformly bounded on $\mathcal{B}_{\lambda^*}$ by ε , which completes the proof. □

B.3 Proof of Lemma 2.2.5

Lemma 2.2.5 (Uniform bound for the global approximation error w.r.t. variations of the initial condition).

Let $T > 0$ be a duration for the simulations. Let n be a positive integer. Let $f : \mathbb{R}^+ \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function defining an ODE in the form of Eq. (1.5). Let $\Omega \subseteq \mathbb{R}^n$ be the (compact) phase space of f , then let $z_0^* \in \Omega$ be an initial condition. Let $\rho > 0$ be a radius, such that the ball $\mathcal{B}_{z_0^*} = \{z_0 \in \mathbb{R}^n \mid \|z_0^* - z_0\|_{\mathbb{R}^n} \leq \rho\} \subseteq \Omega$.

Let $(h_i)_{i \in \mathbb{N}} \in \mathbb{R}_+$ be a sequence of integration steps, such that $\lim_{i \rightarrow \infty} h_i = 0$; for each h_i , we denote $\mathcal{T}_i = \{t_k \in [0, T) \mid \exists i \in \mathbb{N}, t_k = k \times h_i\}$ the discretization it induces. For any $\varepsilon > 0$, there exists $i^* \in \mathbb{N}$ such that

$$\forall i > i^*, \forall z_0 \in \mathcal{B}_{z_0^*}, \quad \varepsilon(\mathcal{T}_i, z_0) < \varepsilon. \quad (2.41)$$

◁

Proof of Lemma 2.2.5. Now, we fix $z_0, z_0' \in \mathcal{B}_{z_0^*}$, and we consider the approximate solutions $\tilde{\mathbf{z}}^{z_0}, \tilde{\mathbf{z}}^{z_0'}$, λ and starting from z_0 and z_0' :

$$\begin{cases} \tilde{\mathbf{z}}_0^{z_0} &= z_0, \\ \tilde{\mathbf{z}}_{k+1}^{z_0} &= \tilde{\mathbf{z}}_k^{z_0} + h \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z_0}, h) \end{cases}, \quad \begin{cases} \tilde{\mathbf{z}}_0^{z_0'} &= z_0', \\ \tilde{\mathbf{z}}_{k+1}^{z_0'} &= \tilde{\mathbf{z}}_k^{z_0'} + h \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z_0'}, h) \end{cases}$$

We recall that the exact and approximate solutions to the ODE defined by f , λ and z_0 are denoted $\mathbf{z}^{z_0}$ and $\tilde{\mathbf{z}}^{z_0}$ respectively. For $z_0 \in \mathcal{B}_{z_0^*}$ and $t \in \mathcal{T}$, we adapt the consistency error defined in Eq. (1.13) to the initial condition:

$$\varepsilon^1(t, z_0) = \|\mathbf{z}^{z_0}(t) - \tilde{\mathbf{z}}^{z_0}(t)\|_{\mathbb{R}^n} \quad (\text{B.12})$$

The global approximation error satisfies $\varepsilon(\mathcal{T}, z_0) = \max_{t \in \mathcal{T}} \varepsilon^1(t, z_0)$. The proof of Lemma 2.2.5 can be derived from the following theorem:

Theorem B.3.1.

If the function $\mathcal{F}$ defined in Eq. (1.11) is κ_1 -Lipschitz w.r.t. λ and κ_2 -Lipschitz continuous w.r.t. y , then the approximation method is stable w.r.t. the consistency error, i.e. there

exists $\mathcal{K} > 0$ such that

$$\forall z_0, z'_0 \in \mathcal{B}(z_0^*, \rho), \max_{0 \leq k \leq J} |\varepsilon^1(t_k, z_0) - \varepsilon^1(t_k, z'_0)| \leq \mathcal{K} \|z_0 - z'_0\|_{\mathbb{R}^m}. \quad (\text{B.13})$$

◁

Proof. By assumption, $\mathcal{F}$ is κ_2 -Lipschitz w.r.t. y , so we have, $\forall 0 \leq k \leq J$,

$$\|\mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z_0}, h) - \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z'_0}, h)\|_{\mathbb{R}^n} \leq \kappa_2 \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n}$$

It follows that

$$\begin{aligned} \|\tilde{\mathbf{z}}_{k+1}^{z_0} - \tilde{\mathbf{z}}_{k+1}^{z'_0}\|_{\mathbb{R}^n} &\leq \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n} + h \|\mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z_0}, h) - \mathcal{F}(t_k, \tilde{\mathbf{z}}_k^{z'_0}, h)\|_{\mathbb{R}^n} \\ &\leq \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n} + h\kappa_2 \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n} \\ &\leq (1 + h\kappa_2) \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n} \\ &\leq (1 + h\kappa_2)^{k+1} \|\tilde{\mathbf{z}}_0^{z_0} - \tilde{\mathbf{z}}_0^{z'_0}\|_{\mathbb{R}^n}, \quad \text{by immediate recursion} \end{aligned}$$

This leads to

$$\max_{0 \leq k \leq J} \|\tilde{\mathbf{z}}_k^{z_0} - \tilde{\mathbf{z}}_k^{z'_0}\|_{\mathbb{R}^n} \leq (1 + h\kappa_2)^J \|\tilde{\mathbf{z}}_0^{z_0} - \tilde{\mathbf{z}}_0^{z'_0}\|_{\mathbb{R}^n} \quad (\text{B.14})$$

Further, it is proved in [17] that if $\mathcal{F}$ is Lipschitz continuous w.r.t. the initial condition z_0 , then the exact solution z^{z_0} is also Lipschitz continuous w.r.t. x_0 , i.e. there exists $\tilde{\kappa}_2 > 0$ such that

$$\forall z_0, \tilde{z}_0 \in \mathcal{B}_{z_0^*}, \forall T_1 \leq t \leq T_2, \|z^{z_0}(t) - z^{\tilde{z}_0}(t)\|_{\mathbb{R}^n} \leq \tilde{\kappa}_2 \|z_0 - \tilde{z}_0\|_{\mathbb{R}^n} \quad (\text{B.15})$$

Thus, we have

$$\begin{aligned} |\varepsilon_h(z_0) - \varepsilon_h(z'_0)| &\leq \|\mathbf{z}^{z_0}(t_k) - \tilde{\mathbf{z}}^{z_0}(t_k) - \mathbf{z}^{z'_0}(t_k) - \tilde{\mathbf{z}}^{z'_0}(t_k)\|_{\mathbb{R}^n} \\ &\leq \|\mathbf{z}^{z_0}(t_k) - \tilde{\mathbf{z}}^{z_0}(t_k)\|_{\mathbb{R}^n} + \|\mathbf{z}^{z'_0}(t_k) - \tilde{\mathbf{z}}^{z'_0}(t_k)\|_{\mathbb{R}^n} \\ &\leq \mathcal{K} \|z_0 - z'_0\|_{\mathbb{R}^n}, \end{aligned}$$

where $\mathcal{K} = (1 + \kappa_2)^J + \tilde{\kappa}_2$, which completes the proof for Eq. (B.13). $\square$

It remains to show that Theorem B.3.1 implies Lemma 2.2.5.

Let $\varepsilon > 0$, then $(h_i)_{i \geq 0}$ be a sequence of discretization steps such that $\lim_{i \rightarrow \infty} h_i = 0$.

Since the approximation method given by Eq. (1.11) is assumed to be convergent, each function $\varepsilon(\mathcal{T}_i, z_0)$ defined in Eq. (B.12) is pointwise convergent to 0. Furthermore, we recall that $\mathcal{F}$ is Lipschitz continuous w.r.t. the initial condition $z_0 \in \mathcal{B}_{z_0^*}$. Hence, Theorem B.3.1 implies that the functions $(\varepsilon(\mathcal{T}_i, \cdot))_{i \geq 0}$ are also Lipschitz continuous, with uniform Lipschitz constant $\mathcal{K}$:

$$|\varepsilon(\mathcal{T}_i, z_0) - \varepsilon(\mathcal{T}_i, z'_0)| \leq \mathcal{K} \|z_0 - z'_0\|_{\mathbb{R}^n}, \quad \forall z_0, z'_0 \in \mathcal{B}_{z_0^*}, \quad \forall i \in \mathbb{N}.$$

Consequently, the functions $(\varepsilon(\mathcal{T}_i, \cdot))_{i \geq 0}$ are uniformly equicontinuous. Then, Arzelà-Ascoli Theorem [22] implies that the sequence $(\varepsilon(\mathcal{T}_i, \cdot))_{i \geq 0}$ converges uniformly to 0 on $\mathcal{B}_{z_0^*}$: there exists $i^* \in \mathbb{N}$ such that

$$\varepsilon_{h_i}(z_0) < \varepsilon, \forall i \geq i^*, \forall x_0 \in \mathcal{B}_{z_0^*}.$$

and Lemma 2.2.5 is proved. □

B.4 Proof of Theorem 2.2.8

Theorem 2.2.8 (Stability of the approximation w.r.t. initial condition z_0).

Let $f : \mathbb{R} \times \mathbb{R}^n \rightarrow \mathbb{R}^n$ be a function defining an ODE of variable z . Let $\rho > 0$ be a radius and $\varepsilon > 0$ be the target integration error for the ODE integration. Let $z_0^ \in \Omega$ be an initial condition to evaluate, such that $\mathcal{B}_{z_0^*} = \{z_0 \in \mathbb{R}^n \mid \|z_0^* - z_0\|_{\mathbb{R}^n} \leq \rho\} \subseteq \mathring{\Omega}$. Let $\theta \in (0, 1)$ be the risk and $\alpha > 0$ be the precision for the Monte-Carlo procedure. Let $r : \mathcal{I} \rightarrow \mathbb{R}$ be a bounded reward function on $\mathcal{I}$; we consider the greatest (resp. lowest) value U (resp. L) of $r(\cdot)$ on $\mathcal{B}_{z_0^*}$.*

Consider the hidden risk $\xi = 1 - \sqrt{1 - \theta}$. The auxiliary rewards $r_{-\varepsilon}$ and $r_{+\varepsilon}$ defined in Definition 2.1.13 satisfy

$$\mathbb{P}(r_- \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{-\varepsilon} + \alpha]) \geq 1 - \xi, \quad \mathbb{P}(r_+ \in [\hat{r}_{+\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \xi, \quad (2.44)$$

where the estimators $\hat{r}_{-\varepsilon} = \hat{r}_{-\varepsilon}(z_0^)$ and $\hat{r}_{+\varepsilon} = \hat{r}_{+\varepsilon}(z_0^*)$ are indepently computed after performing, for each estimator, a number $N' = \frac{\log(2/\xi)}{2\alpha^2} (U - L)^2$ of simulations induced by f and an initial condition z_0 sampled in $\mathcal{B}_{z_0^*}$.*

Furthermore, there exists $\varepsilon^ > 0$ sufficiently small, such that for any tolerance $\varepsilon < \varepsilon^*$ the following statements hold:*

-
- the expected value $r = \mathbb{E}_{\mathcal{B}_{z_0^*}}[r]$ of reward function $r(\cdot)$ on $\mathcal{B}_{z_0^*}$ satisfies the estimation

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq 1 - \theta \quad (2.45)$$

- the distance between $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ satisfies:

$$\mathbb{P}(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha) \geq 1 - \theta \quad (2.46)$$

◁

Proof of Theorem 2.2.8. The first result of this theorem, that is Eq. (2.44) which establishes a confidence interval for the expected value of a reward function $r(\cdot)$ in a neighborhood of z_0^* , is implied by Eq. (2.6). Hence, the proof here will focus on Eq. (2.45) and Eq. (2.46), that is establishing a confidence interval for $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r]$ using the Monte-Carlo estimators $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ of $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r_{-\varepsilon}]$ and $\mathbb{E}_{\mathcal{B}_{z_0^*}}[r_{+\varepsilon}]$, respectively.

The independence of the samples used to determine the estimators $\hat{r}_{-}$ and $\hat{r}_{+}$ guarantees that

$$\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) = \mathbb{P}(r \geq \hat{r}_{-\varepsilon} - \alpha) \times \mathbb{P}(r \leq \hat{r}_{+\varepsilon} + \alpha).$$

Using Eq. (2.45), we have $\mathbb{P}(r_{-\varepsilon} \geq \hat{r}_{-\varepsilon} - \alpha) \geq 1 - \xi$. Next, Definition 2.1.13 implies $\mathbb{P}(r \geq \hat{r}_{-\varepsilon} - \alpha) \geq \mathbb{P}(r_{-\varepsilon} \geq \hat{r}_{-\varepsilon} - \alpha) \geq 1 - \xi$. Similarly, we have $\mathbb{P}(r \leq \hat{r}_{+\varepsilon} + \alpha) \geq 1 - \xi$, and finally $\mathbb{P}(r \in [\hat{r}_{-\varepsilon} - \alpha, \hat{r}_{+\varepsilon} + \alpha]) \geq (1 - \xi)^2 = 1 - \theta$, by independence of $\hat{r}_{-\varepsilon}$ and $\hat{r}_{+\varepsilon}$ and since $\xi = 1 - \sqrt{1 - \theta}$.

Since $r_{-\varepsilon}$ and $r_{+\varepsilon}$ are continuous w.r.t. ε by construction (see Definition 2.1.13), there exists $\varepsilon^* > 0$ such that $|r_{-\varepsilon} - r_{+\varepsilon}| \leq \alpha$, for $\varepsilon < \varepsilon^*$.

Next, we write

$$|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq |\hat{r}_{-\varepsilon} - r_{-\varepsilon}| + |\hat{r}_{+\varepsilon} - r_{+\varepsilon}| + |r_{-\varepsilon} - r_{+\varepsilon}|, \quad (\text{B.16})$$

hence we have, for $\varepsilon < \varepsilon^*$:

$$\mathbb{P}\left(|\hat{r}_{-\varepsilon} - \hat{r}_{+\varepsilon}| \leq 3\alpha\right) \geq \mathbb{P}(|\hat{r}_{-\varepsilon} - r_{-\varepsilon}| \leq \alpha) \times \mathbb{P}(|\hat{r}_{+\varepsilon} - r_{+\varepsilon}| \leq \alpha) \times \mathbb{P}(|r_{-\varepsilon} - r_{+\varepsilon}| \leq \alpha) \quad (\text{B.17})$$

$$\geq (1 - \xi)^2 \times 1 = 1 - \theta. \quad (\text{B.18})$$

In parallel, Lemma 2.2.5 guarantees that a sufficiently small value exists for h such that the global stability error can be uniformly bounded on $\mathcal{B}_{z_0^*}$ by ε , which completes the proof. $\square$

PROOFS FOR CHAPTER 3

C.1 Proof of Corollary 3.1.2

Corollary 3.1.2.

For all $z_0 \in \Omega$, the Cauchy problem defined by Eq. (3.3) and $z(0) = z_0$ admits a unique solution denoted $\mathbf{z}(t, z_0, \lambda)$, defined on $[0, +\infty)$, whose components are non-negative. $\triangleleft$

Proof. Since Ω is compact, Theorem 1.2.6 ensures that the solution $\mathbf{z}(t, z_0, \lambda)$, defined for any $t \in \mathbb{R}$, exists and is unique. Moreover, Theorem 3.1.1 establishes that Ω is positively invariant, hence the trajectory stays within Ω , and its components are positive for any $t \in [0, \infty)$. $\square$

C.2 Proof of Lemma 3.2.5

Lemma 3.2.5.

Let $m(\cdot)$ be a measure on phase space Ω . Let $\mathfrak{Q}$ be a partition of Ω , such that all region $R_i \in \mathfrak{Q}$ have a non-zero measure $m(R_i)$. For any $R_i, R_j \in \mathfrak{Q}$, we consider the value

$$p_{i,j}^{\tau,\lambda} = \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\})}{m(R_i)}. \quad (3.16)$$

For each region $R_i \in \mathfrak{Q}$, the values $p_{i,j}^{\tau,\lambda}$ define a probability distribution over the regions R_j of $\mathfrak{Q}$. $\triangleleft$

Proof. Since $m(\cdot)$ is a measure, both quantities $m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\})$ and $m(R_i)$ are positive or null. By assumption, $m(R_i) > 0$, so the fraction is well-written.

Moreover, we have $\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\} \in R_i$, thus $m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\}) \leq m(R_i)$ and $p_{i,j}^{\tau,\lambda} \in [0, 1]$.

Furthermore, for any $R_i \in \Omega$ we have

$$\begin{aligned}
\sum_{R_j \in \Omega} p_{i,j}^{\tau,\lambda} &= \sum_{R_j \in \Omega} \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\})}{m(R_i)}, \\
&= \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in \bigcup_{R_j \in \Omega} R_j\})}{m(R_i)}, \\
&= \frac{m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in \Omega\})}{m(R_i)}, \\
&= \frac{m(R_i)}{m(R_i)}, \\
&= 1.
\end{aligned} \tag{C.1}$$

Hence, for any $R_i \in \Omega$, the values $p_{i,j}^{\tau,\lambda}, R_j \in \Omega$ satisfy:

- $p_{i,j}^{\tau,\lambda} \in [0, 1]$;
- $\sum_{R_j \in \Omega} p_{i,j}^{\tau,\lambda} = 1$.

As such, they define a probability distribution over Ω , and the proof is complete. $\square$

C.3 Proof of Lemma 3.2.9

Lemma 3.2.9 (*Almost- τ -simulation*).

Let $\mathbb{S}_\lambda = (f, \lambda, \Omega_\lambda)$ be a differential system, then let z_0 be an initial state for $\mathbb{S}_\lambda$. Let $\tau > 0$ be a time step. Consider the following initial distribution over the states of Ω :

$$\mathbb{I}(R) = \begin{cases} 1, & \text{if } z_0 \in R, \\ 0, & \text{otherwise.} \end{cases} \tag{3.17}$$

The MC $\mathbb{M}_\lambda^\tau = (\Omega, \mathbb{P}_\lambda^\tau, \mathbb{I})$ almost- τ -simulates the behaviour of differential system $\mathbb{S}_\lambda$.

$\triangleleft$

Proof. We prove that the relation $\mathcal{R} = \{(z, R) \subseteq \Omega \times \Omega \mid z \in R\}$ defines a τ -simulation relation (see Definition 3.2.8 on page 89) almost everywhere on $\Omega \times \Omega$. To this end, we consider the set $\check{\mathcal{R}} \subset \mathcal{R}$ of pairs (z_i, R_i) such that whenever z_j is a τ -successor of z_i , the region R_j such that $z_j \in R_j$ is not a successor of R_i in $\mathbb{M}$, and prove that its measure $m(\check{\mathcal{R}})$ is null.

-
1. First, we prove that for any initial state $z_0 \in \mathbb{S}_\lambda$, there exists a unique region $R_0 \in \mathfrak{Q}$ such that $z \in R$: this is a direct consequence of $\mathfrak{Q}$ being a partition of Ω . Hence, $\mathcal{I}$ is the Dirac distribution concentrated on R_0 , such that R_0 is an initial state for $\mathbb{M}$.
 2. Second, we prove that point 2 of Definition 3.2.8 holds almost everywhere in $\Omega \times \mathfrak{Q}$, i.e. that $m(\check{\mathcal{R}}) = 0$. Let $(z_i, R_i \in \mathcal{R})$ be a pair in the relation, then let $z_j = \mathbf{z}(\tau, z_i, \lambda)$ be the τ -successor of z_i . Let $R_j \in \mathfrak{Q}$ be the region of Ω such that $z_j \in R_j$. We distinguish two cases:

- z_j is in the interior of R_j .

Let $\rho_j > 0$ be a radius such that the ball $\mathcal{B}_j$ of radius ρ_j around z_j is completely inside R_j . Let $\rho_i > 0$ be such that the trajectory $\mathbf{z}(\tau, \cdot, \lambda)$ is continuous w.r.t. initial condition z_0 in the ball $\mathcal{B}_i$ of radius ρ_i around z_i (see [61], Section 2.3 p.79). Then, there exists $\rho_i^* \in (0, \rho_i]$ such that

$$\|z_i - z_0\|_{\mathbb{R}^n} \leq \rho_i^* \Rightarrow \|z_j - \mathbf{z}(\tau, z_0, \lambda)\|_{\mathbb{R}^n} \leq \rho_j, \quad (\text{C.2})$$

and $z_0 \in \mathcal{B}_i \Rightarrow \mathbf{z}(\tau, z_0, \lambda) \in R_j$. It follows that $m(\{z_0 \in R_i \mid \mathbf{z}(\tau, z_0, \lambda) \in R_j\}) \geq m(\mathcal{B}_i) > 0$, hence $p_{in,out}^{\tau,\lambda} > 0$ and there exists a transition from R_i to R_j in $\mathbb{M}$: R_j is a possible successor of R_i , and point 2 of Definition 3.2.8 holds in this case.

- z_j is on the border ∂R_j of R_j .

This case is problematic because, in general, we cannot prove that R_j is a possible successor of R_i in $\mathbb{M}$. Instead, we will prove that the set $\Delta = \{z_0 \in \Omega \mid \exists R \in \mathfrak{Q}, \mathbf{z}(\tau, z_0, \lambda) \in \partial R\}$ of initial conditions z_0 inducing a τ -transition that stops exactly on the border of a region is such that $m(\Delta) = 0$. Note that $z_0 \in \check{\mathcal{R}} \Rightarrow z_0 \in \Delta$ but the converse is not true.

First, recall that Lebesgue's density theorem (see [51], Corollary 2.14 p.38) ensures that the border ∂R_i of a region $R_i \in \mathfrak{Q}$ has a null measure. Since trajectories are unique w.r.t. initial condition, the inverse image $\mathbf{z}^{[-1]}(\tau, \partial R_i, \lambda) = \Delta_i$ of ∂R_i also has a null measure. Second, additivity of measure $m(\cdot)$ ensures that the measure of a finite union is less than or equal to the sum of measures of each set, hence $0 \leq m(\check{\mathcal{R}}) \leq m(\Delta) = m(\bigcup_{R_i \in \mathfrak{Q}} \Delta_i) \leq \sum_{R_i \in \mathfrak{Q}} m(\Delta_i) = 0$, $\check{\mathcal{R}}$ has a null measure and the proof is complete.

□

C.4 Proof of Corollary 3.2.10

Corollary 3.2.10 (Reachability).

Let $k \in \mathbb{N}$ be a positive integer. Let $R \in \mathfrak{Q}$ be a target region. If $\mathbb{M}_\lambda$ can reach R within k transitions with probability 1, then the abstracted system $\mathbb{S}_\lambda$ will reach R within $k\tau$ time units with probability 1. $\triangleleft$

Proof. Without loss of generality, we consider only the minimal behaviors satisfying the property φ = “reaching R within k steps”, i.e. behaviors of length $l \in [1..k]$ in which the first occurrence of R is the last state. In order to prove that the set $\bar{\mathcal{B}}_l$ of behaviors of length l in $\mathbb{S}_\lambda$ that do not satisfy φ is of null measure, it suffice to prove that the set $\check{\mathcal{B}}_l$ of behaviors for which we cannot prove that they satisfy φ (which contains $\bar{\mathcal{B}}_l$) is of null measure.

Assume that $\mathbb{M}_\lambda$ satisfies φ with probability 1. Let $l \in [1..k]$ be a length of behavior. Since the probability distribution $\mathbb{P}_{l+1}$ over the behaviors of length $l+1$ in $\mathbb{M}_\lambda$ is discrete, we know that all behaviors of length $l+1$ in $\mathbb{M}_\lambda$ satisfy the property φ . In particular, we know that the set $\mathcal{B}_{\mathbb{S}_\lambda}$ of behaviors in $\mathbb{M}$ that abstract a behavior in $\mathbb{S}_\lambda$ do satisfy φ .

Now, let us consider the behavior $\pi_{\mathbb{S}_\lambda} = z_0 \rightarrow z_1 \rightarrow z_2 \cdots \rightarrow z_l$ of length $l+1$ in $\mathbb{S}_\lambda$ (hence, of duration $l\tau$); let $R_l \in \mathfrak{Q}$ be the region such that $z_l \in R_l$. Since all behaviors in $\mathbb{M}_\lambda$ satisfy φ , we know that the abstracted behavior (if it exist) $\pi_{\mathbb{M}} = R_0 R_1 R_2 \dots R_l$ of $\pi_{\mathbb{S}_\lambda}$ satisfies φ : $R_l = R$, and $\pi_{\mathbb{S}_\lambda}$ does satisfy φ if it can be abstracted.

Hence, we cannot prove that $\pi_{\mathbb{S}_\lambda}$ satisfies φ only if we cannot prove that it can be abstracted as a behavior of $\mathbb{M}_\lambda$: since $\mathbb{M}_\lambda$ almost- τ -simulates $\mathbb{S}_\lambda$, the set of transitions which cannot be abstracted in $\mathbb{M}_\lambda$ is of null measure, hence the set of finite behaviors in $\mathbb{S}_\lambda$ for which we cannot prove that they satisfy φ is of null measure, and the proof is complete. $\square$

C.5 Proof of Lemma 3.2.13

Lemma 3.2.13.

Let $\mathbf{P}_\forall$ and $\mathbf{P}_\tau$ be the functions defined in Eqs. (3.26) and (3.27) on page 95, respectively.

The function

$$\begin{aligned} \mathbf{P} : \quad \mathbf{Q} \times \mathbf{A} \times \mathbf{Q} &\rightarrow [0, 1] \\ ((R_i, \lambda_i), a, (R_j, \lambda_j)) &\mapsto \begin{cases} \mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\tau \\ \mathbf{P}_\mathbb{V}((R_i, \lambda_i), (R_j, \lambda_j)), & \text{if } a = a_\mathbb{V} \\ \mathbf{1}_{(R_i, \lambda_i)}((R_j, \lambda_j)), & \text{if } a = \emptyset \end{cases} \end{aligned} \quad (3.29)$$

defines a transition probability distribution over the states of $\mathbf{Q}$. $\triangleleft$

Proof. We will prove that both $\mathbf{P}_\mathbb{V}$ and $\mathbf{P}_\tau$ are probability distributions. Let (R_i, λ_i) be a state of $\mathbf{Q}$.

First, we prove that $\mathbf{P}_\tau$ defines a probability distribution. For any $(R_j, \lambda_j) \in \mathbf{Q}$, we have $\mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_j)) \in [0, p_{i,j}^{\lambda_i}] \subseteq [0, 1]$. Moreover,

$$\begin{aligned} \sum_{(R_j, \lambda_j) \in \mathbf{Q}} \mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_j)) &= \sum_{(R_j, \lambda_j) \in \mathbf{Q}} \mathbf{P}_\tau((R_i, \lambda_i), (R_j, \lambda_i)) \\ &= \sum_{(R_j, \lambda_j) \in \mathbf{Q}} p_{i,j}^{\lambda_i} \\ &= 1. \end{aligned} \quad (C.3)$$

Second, we prove that $\mathbf{P}_\mathbb{V}$ defines a probability distribution. For any $(R_j, \lambda_j) \in \mathbf{Q}$, we have

$$\begin{aligned} b\mathbb{V}(\lambda_i, \lambda_j) &\leq 1, \\ 1 - b &\leq 1, \end{aligned} \quad (C.4)$$

and $\mathbf{P}_\mathbb{V}((R_i, \lambda_i), (R_j, \lambda_j)) \in [0, 1]$. Moreover,

$$\begin{aligned}
\sum_{(R_j, \lambda_j) \in \mathbf{Q}} \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_j, \lambda_j)) &= \sum_{R_j \in \Omega} \sum_{\lambda_j \in \Lambda} \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_j, \lambda_j)), \\
&= \sum_{\lambda_j \in \Lambda} \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_i, \lambda_j)), \\
&= \sum_{\lambda_j \in \Lambda} (b\mathbb{V}(\lambda_i, \lambda_j) + (1-b)\mathbf{1}_{\lambda_i}(\lambda_j)), \\
&= (1-b) + b \sum_{\lambda_j \in \Lambda} \mathbb{V}(\lambda_i, \lambda_j) \\
&= (1-b) + b \\
&= 1.
\end{aligned} \tag{C.5}$$

Finally, since $\mathbf{1}_{(R_i, \lambda_i)}$ is a Dirac distribution on $\mathbf{Q}$, it is a probability distribution.

As such, $\mathbf{P}$ defines a probability distribution no matter the chosen action, and the proof is complete. $\square$

C.6 Proof of Lemma 3.2.16

Lemma 3.2.16 (Almost- τ -simulation of the hybrid system and $\mathbb{Y}$).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a controller, whose action concatenates a Bernoulli distribution $\mathbb{B}$ of parameter $b \in [0, 1]$ and a discrete distribution function $\mathbb{V}$ over the parameter set Λ . Let (z_0, λ_0) be an initial state for $\mathcal{S}$. Consider the action set $\mathbf{A} = \{a_\tau, a_{\mathbb{V}}\}$, as well as the probability distribution function $\mathbf{P}$ defined in Eq. (3.28). Consider the following initial distribution over the states of $\mathbf{Q}$:

$$\mathbf{I}(R, \lambda) = \begin{cases} 1, & \text{if } z_0 \in R_0 \text{ and } \lambda = \lambda_0, \\ 0, & \text{otherwise.} \end{cases} \tag{3.30}$$

The MDP $\mathbb{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ almost- τ -simulates $\mathcal{S}$ under controller $\mathbb{Y}$. $\triangleleft$

Proof. We prove that the relation $\mathcal{R} = \{((z, \lambda), (R, \lambda)) \in \Omega \times \Lambda \times \mathbf{Q} \mid z \in R\}$ defines a τ -simulation almost everywhere in $\Omega \times \Lambda \times \mathbf{Q}$.

1. First, we prove that for any initial state (z_0, λ_0) of $\mathcal{S}$, there exists an initial state (R_0, λ_0) of $\mathbb{M}$ such that $z_0 \in R_0$. This is ensured by the definition of the initial

distribution $\mathbf{I}$ of $\mathbf{M}$, which is concentrated on (R_0, λ_0) with $z_0 \in R_0$.

2. Second, we prove that point 2. of the τ -simulation of a hybrid system (see Definition 3.2.8 on page 89) holds almost everywhere in $\Omega \times \Lambda \times \mathbf{Q}$. For convenience in the wording of the proof, we will say that a transition $(z_i, \lambda_i) \rightarrow (z_j, \lambda_j)$ in $\mathcal{S}$ can be matched in $\mathbf{M}$ iff there exists R_i, R_j in partition $\mathfrak{Q}$ such that $z_i \in R_i$, $z_j \in R_j$ and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$ for some action $a \in \mathbf{A}$, i.e. $\mathbf{P}((R_i, \lambda_i), a, (R_j, \lambda_j)) > 0$. Then, proving that the relation $\mathcal{R}$ holds almost everywhere in $\Omega \times \Lambda \times \mathbf{Q}$ equates to proving that almost every transition in $\mathcal{S}$ can be matched in $\mathbf{M}$, i.e. that the set of transitions in $\mathcal{S}$ which cannot be matched in $\mathbf{M}$ has a measure equal to zero.

Let $(z_i, \lambda_i) \in \Omega \times \Lambda$ be a state for $\mathcal{S}$, then let $(R_i, \lambda_i) \in \mathbf{Q}$ be such that $z_i \in R_i$. It follows that $((z_i, \lambda_i), (R_i, \lambda_i)) \in \mathcal{R}$. A transition in $\mathcal{S}$ may be of two kinds, which are recalled in the following sequence:

$$(z_i, \lambda_i) \xrightarrow{\tilde{\mathbb{V}}_{\lambda_i}} (z_j, \lambda_j) \xrightarrow{\tau} (z_k, \lambda_k), \quad (\text{C.6})$$

where $\tilde{\mathbb{V}}_{\lambda_i}$ is the split distribution corresponding to the probability distribution function $\mathbb{V}$ and incorporating the Bernoulli process $\mathbb{B}$, for controller $\mathbb{Y}$ (see Eq. (3.20) on page 93). Note that the order of transitions is irrelevant for the proof.

Let R_j, R_k be two regions in $\mathfrak{Q}$, such that $z_j \in R_j$ and $z_k \in R_k$.

- We first focus on $\xrightarrow{\tilde{\mathbb{V}}_{\lambda_i}}$. Since $\tilde{\mathbb{V}}_{\lambda_i}$ operates only on the parameter λ of the system, we have $z_i = z_j$, hence $R_i = R_j$. Moreover, since $(z_i, \lambda_i) \xrightarrow{\tilde{\mathbb{V}}_{\lambda_i}} (z_j, \lambda_j)$ is a transition in $\mathcal{S}$, we have $\mathbb{V}(\lambda_i, \lambda_j) > 0$.

Then,

$$\begin{aligned} \mathbf{P}_{\mathbb{V}}((R_i, \lambda_i), (R_j, \lambda_j)) &= \tilde{\mathbb{V}}_{\lambda_i}(\lambda_j) \\ &= b\mathbb{V}(\lambda_i, \lambda_j) + (1 - b)\mathbf{1}_{\lambda_i}(\lambda_j) \\ &\geq \max(b\mathbb{V}(\lambda_i, \lambda_j), 1 - b) \\ &> 0. \end{aligned} \quad (\text{C.7})$$

Hence, $\mathbf{P}((R_i, \lambda_i), a_{\mathbb{V}}, (R_j, \lambda_j)) > 0$ and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$.

-
- We now focus on $\xrightarrow{\tau}$. Since the continuous transitions in $\mathcal{S}$ do not change the value of λ , we have $\lambda_j = \lambda_k$.

Then,

$$\begin{aligned} \mathbf{P}((R_j, \lambda_j), a_\tau, (R_k, \lambda_k)) &= \mathbf{P}_\tau((R_j, \lambda_j), (R_k, \lambda_k)), \\ &= p_{j,k}^{\lambda_j}. \end{aligned} \tag{C.8}$$

Similarly to Lemma 3.2.9, proving that $p_{j,k}^{\lambda_j} > 0$ is difficult if both z_j, z_k are on the border of their respective region.

As such, this case generates a set Δ_j of states in Ω which induce a continuous transition of duration τ for which we cannot prove that a matching transition exists in $\mathbf{M}$; this set is of measure zero according to Lemma 3.2.9, for each $\lambda_j \in \Lambda$.

The union $\Delta = \bigcup_{\lambda_j \in \Lambda} \Delta_j \times \{\lambda_j\}$ is a finite union of sets of measure zero, hence it is also of measure zero. Thus, the set Δ of states in $\Omega \times \Lambda$ inducing a continuous transition of duration τ for which we cannot find a matching transition in $\mathbf{M}$ through action a_τ is of measure zero: we have $\mathbf{P}((R_i, \lambda_i), a_\tau, (R_j, \lambda_j)) > 0$ except for a set of measure zero.

Thus, $\check{\mathcal{R}} \subseteq \Delta$ is of measure zero, hence the simulation relation $\mathcal{R}$ holds almost everywhere: the proof is complete. □

C.7 Proof of Theorem 3.2.21

Theorem 3.2.21 (*Almost- τ -simulation of the hybrid system*).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a controller, whose action concatenates a Bernoulli distribution $\mathbb{B}$ of parameter $b \in [0, 1]$ and a discrete distribution function $\mathbb{V}$ over the parameter set Λ . Let (z_0, λ_0) be an initial state for $\mathcal{S}$. Consider the action set $\mathbf{A} = \{a_\tau, a_\mathbb{V}, \emptyset\} \cup \{a_\mathbb{D} \mid \mathbb{D} \in \mathcal{D}\}$, as well as the probability distribution function $\mathbf{P}$ defined in Eq. (3.38). Consider the following initial distribution over the states of $\mathbf{Q}$:

$$\mathbf{I}(R, \lambda) = \begin{cases} 1, & \text{if } z_0 \in R_0 \text{ and } \lambda = \lambda_0, \\ 0, & \text{otherwise.} \end{cases} \tag{3.39}$$

The MDP $\mathbf{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ almost τ -simulates $\mathcal{S}$. $\triangleleft$

Proof. We prove that the relation $\mathcal{R} = \{((z, \lambda), (R, \lambda)) \in \Omega \times \Lambda \times \mathbf{Q} \mid z \in R\}$ defines a τ -simulation **almost** everywhere in $\Omega \times \Lambda \times \mathbf{Q}$.

1. First, we prove that for any initial state (z_0, λ_0) of $\mathcal{S}$, there exists an initial state (R_0, λ_0) of $\mathbf{M}$ such that $z_0 \in R_0$. This is ensured by the definition of the initial distribution $\mathbf{I}$ of $\mathbf{M}$, which is concentrated on (R_0, λ_0) with $z_0 \in R_0$.
2. Second, we prove that point 2 of the t -simulation of a hybrid system (see Definition 3.2.8 on page 89) holds almost everywhere in $\Omega \times \Lambda \times \mathbf{Q}$.

For convenience in the wording of the proof, we will say that a transition $(z_i, \lambda_i) \rightarrow (z_j, \lambda_j)$ in $\mathcal{S}$ can be matched in $\mathbf{M}$ iff there exists R_i, R_j in partition $\mathfrak{Q}$ such that $z_i \in R_i$, $z_j \in R_j$ and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$ for some action $a \in \mathbf{A}$, i.e. $\mathbf{P}((R_i, \lambda_i), a, (R_j, \lambda_j)) > 0$. Then, proving that the relation $\mathcal{R}$ holds almost everywhere in $\Omega \times \Lambda \times \mathbf{Q}$ equates to proving that almost every transition in $\mathcal{S}$ can be matched in $\mathbf{M}$, i.e. that the set of transitions in $\mathcal{S}$ which cannot be matched in $\mathbf{M}$ has a measure equal to zero.

Let $(z_i, \lambda_i) \in \Omega \times \Lambda$ be a state for $\mathcal{S}$, then let $(R_i, \lambda_i) \in \mathbf{Q}$ be such that $z_i \in R_i$. It follows that $((z_i, \lambda_i), (R_i, \lambda_i)) \in \mathcal{R}$. A transition in $\mathcal{S}$ may be of three kinds, which are recalled in the following sequence:

$$(z_i, \lambda_i) \xrightarrow{\mathbb{D}^{\mathbb{X}}} (z_j, \lambda_j) \xrightarrow{\tilde{\mathbb{V}}_{\lambda_j}} (z_k, \lambda_k) \xrightarrow{\varepsilon} (z_l, \lambda_l), \quad (\text{C.9})$$

where $\mathbb{D}^{\mathbb{X}}$ is a distribution chosen by deterministic controller $\mathbb{X}$, and $\tilde{\mathbb{V}}_{\lambda_j}$ is the probability distribution associated to natural controller $\mathbb{Y}$. Note that the order of transitions $\xrightarrow{\mathbb{D}^{\mathbb{X}}}, \xrightarrow{\mathbb{V}}, \xrightarrow{\varepsilon}$ is irrelevant for the proof.

- We first focus on $\xrightarrow{\mathbb{D}^{\mathbb{X}}}$, for some distribution $\mathbb{D} \in \mathcal{D} \cup \mathcal{W}$.

By definition of $\mathbf{A}$ (see Eq. (3.37) on page 102), every distribution $\mathbb{D}$ in $\mathcal{D} \cup \mathcal{W}$ is mapped to an action $a_{\mathbb{D}}$ in $\mathbf{A}$; it follows that every discrete transition in $\mathcal{S}$ induced by a distribution $\mathbb{D}^{\mathbb{X}}$ chosen by $\mathbb{X}$ can be matched by a transition induced by an action $a_{\mathbb{D}^{\mathbb{X}}}$ in $\mathbf{M}$. Let $R_j \in \mathfrak{Q}$ be such that $z_j \in R_j$. We distinguish three cases w.r.t. $\mathbb{D}^{\mathbb{X}}$:

- (a) $\mathbb{D}^{\mathbb{X}} \in \mathcal{D}^{\Lambda}$ only resets the parameter of $\mathcal{S}$.

Since $\mathbb{D}^{\mathbb{X}}(\lambda_j) > 0$, then by definition of $\mathbf{P}$ (see Eq. (3.38) on page 102), we have $\mathbf{P}((R_i, \lambda_i), a_{\mathbb{D}^{\mathbb{X}}}, (R_j, \lambda_j)) \geq \mathbb{D}^{\mathbb{X}}(\lambda_j) > 0$ and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$.

(b) $\mathbb{D}^{\mathbb{X}} \in \mathcal{D}^{\Omega}$ only resets the position of $\mathcal{S}$.

Since $\mathbb{D}^{\mathbb{X}}(z_j) > 0$, then by definition of $\mathbf{P}$ (see Eq. (3.38) on page 102), we have $\mathbf{P}((R_i, \lambda_i), a_{\mathbb{D}^{\mathbb{X}}}, (R_j, \lambda_j)) \geq \mathbb{D}^{\mathbb{X}}(z_j) > 0$ and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$.

(c) $\mathbb{D}^{\mathbb{X}} \in \mathcal{W}$ is the Dirac distribution concentrated on (R_i, λ_i) .

In that case $(R_j, \lambda_j) = (R_i, \lambda_i)$, we have $\mathbf{P}((R_i, \lambda_i), \emptyset, (R_j, \lambda_j)) = 1$, and the transition $(R_i, \lambda_i) \rightarrow (R_j, \lambda_j)$ exists in $\mathbf{M}$.

Hence, for every transition $(z_i, \lambda_i) \xrightarrow{\mathbb{D}^{\mathbb{X}}} (z_j, \lambda_j)$ in $\mathcal{S}$, there exists a state $(R_j, \lambda_j) \in \mathbf{Q}$ such that (R_j, λ_j) is a successor of (R_i, λ_i) in $\mathbf{M}$.

- We now focus on $\xrightarrow{\tilde{\nabla}_{\lambda_j}}$.

There exists an action $a_{\nabla} \in \mathbf{A}$, which induces a transition according to probability distribution function $\mathbf{P}_{\nabla}$, defined in Eq. (3.26) (see page 95). This case is covered in Lemma 3.2.16 on page 97, and the transition $(R_j, \lambda_j) \rightarrow (R_k, \lambda_k)$ exists in $\mathbf{M}$.

- Finally, we focus on $\xrightarrow{\tau}$. Again, this case is covered in Lemma 3.2.16 on page 97, and the action a_{τ} generates a set $\Delta \subset \Omega \times \Lambda$ of states in $\mathcal{S}$ for which we cannot prove that a matching transition exists in $\mathbf{M}$, such that the measure of Δ is zero. As such the transition $(R_k, \lambda_k) \rightarrow (R_l, \lambda_l)$ exists in $\mathbf{M}$, except for a set Δ of states in $\mathcal{S}$, which is of measure zero.

To conclude, we have proved that each discrete transition in $\mathcal{S}$ can be matched in $\mathbf{M}$, no matter if it is induced by $\mathbb{X}$ or $\mathbb{Y}$. We also proved that almost every continuous transition of duration τ in $\mathcal{S}$ can be matched in $\mathbf{M}$, except for a set Δ of states in $\mathcal{S}$ for which we cannot prove that a matching transition exists, but is of measure zero.

Thus, $\check{\mathcal{R}} \subseteq \Delta$ is of measure zero, hence the simulation relation $\mathcal{R}$ holds almost everywhere: the proof is complete. $\square$

C.8 Proof of Corollary 3.2.22

Corollary 3.2.22 (Bounded reachability).

Let $\mathcal{S} = (f, \Lambda, \Omega, \mathcal{T}, \mathcal{D}, \mathcal{A})$ be a hybrid system. Let $\mathbb{Y}$ be a natural process controller

related to a distribution function $\mathbb{V}$ over the parameter set Λ . Given a second controller $\mathbb{X}$ for $\mathcal{S}$, we assume that $\mathbb{X}$ and $\mathbb{Y}$ can pick one action each, in that order, and that an action ζ_τ is always picked once before $\mathbb{X}$ can make a choice. Let $\mathbf{M} = (\mathbf{Q}, \mathbf{A}, \mathbf{P}, \mathbf{I})$ be the MDP abstraction of $\mathcal{S}$, then let Ψ be the controller mimicking $\mathbb{Y}$ by being able to pick the corresponding action $a_\mathbb{V}$ in $\mathbf{A}$.

Let $i \in \mathbb{N}_{>0}$ be a positive integer, such that $i = 3k$ for some $k \in \mathbb{N}$. Let $R^* \in \Omega$ be a target region in Ω . Let Φ be a second controller for $\mathbf{M}$. We assume that Φ and Ψ can pick one action each, in that order, and that an action a_τ is always picked once before Φ can make a choice.

If Φ is such that $\mathbf{M}$ reaches R^* in i steps or less with probability 1, then there exists a corresponding controller $\mathbb{X}$ for $\mathcal{S}$ such that $\mathcal{S}$ reaches R^* in at most i transitions, and therefore in at most $k\tau$ time units, with probability 1.

In this case, $\mathbb{X}$ consists in a sequence of choices for $\mathcal{S}$ corresponding to the actions chosen in $\mathbf{M}$ under controller Φ , adapted to its setting:

- $a_\mathbb{D}$ translates to having process $\mathbb{X}$ choose action $\zeta_\mathbb{D}, \mathbb{D} \in \mathcal{D}$ in order to influence the evolution of $\mathcal{S}$ in a similar manner as $\zeta_\mathbb{D}$ influences $\mathbf{M}$;
- $\emptyset$ translates to having $\mathbb{X}$ pick the Dirac distribution $\mathbf{1}_\lambda \in \mathcal{W}$, where λ is the current parameter value.

◁

Proof. Let $\mathbb{X}$ be the controller corresponding (in $\mathcal{S}$) to Φ (in $\mathbf{M}$). This controller induces a behaviour $\pi_\mathcal{S} = s_0 \dots s_i$. In order to prove this corollary, we need to prove that the measure of the set of behaviours induced by $\mathbb{X}$ (in $\mathcal{S}$) which do not lead to region R^* is zero. Given the relation $\mathcal{R}$ between $\mathcal{S}$ and $\mathbf{M}$, we know that the measure of the set of behaviours in $\mathcal{S}$ corresponding to behaviours in $\mathbf{M}$ which do not stop in R^* is zero: since $\mathcal{R}$ defines a function between the states of $\mathcal{S}$ and $\mathbf{M}$, we can trivially define a function between finite behaviours of fixed length (as a function between the cardinal product of the respective state spaces of $\mathcal{S}$ and $\mathbf{M}$) which maps each state in a behaviour of $\mathcal{S}$ to one in a behaviour of $\mathbf{M}$. Thus, it suffices to prove that the measure of the set of behaviours in $\mathcal{S}$ which cannot be matched in $\mathbf{M}$ is zero.

Given the proof of Lemma 3.2.16 (see Section C.6 on page 137), we know that only the continuous transitions can induce a problem with their abstraction, for each $\lambda \in \Lambda$. Moreover, it follows from Theorem 3.2.21 that almost any continuous transition in $\mathcal{S}$ can

be abstracted to $\mathbf{M}$. Thus, we know that at each step $j \in [1..i]$ in a behaviour in $\mathcal{S}$, only a set Δ_λ of points in Ω induce a transition that cannot be matched in $\mathbf{M}$, for each $\lambda \in \Lambda$; hence, at each step j , no matter the value of λ , we have a set $\Delta = \bigcup_{\lambda \in \Lambda} \Delta_\lambda$ of points which *may* induce a transition which cannot be matched in $\mathbf{M}$. Since every Δ_λ is of measure zero, we know that Δ is of measure zero (as a finite product a sets of measure zero).

It follows that the set of behaviours π for which the j th transition cannot be matched is:

$$\{\pi = s_0 \dots s_j \dots s_i \mid j \in [1..i] \wedge s_j \in \Delta\}. \quad (\text{C.10})$$

Moreover, since we work with the Lebesgue measure, which is a product measure, we have:

$$\begin{aligned} & m(\{\pi = s_0 \dots s_j \dots s_i \mid j \in [1..i] \wedge s_j \in \Delta_j\}) \\ & \leq \prod_{j \in [1..i]} m((\Omega \times \Lambda)^{j-1}) \times m(\Delta) \times m((\Omega \times \Lambda)^{n-j}), \\ & \leq \prod_{j \in [1..i]} m((\Omega \times \Lambda)^{j-1}) \times 0 \times m((\Omega \times \Lambda)^{n-j}), \\ & \leq \prod_{j \in [1..i]} 0, \\ & = 0. \end{aligned} \quad (\text{C.11})$$

Thus, the set of all behaviours with at least one transition which cannot be matched in $\mathbf{M}$ is of measure zero, and the proof is complete.

Since the measure of the set of behaviours in $\mathcal{S}$ that cannot be abstracted to $\mathbf{M}$ is zero, it means that the set of behaviours that do not satisfy φ is of measure zero as well, and that the probability that a behaviour in $\mathcal{S}$ induced by $\mathbb{X}$ (picked uniformly at random in the set $\mathcal{B}_{\mathbb{X}}$ of behaviours induced by $\mathbb{X}$, see Eq. (1.25) on page 30) does not satisfy φ is zero; the proof is complete. $\square$

BIBLIOGRAPHY

Publications of the author

- [1] Ardourel, Gilles and Cantin, Guillaume and Delahaye, Benoît and Derroire, G eral-
dine and Funatsu, Beatriz M. and Julien, David, “Computational assessment of
Amazon forest plots regrowth capacity under strong spatial variability for simulat-
ing logging scenarios”, in: *Ecological Modelling* 495 (Sept. 2024), p. 110812, ISSN:
0304-3800, DOI: [10.1016/j.ecolmodel.2024.110812](https://doi.org/10.1016/j.ecolmodel.2024.110812).
- [2] David Julien et al., “End-to-End Statistical Model Checking for Parameterization
and Stability Analysis of ODE Models”, in: *ACM Transactions on Modeling and
Computer Simulation* 34.3 (Mar. 2023), pp. 1–25, ISSN: 1049-3301, DOI: [10.1145/
3649438](https://doi.org/10.1145/3649438), URL: <https://hal.science/hal-04478120>.
- [3] Julien, David and Cantin, Guillaume and Delahaye, Benoît, “End-to-End Statistical
Model Checking for Parametric ODE Models”, in: *QEST: International Conference
on Quantitative Evaluation of Systems*, ed. by Erika  brah am and Marco Paolieri,
vol. 13479, Lecture Notes in Computer Science, Warsaw, Poland: Springer Interna-
tional Publishing, Sept. 2022, pp. 85–106, ISBN: 978-3-031-16336-4, DOI: [10.1007/
978-3-031-16336-4_5](https://doi.org/10.1007/978-3-031-16336-4_5), URL: [https://hal.archives-ouvertes.fr/hal-
03824899](https://hal.archives-ouvertes.fr/hal-03824899).

Bibliography

- [4] Rajeev Alur et al., “Hybrid Automata: An Algorithmic Approach to the Specification and Verification of Hybrid Systems”, in: *Hybrid Systems*, vol. 736, Lecture Notes in Computer Science, Berlin, Germany: Springer-Verlag, Jan. 1993, pp. 209–229, ISBN: 978-354057318, DOI: [10.1007/3-540-57318-6_30](https://doi.org/10.1007/3-540-57318-6_30).
- [5] Benjamin Ambrosio and M. A. Aziz-Alaoui, “On a Coupled Time-Dependent SIR Models Fitting with New York and New-Jersey States COVID-19 Data”, in: *Biology* 9.6 (June 2020), p. 135, ISSN: 2079-7737, DOI: [10.3390/biology9060135](https://doi.org/10.3390/biology9060135).
- [6] William J. Anderson, *Continuous-Time Markov Chains*, New York, NY, USA: Springer, 1991, ISBN: 978-1-4612-3038-0, URL: <https://doi.org/10.1007/978-1-4612-3038-0>.
- [7] David Angeli and Eduardo D. Sontag, “Forward completeness, unboundedness observability, and their Lyapunov characterizations”, in: *Systems & Control Letters* 38.4 (Dec. 1999), pp. 209–217, ISSN: 0167-6911, DOI: [10.1016/S0167-6911\(99\)00055-9](https://doi.org/10.1016/S0167-6911(99)00055-9).
- [8] M.Ya. Antonovsky et al., “Forest-pest interaction dynamics: The simplest mathematical models”, in: *Theoretical Population Biology* 37.2 (1990), pp. 343–367, ISSN: 0040-5809, DOI: [10.1016/0040-5809\(90\)90043-U](https://doi.org/10.1016/0040-5809(90)90043-U), URL: <https://www.sciencedirect.com/science/article/pii/004058099090043U>.
- [9] Christel Baier and Joost-Pieter Katoen, *Principles of model checking*, MIT press, 2008, ISBN: 978-0-262-02649-9, URL: <https://mitpress.mit.edu/9780262026499/principles-of-model-checking/>.
- [10] Adrien Banse et al., *Data-driven memory-dependent abstractions of dynamical systems via a Cantor-Kantorovich metric*, 2024, arXiv: [2405.08353](https://arxiv.org/abs/2405.08353) [eess.SY], URL: <https://arxiv.org/abs/2405.08353>.
- [11] Adrien Banse et al., “Data-driven memory-dependent abstractions of dynamical systems via a Cantor-Kantorovich metric”, in: *IEEE Transactions on Automatic Control* (2025), pp. 1–12, ISSN: 2334-3303, DOI: [10.1109/tac.2025.3582534](https://doi.org/10.1109/tac.2025.3582534).
- [12] Bao, Ran and Attiogbe, Christian and Delahaye, Benoît and Fournier, Paulin and Lime, Didier, “Parametric Statistical Model Checking of UAV Flight Plan”, in: *Formal Techniques for Distributed Objects, Components, and Systems*, Springer In-

-
- ternational Publishing, 2019, pp. 57–74, ISBN: 9783030217594, DOI: [10.1007/978-3-030-21759-4_4](https://doi.org/10.1007/978-3-030-21759-4_4).
- [13] Ivar Bendixson, “Sur les courbes définies par des équations différentielles”, in: *Acta Mathematica* 24.0 (1901), pp. 1–88, ISSN: 0001-5962, DOI: [10.1007/bf02403068](https://doi.org/10.1007/bf02403068).
 - [14] Béatrice Bérard, Patricia Bouyer, and Vincent Jugé, “Finite Bisimulations for Dynamical Systems with Overlapping Trajectories”, in: *27th EACSL Annual Conference on Computer Science Logic (CSL 2018)*, ed. by Dan R. Ghica and Achim Jung, vol. 119, Leibniz International Proceedings in Informatics (LIPIcs), Dagstuhl, Germany: Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2018, 26:1–26:17, ISBN: 978-3-95977-088-0, DOI: [10.4230/LIPIcs.CSL.2018.26](https://doi.org/10.4230/LIPIcs.CSL.2018.26), URL: <https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.CSL.2018.26>.
 - [15] Fischer Black and Myron Scholes, “The Pricing of Options and Corporate Liabilities”, in: *Journal of Political Economy* 81.3 (May 1973), pp. 637–654, ISSN: 1537-534X, DOI: [10.1086/260062](https://doi.org/10.1086/260062).
 - [16] John Charles Butcher, *Numerical methods for ordinary differential equations*, Third Edition, John Wiley & Sons, 2016, ISBN: 978-1-119-12150-3, DOI: [10.1002/9781119121534](https://doi.org/10.1002/9781119121534).
 - [17] Michel Crouzeix and Alain L Mignot, *Analyse numérique des équations différentielles*, vol. 1, Masson, 1984, ISBN: 2-225-77341-6.
 - [18] M. Davis, *The Undecidable: Basic Papers on Undecidable Propositions, Unsolvability Problems and Computable Functions*, Dover books on mathematics, Dover Publications, 2004, ISBN: 9780486432281.
 - [19] Jean-Pierre Demailly, *Analyse numérique et équations différentielles*, Grenoble Sciences, EDP Sciences, 2012, ISBN: 978-2-759-80112-1.
 - [20] Jörg Desel and Gabriel Juhás, ““What Is a Petri Net?” Informal Answers for the Informed Reader”, in: *Unifying Petri Nets*, Springer Berlin Heidelberg, 2001, pp. 1–25, ISBN: 9783540455417, DOI: [10.1007/3-540-45541-8_1](https://doi.org/10.1007/3-540-45541-8_1).
 - [21] Georg Duffing and Fritz Emde, *Erzwungene Schwingungen bei veränderlicher Eigenfrequenz und ihre technische Bedeutung*, Vieweg, 1918.
 - [22] Nelson Dunford and Jacob Theodore Schwartz, *Linear operators. 1. General theory*, A Wiley Interscience Publication, Interscience Publishers, 1967, ISBN: 978-0-471-60848-6.

-
- [23] José Ferreirós, “The Road to Modern Logic—An Interpretation”, *in: Bulletin of Symbolic Logic* 7.4 (Dec. 2001), pp. 441–484, ISSN: 1943-5894, DOI: [10.2307/2687794](https://doi.org/10.2307/2687794).
- [24] Randy A. Freeman and Petar Kokotović, “Robust Control Lyapunov Functions”, *in: Robust Nonlinear Control Design: State-Space and Lyapunov Techniques*, Boston, MA, USA: Birkhäuser, 2008, pp. 33–63, ISBN: 978-0-8176-4759-9, DOI: [10.1007/978-0-8176-4759-9_3](https://doi.org/10.1007/978-0-8176-4759-9_3).
- [25] Peyman Ghaffari, Cristiana J. Silva, and Delfim F. M. Torres, “Mathematical Models and Optimal Control in Mosquito Transmitted Diseases”, *in: Bio-mathematics, Statistics, and Nano-Technologies*, Chapman and Hall/CRC, June 2023, pp. 143–156, ISBN: 9781003035992, DOI: [10.1201/9781003035992-7](https://doi.org/10.1201/9781003035992-7).
- [26] Georges Gonthier, *A computer-checked proof of the Four Color Theorem*, tech. rep., Inria, Mar. 2023, URL: <https://inria.hal.science/hal-04034866>.
- [27] John Guckenheimer and Philip J. Holmes, *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, Springer New York, 1983, ISBN: 9781461211402, DOI: [10.1007/978-1-4612-1140-2](https://doi.org/10.1007/978-1-4612-1140-2).
- [28] Benjamin M. Gyori et al., “Approximate probabilistic verification of hybrid systems”, *in: International Workshop on Hybrid Systems Biology*, vol. 9271, Lecture Notes in Computer Science, Springer, 2015, pp. 96–116, DOI: [10.1007/978-3-319-26916-0_6](https://doi.org/10.1007/978-3-319-26916-0_6).
- [29] Chihiro Hayashi, *Nonlinear Oscillations in Physical Systems*, Princeton, NJ, USA: Princeton University Press, July 2014, ISBN: 978-1-40085287-1, DOI: [10.1515/9781400852871](https://doi.org/10.1515/9781400852871).
- [30] Christian Hensel et al., “The probabilistic model checker Storm”, *in: International Journal on Software Tools for Technology Transfer* 24.4 (Aug. 2022), pp. 589–610, ISSN: 1433-2787, DOI: [10.1007/s10009-021-00633-z](https://doi.org/10.1007/s10009-021-00633-z).
- [31] Thomas A. Henzinger et al., “What’s Decidable about Hybrid Automata?”, *in: Journal of Computer and System Sciences* 57.1 (Aug. 1998), pp. 94–124, ISSN: 0022-0000, DOI: [10.1006/jcss.1998.1581](https://doi.org/10.1006/jcss.1998.1581).
- [32] Wassily Hoeffding, “Probability inequalities for sums of bounded random variables”, *in: Journal of the American Statistical Association* 58.301 (1963), pp. 13–30, DOI: [10.1080/01621459.1963.10500830](https://doi.org/10.1080/01621459.1963.10500830).

-
- [33] Philip J. Holmes and David R. Rand, “Phase portraits and bifurcations of the non-linear oscillator: $\ddot{x} + \alpha\dot{x} + \gamma x^2\dot{x} + \beta x + \delta x^3 = 0$ ”, in: *International Journal of Non-Linear Mechanics* 15.6 (1980), pp. 449–458, ISSN: 0020-7462, DOI: [10.1016/0020-7462\(80\)90031-1](https://doi.org/10.1016/0020-7462(80)90031-1).
- [34] Kenneth A. Johnson and Roger S. Goody, “The Original Michaelis Constant: Translation of the 1913 Michaelis–Menten Paper”, in: *Biochemistry* 50.39 (2011), PMID: 21888353, pp. 8264–8269, DOI: [10.1021/bi201284u](https://doi.org/10.1021/bi201284u), URL: <https://doi.org/10.1021/bi201284u>.
- [35] Sebastian Junges et al., “Parameter synthesis for Markov models: covering the parameter space”, in: *Formal Methods in System Design* 62.1–3 (Feb. 2024), pp. 181–259, ISSN: 1572-8102, DOI: [10.1007/s10703-023-00442-x](https://doi.org/10.1007/s10703-023-00442-x).
- [36] A. B. Kahn, “Topological sorting of large networks”, in: *Communications of the ACM* 5.11 (Nov. 1962), pp. 558–562, ISSN: 0001-0782, DOI: [10.1145/368996.369025](https://doi.org/10.1145/368996.369025).
- [37] Olav Kallenberg, *Foundations of Modern Probability*, Springer International Publishing, 2021, DOI: [10.1007/978-3-030-61871-1](https://doi.org/10.1007/978-3-030-61871-1).
- [38] Malvin H. Kalos and Paula A. Whitlock, *Monte Carlo Methods, 2nd Edition*, Weinheim, Germany: Wiley, June 2009, ISBN: 978-3-527-62622-9, URL: <https://www.wiley.com/en-us/Monte+Carlo+Methods%2C+2nd+Edition-p-9783527626229#description-section>.
- [39] Edward W. Kamen, *Industrial Controls and Manufacturing*, Elsevier, Academic Press, 1999, ISBN: 978-0-12-394850-2, DOI: [10.1016/B978-0-12-394850-2.X5000-0](https://doi.org/10.1016/B978-0-12-394850-2.X5000-0).
- [40] Mahinda Mailagaha Kumbure et al., “Machine learning techniques and data for stock market forecasting: A literature review”, in: *Expert Systems with Applications* 197 (July 2022), p. 116659, ISSN: 0957-4174, DOI: [10.1016/j.eswa.2022.116659](https://doi.org/10.1016/j.eswa.2022.116659).
- [41] M. Kwiatkowska, G. Norman, and D. Parker, “PRISM 4.0: Verification of Probabilistic Real-time Systems”, in: *Proc. 23rd International Conference on Computer Aided Verification (CAV’11)*, ed. by G. Gopalakrishnan and S. Qadeer, vol. 6806, LNCS, Springer, 2011, pp. 585–591.

-
- [42] Gerardo Lafferriere, George J. Pappas, and Shankar Sastry, “O-Minimal Hybrid Systems”, in: *Mathematics of Control, Signals and Systems* 13.1 (Feb. 2000), pp. 1–21, ISSN: 1435-568X, DOI: [10.1007/PL00009858](https://doi.org/10.1007/PL00009858).
- [43] Gerardo Lafferriere, George J. Pappas, and Sergio Yovine, “A New Class of Decidable Hybrid Systems”, in: *Hybrid Systems: Computation and Control*, Berlin, Germany: Springer, Apr. 1999, pp. 137–151, ISBN: 978-3-540-48983-2, DOI: [10.1007/3-540-48983-5_15](https://doi.org/10.1007/3-540-48983-5_15).
- [44] Abolfazl Lavaei et al., “Constructing MDP abstractions using data with formal guarantees”, in: *IEEE Control Systems Letters* 7 (2022), pp. 460–465, DOI: [10.1109/LCSYS.2022.3188535](https://doi.org/10.1109/LCSYS.2022.3188535).
- [45] John M. Lee, *Introduction to Topological Manifolds*, Springer New York, 2011, ISBN: 9781441979407, DOI: [10.1007/978-1-4419-7940-7](https://doi.org/10.1007/978-1-4419-7940-7).
- [46] Legay, Axel and Delahaye, Benoît and Bensalem, Saddek, “Statistical model checking: An overview”, in: *International Conference on Runtime Verification*, Springer, 2010, pp. 122–135, DOI: [10.1007/978-3-642-16612-9_11](https://doi.org/10.1007/978-3-642-16612-9_11).
- [47] Bing Liu, Benjamin M. Gyori, and P. S. Thiagarajan, *Statistical Model Checking based Analysis of Biological Networks*, 2018, DOI: [10.48550/ARXIV.1812.01091](https://doi.org/10.48550/ARXIV.1812.01091).
- [48] Loève, M., *Probability Theory I*, Springer New York, 1977, ISBN: 9781468494648, DOI: [10.1007/978-1-4684-9464-8](https://doi.org/10.1007/978-1-4684-9464-8).
- [49] Max Lotkin, “On the accuracy of Runge-Kutta’s method”, in: *Mathematical Tables and Other Aids to Computation* 5.35 (1951), pp. 128–133, DOI: [10.1090/S0025-5718-1951-0043566-3](https://doi.org/10.1090/S0025-5718-1951-0043566-3).
- [50] Aleksandr Mikhailovich Lyapunov, *General problem of the stability of motion*, Control Theory and Applications Series, Taylor & Francis, 1992, ISBN: 9780748400621.
- [51] Pertti Mattila, *Geometry of Sets and Measures in Euclidean Spaces: Fractals and Rectifiability*, Cambridge Studies in Advanced Mathematics, Cambridge University Press, 1995.
- [52] Valentina Melica, Sergio Invernizzi, and Gabriella Caristi, “Logistic density-dependent growth of an *Aurelia aurita* polyps population”, in: *Ecological Modelling* 291 (2014), pp. 1–5, ISSN: 0304-3800, DOI: [10.1016/j.ecolmodel.2014.07.009](https://doi.org/10.1016/j.ecolmodel.2014.07.009).
- [53] John Milnor, “On the concept of attractor”, in: *Communications in Mathematical Physics* 99.2 (June 1985), pp. 177–195, ISSN: 1432-0916, DOI: [10.1007/bf01212280](https://doi.org/10.1007/bf01212280).

-
- [54] Nikolai Minorsky, *Nonlinear Oscillations*, Van Nostrand, 1962.
- [55] Anna del Moral Méndez et al., “Toward Improved Short-Term Forecasting for Lake Victoria Basin. Part I: A Radar-Based Convective Mode Analysis”, in: *Weather and Forecasting* 38.12 (2023), pp. 2509–2526, DOI: [10.1175/WAF-D-23-0039.1](https://doi.org/10.1175/WAF-D-23-0039.1), URL: <https://journals.ametsoc.org/view/journals/wefo/38/12/WAF-D-23-0039.1.xml>.
- [56] Faïçal Ndaïrou et al., “Mathematical modeling of COVID-19 transmission dynamics with a case study of Wuhan”, in: *Chaos, Solitons & Fractals* 135 (June 2020), p. 109846, ISSN: 0960-0779, DOI: [10.1016/j.chaos.2020.109846](https://doi.org/10.1016/j.chaos.2020.109846), URL: <http://dx.doi.org/10.1016/j.chaos.2020.109846>.
- [57] Charles F. Nims, “The Bread and Beer Problems of the Moscow Mathematical Papyrus”, in: *The Journal of Egyptian Archaeology* 44.1 (Dec. 1958), pp. 56–65, ISSN: 2514-0582, DOI: [10.1177/030751335804400112](https://doi.org/10.1177/030751335804400112).
- [58] William Noel et al., *The Archimedes Palimpsest*, ed. by William Noel et al., The Archimedes palimpsest publications, Cambridge [u.a.]: Cambridge Univ. Press, 2011, ISBN: 9781107016842.
- [59] John Oprea, “Geometry and the Foucault Pendulum”, in: *The American Mathematical Monthly* 102.6 (June 1995), pp. 515–522, ISSN: 1930-0972, DOI: [10.1080/00029890.1995.12004611](https://doi.org/10.1080/00029890.1995.12004611).
- [60] Stefania Ottaviano, Mattia Sensi, and Sara Sottile, “Global stability of SAIRS epidemic models”, in: *Nonlinear Analysis: Real World Applications* 65 (June 2022), p. 103501, ISSN: 1468-1218, DOI: [10.1016/j.nonrwa.2021.103501](https://doi.org/10.1016/j.nonrwa.2021.103501).
- [61] Lawrence Perko, *Differential equations and dynamical systems*, Texts in Applied Mathematics, Springer New York, 2013, ISBN: 978-1-461-30003-8, DOI: [10.1007/978-1-4613-0003-8](https://doi.org/10.1007/978-1-4613-0003-8).
- [62] Valentin Vladimirovitch Petrov, *Sums of independent random variables*, De Gruyter, 1975, ISBN: 978-3-112-57300-6, DOI: [10.1515/9783112573006](https://doi.org/10.1515/9783112573006).
- [63] Henri Poincaré, *Mémoire sur les courbes définies par une équation différentielle*, Facsim. de textes extr. de diverses revues et publications, 1856-1921, Ursprüngliches Erscheinungsjahr: 1856-1921, Paris: Gabay, 1993, 550 pp., ISBN: 2876470985.

-
- [64] Tim Quatmann et al., “Parameter Synthesis for Markov Models: Faster Than Ever”, in: *ATVA: Automated Technology for Verification and Analysis*, Cham, Switzerland: Springer, Sept. 2016, pp. 50–67, ISBN: 978-3-319-46520-3, DOI: [10.1007/978-3-319-46520-3_4](https://doi.org/10.1007/978-3-319-46520-3_4).
- [65] Simon Ramondenc et al., “Probabilistic modeling to estimate jellyfish ecophysiological properties and size distributions”, in: *Scientific Reports* 10.1 (Apr. 2020), ISSN: 2045-2322, DOI: [10.1038/s41598-020-62357-5](https://doi.org/10.1038/s41598-020-62357-5).
- [66] Daniel Richardson, “Some Undecidable Problems Involving Elementary Functions of a Real Variable on JSTOR”, in: *Journal of Symbolic Logic* 33.4 (Dec. 1968), [Online; accessed 29. Feb. 2024], pp. 514–520, URL: <https://www.jstor.org/stable/2271358>.
- [67] Marguerite Robinson and Nikolaos I. Stilianakis, “A model for the emergence of drug resistance in the presence of asymptomatic infections”, in: *Mathematical Biosciences* 243.2 (June 2013), pp. 163–177, ISSN: 0025-5564, DOI: [10.1016/j.mbs.2013.03.003](https://doi.org/10.1016/j.mbs.2013.03.003).
- [68] M. L. Rosenzweig and R. H. MacArthur, “Graphical Representation and Stability Conditions of Predator-Prey Interactions”, in: *The American Naturalist* 97.895 (July 1963), pp. 209–223, ISSN: 1537-5323, DOI: [10.1086/282272](https://doi.org/10.1086/282272), eprint: <https://doi.org/10.1086/282272>, URL: <https://doi.org/10.1086/282272>.
- [69] David Ruelle and Floris Takens, “On the nature of turbulence”, in: *Communications in Mathematical Physics* 20.3 (Sept. 1971), pp. 167–192, ISSN: 1432-0916, DOI: [10.1007/bf01646553](https://doi.org/10.1007/bf01646553).
- [70] Cristiana J. Silva et al., “Complex network model for COVID-19: Human behavior, pseudo-periodic solutions and multiple epidemic waves”, in: *Journal of Mathematical Analysis and Applications* 514.2 (Oct. 2022), p. 125171, ISSN: 0022-247X, DOI: [10.1016/j.jmaa.2021.125171](https://doi.org/10.1016/j.jmaa.2021.125171), URL: <http://dx.doi.org/10.1016/j.jmaa.2021.125171>.
- [71] Jeremy Sproston, “Verification and Control of Probabilistic Rectangular Hybrid Automata”, in: *Formal Modeling and Analysis of Timed Systems*, Cham, Switzerland: Springer, Aug. 2015, pp. 1–9, ISBN: 978-3-319-22975-1, DOI: [10.1007/978-3-319-22975-1_1](https://doi.org/10.1007/978-3-319-22975-1_1).

-
- [72] Endre Süli and David F. Mayers, *An Introduction to Numerical Analysis*, Cambridge University Press, Aug. 2003, ISBN: 9780521810265.
- [73] Paulo Tabuada, *Verification and Control of Hybrid Systems*, Springer US, 2009, ISBN: 978-1-4419-0224-5, DOI: [10.1007/978-1-4419-0224-5](https://doi.org/10.1007/978-1-4419-0224-5).
- [74] Stanley B. Trier et al., “Composite Mesoscale Environmental Conditions Influencing Tornado Frequencies in Landfalling Tropical Cyclones”, in: *Weather and Forecasting* 38.12 (2023), pp. 2481–2508, DOI: [10.1175/WAF-D-22-0227.1](https://doi.org/10.1175/WAF-D-22-0227.1), URL: <https://journals.ametsoc.org/view/journals/wefo/38/12/WAF-D-22-0227.1.xml>.
- [75] Ferdinand Verhulst, *Nonlinear differential equations and dynamical systems*, Springer Berlin, 1996, DOI: [10.1007/978-3-642-61453-8](https://doi.org/10.1007/978-3-642-61453-8).
- [76] W. Weaver Jr., S. P. Timoshenko, and D. H. Young, *Vibration Problems in Engineering, 5th Edition*, Hoboken, NJ, USA: Wiley, Jan. 1991, ISBN: 978-0-471-63228-3, URL: <https://www.wiley.com/en-us/Vibration+Problems+in+Engineering%2C+5th+Edition-p-9780471632283>.
- [77] Håkan L. S. Younes, “Verification and Planning for Stochastic Processes with Asynchronous Events”, PhD thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, Pennsylvania, 2005.
- [78] Håkan L. S. Younes and Reid G. Simmons, “Probabilistic Verification of Discrete Event Systems Using Acceptance Sampling”, in: *Proceedings of the 14th International Conference on Computer Aided Verification, CAV ’02*, Berlin, Heidelberg: Springer-Verlag, 2002, pp. 223–235, ISBN: 3540439978.
- [79] Håkan L. S. Younes et al., “Numerical vs. statistical probabilistic model checking”, in: *International Journal on Software Tools for Technology Transfer* 8.3 (June 2006), pp. 216–228, ISSN: 1433-2787, DOI: [10.1007/s10009-005-0187-8](https://doi.org/10.1007/s10009-005-0187-8).

Titre : Abstraction et vérification probabilistes de systèmes dynamiques hybrides

Mot clés : Équations différentielles - Systèmes hybrides - Processus Markoviens - Simulation

Résumé : Cette thèse porte sur la vérification formelle de systèmes décrits par des équations différentielles ordinaires. Deux résultats principaux peuvent être distingués. Premièrement, l'adaptation de la Vérification Statistique de Modèles (*Statistical Model Checking* ou SMC en anglais) au contexte des équations différentielles, via la majoration de l'erreur introduite par les techniques de résolutions de ces équations sur l'ensemble des

simulations possibles. Deuxièmement, l'abstraction de systèmes dits hybrides, dont la dynamique décrite par une équation différentielle peut être modifiée au cours du temps, sous la forme de processus Markovien : ceci permet de dépasser certaines limites imposées par le formalisme originel. Chacune des deux techniques présentées fait l'objet d'un exemple d'application complet.

Title: Probabilistic Abstraction and Verification of Hybrid Dynamical Systems

Keywords: Ordinary Differential Equations - Hybrid systems - Markov processes - Simulation

Abstract: This thesis presents the formal verification of systems described with ordinary differential equations. Two main results may be distinguished. First, the adaptation of Statistical Model Checking (SMC) to the context of differential equations, by bounding the error introduced by integrations methods for the resolution of these equations on the whole simu-

lation space. Second, the abstraction of so-called hybrid systems, whose dynamics is described by a differential equations which can be modified during the simulation, as a Markov process: this allows to overcome some limitations imposed by the original formalism. Each of the two presented techniques is completely unfolded on a case study.