



HAL
open science

End-to-End Statistical Model Checking for Parametric ODE Models

David Julien, Guillaume Cantin, Benoît Delahaye

► **To cite this version:**

David Julien, Guillaume Cantin, Benoît Delahaye. End-to-End Statistical Model Checking for Parametric ODE Models. QEST: International Conference on Quantitative Evaluation of Systems, Sep 2022, Warsaw, Poland. pp.85-106, 10.1007/978-3-031-16336-4_5 . hal-03824899

HAL Id: hal-03824899

<https://hal.science/hal-03824899v1>

Submitted on 21 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

End-to-end Statistical Model Checking for Parametric ODE Models

David JULIEN^{1,2}[0000–0002–1592–8042], Guillaume CANTIN¹[0000–0001–5122–1194] and Benoît DELAHAYE^{1,*}[0000–0002–9104–4361]

¹ Nantes Université, École Centrale Nantes, CNRS, LS2N, UMR 6004, F-44000 Nantes, France

² École Normale Supérieure de Lyon, 69007 Lyon, France

* benoit.delahaye@univ-nantes.fr

Abstract. We propose a simulation-based technique for the verification of structural parameters in Ordinary Differential Equations. This technique is an adaptation of Statistical Model Checking, often used to verify the validity of biological models, to the setting of Ordinary Differential Equations systems. The aim of our technique is to search the parameter space for the parameter values that induce solutions that best fit experimental data under variability, with any metrics of choice. To do so, we discretize the parameter space and use statistical model checking to grade each individual parameter value w.r.t experimental data. Contrary to other existing methods, we provide statistical guarantees regarding our results that take into account the unavoidable approximation errors introduced through the numerical resolution of the ODE system performed while simulating. In order to show the potential of our technique, we present its application to two case studies taken from the literature, one relative to the growth of a jellyfish population, and another concerning a prey-predator model.

Keywords: Statistical Model Checking · ODE models · structural parameters.

1 Introduction

All scientific branches share the common concept of modeling. When a scientist studies a real-life system, the first step he or she goes through is to build a model that gathers all the existing knowledge of the target system. This model is then used as a proxy of the system it represents in order to analyze it, perform simulation or predictions. In several fields, such as Biology, Chemistry, Physics or Engineering, models do not represent a single system but are instead an abstraction for a *family* of systems that share common traits but might exhibit some internal variability. This internal variability can either be left out by considering that the model represents the “average” individual in the family, or taken into account inside of the model through the use of non-determinism, probabilities or parametricity.

When considering parametric models, scientists have to go through a phase of parameterization, which consists in confronting the model with experimental observations of the (family of) system(s) it represents in order to find the

parameter values that best fit this (family of) system(s). In most cases, parameterization techniques are *deterministic* [21, 24]. They lead to deterministic parameter values that best fit the experimental data, i.e. producing the best fit for the “average” individual. In this paper, we instead focus on a technique that allows to select parameter values that best fit *under variability*, i.e. that produce the best *probabilistic* fit for the *whole family*.

Parameterization, or *parameter synthesis* has been the topic of many works in the context of probabilistic systems [6, 9-11, 13]. Symbolic techniques such as parametric model checking [1, 5] are often difficult to use in practice because they require automata-based models while real-life models are often expressed either with computer programs or with differential equation models. Statistical Model Checking (SMC) [14], on the other hand, is a simulation-based technique that allows to estimate, with formal guarantees, the probability that a given (probabilistic) model satisfies a given property. Because it is simulation-based, it can be applied to any stochastic model for which simulations can be performed. SMC has been successfully applied to perform parameterization of real-life models expressed using several formalisms such as parametric Markov chains [2], parametric Python programs [20], or even parametric Ordinary Differential Equation systems (ODEs) [15]. Unfortunately, the formal guarantees obtained through SMC are linked to the simulation space (i.e. the produced traces) and not to the original model itself. When the model consists in sets of ODEs, as in [15], numerical resolution methods are used in order to solve the ODEs and perform simulations, which means that the formal guarantees obtained through SMC cannot apply to the original ODE model.

In this paper, our main contribution is to bridge the gap between the original ODE model and the results of the parameterization procedure by *combining the statistical guarantees of SMC with the global approximation error of standard numerical resolution methods*. As in [15], we consider ODE models with structural parameters. We assume that these models represent families of real-life systems that need to match some experimental data through simulation. We build on the logic proposed in [15] to express our properties of interest and also consider expected reward properties that might be of interest in practice. We use SMC to grade parameter values by estimating the expectation of a given reward function for these values while taking internal variability into account. Contrarily to what is done in [15], the accuracy of this estimation is guaranteed w.r.t. the original ODE model.

To illustrate our results, we perform the parameterization of two state-of-the-art models taken from the literature using our technique. In this context, and because modelers are often interested by this information in practice, we propose a global evaluation of the parameter space that allows us to get a complete picture of the adequacy of the parameter values w.r.t. the given experimental data. This choice is done by interest only, since our results are generic and could be applied to any search technique, such as the local ones performed in [15].

Intuition. To give an intuition of our contribution, we provide an informal summary of the method we present in this paper. Recall that, given a dataset

relative to an experiment and a parametric ODE system, the objective is to find a solution to a parametric ODE system (i.e. parameter values) that satisfies a property φ w.r.t. the dataset, which is, given a distance $\delta > 0$, “the solution stays in a tunnel of radius δ around the experimental data”; we also want to acquire statistical guarantees on said result. The main issue is that we can only simulate our model by solving the ODE system using numerical resolution methods. Hence, we cannot directly verify whether exact solutions (z) of the system satisfy φ and instead have to rely on approximate solutions (y). We therefore proceed as follows: we start by discretizing the set of parameter values into a grid; we then evaluate each point of this grid using the procedure detailed below; finally, we use the resulting scores to select the “best” parameter values w.r.t φ . The score of a given parameter value λ is computed as follows, and illustrated in Figures 1 and 2 in the context of the case study presented in section 4.1.

1. We set the parameter value to λ . By a careful study of the ODE system, we give a bound on the distance ε between exact (z) and approximate (y) solutions. We emphasize that this bound depends on (1) the chosen resolution technique and (2) the chosen integration step. We show that this distance is uniformly stable w.r.t. internal variability around λ , but also that it can be uniformly bounded on the global set of solutions (i.e. independently of λ).
2. We propose two new properties φ_1 and φ_2 that will be verified on the approximate solutions y , and depend on the above distance. This amounts to changing the size of the tunnel around the experimental dataset. We compute (estimations of) the respective probabilities p_1 and p_2 and prove that the probability p that z satisfies φ lies between p_1 and p_2 .
3. We provide statistical guarantees of our estimation, i.e. a confidence interval for our estimation of p , and use this estimation as the score for parameter value λ .

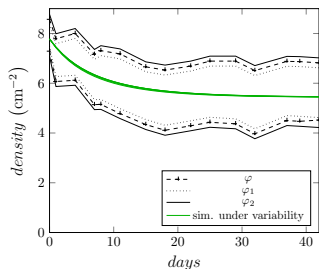


Fig. 1. Tunnels corresponding to the properties $\varphi, \varphi_1, \varphi_2$ and accepted simulations.

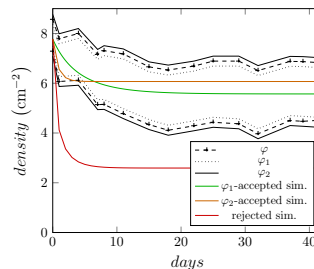


Fig. 2. φ -accepted, φ_2 -accepted and rejected solutions.

It is worth noting that the underlying theory is generic: the integration method as well as the statistical estimation method can be chosen arbitrarily as

long as they provide the usual guarantees. In this paper, we use Runge-Kutta and Monte-Carlo for the sake of example.

Outline. In section 2, we introduce required preliminaries and notations for the rest of the paper. In section 3, we state the main result of the paper, i.e. we compute the approximation error for ODE solutions, show that this error is uniformly stable, and provide the statistical guarantees for the estimation of the probabilities. In section 4, we illustrate our approach on two case studies taken from the literature. Finally, we conclude in section 5 and give perspectives for future work.

2 Background and notations

In this section, we present the basic notations and definitions that will be used throughout the paper. More precisely, we recall the definition of an ODE, and present the logic used in the paper. Finally, we extend this logic by introducing reward functions.

2.1 ODE preliminaries

First, we consider an evolution problem described by an Ordinary Differential Equation (ODE) of the form

$$\frac{dz}{dt}(t) = f(z(t), \boldsymbol{\lambda}), \quad t > 0. \quad (1)$$

In eq. (1), the unknown function z is defined in \mathbb{R}^+ with values in \mathbb{R}^n ; $\boldsymbol{\lambda} \in \mathbb{R}^m$ is a vector of parameters; f is a function defined on $\mathbb{R}^n \times \mathbb{R}^m$ with values in \mathbb{R}^n , whose regularity will be detailed below; n, m are positive integers. In the following, we write $z_i(t)$, $1 \leq i \leq n$, for the projection of $z(t)$ on its i th component. As mentioned in our introduction, eq. (1) can model various real-world problems arising in life sciences. Our goal is to study some properties of the trajectories determined by eq. (1), by developing an innovative model-checking framework suitable for the continuous dynamics of ODEs.

Here and for the rest of the paper, we fix an initial condition $z_0 \in \mathbb{R}^n$. Standard results of the theory of differential equations (see for instance [18]) ensure that, for any value of the parameter $\boldsymbol{\lambda} \in \mathbb{R}^m$, the Cauchy problem determined by eq. (1) and the initial value $z(0) = z_0$ admits a unique solution, provided f is C^1 on $\mathbb{R}^n \times \mathbb{R}^m$; we denote by $z^\lambda(t)$ the corresponding trajectory, which we assume to be defined on $[0, T]$ with $T > 0$. If the context is sufficiently clear, we may write $z(t)$ for short. As before, we write $z_i^\lambda(t)$ (resp. $z_i(t)$), $1 \leq i \leq n$, for the projection on its i th component. We assume that the component λ_j of the parameter vector $\boldsymbol{\lambda} \in \mathbb{R}^m$ ($1 \leq j \leq m$) satisfies $\lambda_j \in [L_j^\lambda, U_j^\lambda]$, with real coefficients $L_j^\lambda < U_j^\lambda$ and we consider the compact sets \mathbf{W} and INIT defined by

$$\mathbf{W} = \prod_{j=0}^m [L_j^\lambda, U_j^\lambda] \quad (2) \quad \text{INIT} = \{z_0\} \times \mathbf{W}. \quad (3)$$

For $\boldsymbol{\lambda} \in W$, we consider the Euclidian norm defined by $\|\boldsymbol{\lambda}\| = \left(\sum_{j=1}^m |\lambda_j|^2\right)^{1/2}$.

We assume that the trajectories of eq. (1) starting from z_0 admit a rectangular invariant region, uniform w.r.t. the parameter λ , that is $z_i(t, \lambda) \in [L_i, U_i]$ for $t \in [0, T]$, with real coefficients $L_i < U_i$, for all $\lambda \in \mathbf{W}$ and for $1 \leq i \leq n$. The global invariant region for z is written $\mathbf{V} = \prod_{i=0}^n [L_i, U_i]$.

Finally, we write TRAJ for the set of all potential trajectories of the solutions to our ODE system. Formally, $\text{TRAJ} = \{z^\lambda(t) \mid \lambda \in \mathbf{W}\}$.

It is well-known that eq. (1) determines time continuous trajectories, which moreover depend continuously on a variation of the initial condition z_0 and of the parameter $\lambda \in \mathbf{W}$ (see for instance [18]). In section 3, we will be interested in the variation of those trajectories under a variation of the parameter $\lambda \in \mathbf{W}$. We now move to the description of the properties for our models.

2.2 Bounded Linear Time Logic

As explained in the introduction, our aim is to find the parameter values that allow our model to best fit some given experimental data. In the following, we therefore assume that we are given a finite set of experimental observations that correspond to a finite set of time points and a tolerance value $\delta > 0$ ¹. We write

$$\mathcal{T} = \{0 = t_0, t_1, \dots, t_N = T\} \quad (4)$$

for a set of time points and assume that we have a finite set of observations $\mathcal{O}_i^t = \{o_{i,1}^t, \dots, o_{i,k}^t\}$ for each of them and for each coordinate i . We assume, in practice, that \mathcal{T} indeed includes all the time points where experimental observations are available. Remark nonetheless that \mathcal{T} is not necessarily limited to this set, as we could have $\mathcal{O}^t = \emptyset$ for a number of $t \in \mathcal{T}$. In practice, since $\mathcal{T} = \{t_0, \dots, t_N\}$ is finite, we abuse notations and substitute it, when convenient, with the integer set $\mathcal{T} = \{0, \dots, N\}$.

We start by recalling the logic defined in [15], which allows to express our properties of interest, i.e. that a given solution agrees with the experimental observations available at given time points. This logic is a slightly modified version of Bounded LTL, where atomic propositions are of the form (i, l, u) with $L_i \leq l \leq u \leq U_i$, where L_i and U_i are the boundaries of the set \mathbf{V} defined above. The intuition is that, for $q \in \mathcal{T}$, z satisfies the atomic proposition (i, l, u) at time point q if and only if $l \leq z_i(q) \leq u$. Since there is a finite number of time points and a finite number of observations, we only consider the finite number of atomic propositions where $1 \leq i \leq n$ and $l, u \in \cup_{q \in \mathcal{T}} (\mathcal{O}_i^q \cup \mathcal{O}_i^q - \delta \cup \mathcal{O}_i^q + \delta)$. We also allow $l, u = +\infty, -\infty$ to account for timepoints q where $\mathcal{O}^q = \emptyset$.

The rest of the logic is defined as usual:

- every atomic proposition and the constants *true*, *false* are BLTL formulas,
- the negation and conjunction of BLTL formulas are BLTL formulas,
- if Ψ and Ψ' are BLTL formulas, then $\Psi \mathcal{U}^q \Psi'$ and $\Psi \mathcal{U}^{\leq q} \Psi'$ are BLTL formulas for any positive integer $q \in \mathcal{T}$,
- if Ψ is a BLTL formula, then $\mathcal{X} \Psi$ is a BLTL formula.

¹ Note that the method does not depend on the value of δ . We assume its value is provided by the user.

The interpretation of $\Psi \mathcal{U}^{\leq q} \Psi'$ is standard, i.e. Ψ' must happen before q time points have elapsed, while the interpretation of $\Psi \mathcal{U}^q \Psi'$ is that Ψ must hold for exactly q time points before Ψ' holds. The interpretation of $\mathcal{X} \Psi$ is standard as well, i.e. $\mathcal{X} \Psi \Leftrightarrow \text{true} \mathcal{U}^1 \Psi$. We invite the interested reader to consult [15] for the formal semantics of this logic.

Given a BLTL formula Ψ , we define $\text{models}(\Psi) = \{z \in \text{TRAJ} \mid z, 0 \models \Psi\}$.

Recall that the properties we want our models to verify are the following: the traces of the model need to agree with the given experimental data. One way to rephrase this property is as follows: at all time points where experimental data is available, the trace of our model needs to be between the lower and upper values taken from the experimental data with a given tolerance $\delta > 0$. This is easily expressed in BLTL as the property

$$\Psi^* = \bigwedge_{1 \leq i \leq m} \psi_i^0 \wedge \mathcal{X}(\psi_i^1 \wedge \mathcal{X}(\psi_i^2 \wedge \dots \wedge \mathcal{X} \psi_i^N) \dots) \quad (5)$$

where $\psi_i^q = (i, \min(\mathcal{O}_i^q) - \delta, \max(\mathcal{O}_i^q) + \delta)$. Since our aim is to consider variability on the ODE models of interest, we may use statements of the form $\mathbb{P}_{\geq p}(\Psi^*)$, whose interpretation is expressed as follows: “the probability that a trajectory in TRAJ is in $\text{models}(\Psi^*)$ is greater than p ”. In this regard, we need to define a probability measure \mathbb{P} over TRAJ .

We start by noticing that each parameter value λ completely determines the trajectory $z^\lambda \in \text{TRAJ}$, since the initial condition $z_0 \in \mathbb{R}^n$ has been fixed. As a consequence, TRAJ can be completely identified with INIT (see eq. (3)). Formally, we define $\text{Models}(\Psi) \subseteq \text{INIT}$ as the set

$$\{(z_0, \lambda) \in \text{INIT} \mid z^\lambda(t) \in \text{models}(\Psi)\} \quad (6)$$

and consider the Σ -algebra \mathcal{B} generated by the m -dimensional open intervals of INIT . As expected, it is shown in [15] that \mathcal{B} is an adequate support to prove the measurability of $\text{Models}(\Psi)$ for any BLTL formula Ψ .

In the following, we will consider a number of probability distributions \mathbb{P}^λ on \mathcal{B} (one for each parameter value λ), and use these probability distributions to evaluate whether our ODE model meets a specification of the form $\mathbb{P}_{\geq p}^\lambda(\Psi)$. This will amount to checking whether $\mathbb{P}^\lambda(\text{Models}(\Psi)) \geq p$. We will refer to the formulas such as $\mathbb{P}_{\geq p}^\lambda(\Psi)$ as PBLTL formulas.

In our context, each parameter value λ in \mathbf{W} will give rise to a probability distribution \mathbb{P}^λ taking into account internal variability. This probability distribution will be used for evaluating the model against the property Ψ^* , which will yield a score $\text{grade}(\lambda)$ that represents the adequacy of parameter value λ w.r.t. the given experimental data while taking into account internal variability.

However, it might happen that many of the values λ in INIT have a maximal score $\text{grade}(\lambda) = 1$, i.e. satisfy the PBLTL property $\mathbb{P}_{\geq 1}^\lambda(\Psi^*)$. This could be the case for example if all the traces generated using \mathbb{P}^λ satisfy the property Ψ^* . In this case, we will need to consider more complex properties to filter those values and rank them. To this purpose, we introduce the notion of reward function.

2.3 Reward function

The purpose of statistical model checking in general, and Monte-Carlo in particular, which will be presented in detail in section 3.2, is to estimate with formal guarantees the expected value of a given function on a measurable set. In the context of model checking, this procedure is used to estimate the probability that a given model satisfies a property. To do this, each sample of the system is checked against the property and a Boolean reward is computed accordingly (i.e. 1 if the property is satisfied and 0 otherwise). Statistical model checking then amounts to estimating the expected value of this particular reward function on the measurable set of traces of the model at hand.

In our case, this boils down to defining a reward function $r_{\Psi^*} : \text{TRAJ} \rightarrow \{0, 1\}$ that evaluates to 1 if the trajectory satisfies Ψ^* and 0 otherwise. Statistical model checking will then compute an estimation of the expected value of r_{Ψ^*} on the set of traces TRAJ under the probability distribution \mathbb{P}^λ , which in the end will be an estimation of the measure of $\text{Models}(\Psi^*)$ for the parameter value λ . Remark that this construction would work for any other BLTL property Ψ .

In order to grade the parameter values in a more discriminating way, we allow the use of non-Boolean reward functions. This will allow expressing more powerful properties than those that can be defined using the BLTL logic. For instance, one can use those reward functions in order to measure the number of time points for which the current trace does not agree with the given experimental data, or to measure the cumulative distance between the trace and the experimental data at all time points.

In the following, we will therefore consider a given reward function $r : \text{TRAJ} \rightarrow \mathbb{R}$ and use statistical model checking to estimate its expected value on the trajectories of our model under a given probability distribution \mathbb{P} . When convenient, we will identify a given BLTL property Ψ with its associated reward function obtained through the above construction r_Ψ .

3 Global statistical guarantees

In this section, we state our main result, which provides statistical guarantees on the verification of specific properties. Namely, given a property Ψ (resp. the corresponding reward function r_Ψ) on the trajectories of eq. (1), we will establish confidence intervals regarding the estimation of the *probability* of satisfaction of that property (resp. the *expected value* of r_Ψ), which shall be computed using *approximate solutions* to eq. (1), as well as a bound on the errors w.r.t. the *exact probability* corresponding to the *exact solutions* to eq. (1).

We start by recalling how ODE numerical resolution methods work, and we propose a definition for the approximation errors introduced in the process. Next, we introduce a method for estimating the probability p that exact solutions of our ODE system satisfy a given property Ψ using an estimator \hat{p} that takes the approximation error into account. Finally, we explain how those results along with their statistical guarantees can be extended to the estimation of the expected values of given reward functions.

3.1 Approximation method for the numerical integration of the ODE

We recall that an approximation method, which determines the approximate solution y^λ to the ODE induced by parameter λ , can be written

$$y^\lambda(0) = z_0, \quad y^\lambda(\tau_{j+1}) = y^\lambda(\tau_j) + h\Phi(\tau_j, y^\lambda(\tau_j), \lambda, h), \quad 0 \leq j < J, \quad (7)$$

where Φ is a continuous function defined in $[0, T] \times \mathbb{R}^n \times \mathbf{W} \times \mathbb{R}$ with values in \mathbb{R}^n , τ_j are the discrete points of definition of y^λ , and $h \in \mathbb{R}$. Intuitively, those methods compute each point thanks to the previous one. In this paper, we use the well-known Runge-Kutta 4 method, which is a standard method for ODE resolution.

For the sake of simplicity, we focus in the following on the theoretical study of 1-dimensional systems ($n = 1$), but our method can be adapted to larger systems ($n \geq 2$) as shown in our second case study presented in section 4.2, mostly by adapting the definition of distance introduced below.

As explained in section 2, we consider a set γ of observation data samples, recorded at $(N + 1)$ time points forming a set \mathcal{T} (see eq. (4)) with values in \mathbb{R}^n ($N > 0$). We start by defining a notion of distance between functions that will, in the end, allow us to compare the solutions of our ODE model with the given experimental data. Given any two functions y, \tilde{y} in the set $F_{\mathcal{T}} = \{g : I \rightarrow \mathbb{R} \mid \mathcal{T} \subseteq I\}$, where I denotes an interval included in \mathbb{R} , we consider the distance d defined by

$$d(y, \tilde{y}) = \max_{t \in \mathcal{T}} |y(t) - \tilde{y}(t)|. \quad (8)$$

Note that d is rigorously only a *pseudo*-distance, since two functions y and \tilde{y} defined on $[0, T]$, that are distinct on $[0, T]$, might coincide on the finite set \mathcal{T} , thus could satisfy $d(y, \tilde{y}) = 0$. Nevertheless, since our purpose is to measure the distance to the dataset γ , we do not need to distinguish such two functions. Moreover, one may use any (pseudo-)distance of their choice, since all norms are equivalent in the finite-dimensional space \mathbb{R} (\mathbb{R}^n in the general setting). In the rest of the paper, we will abuse notations and use d to compare a given function $y \in F_{\mathcal{T}}$ to γ , even though γ is only defined on \mathcal{T} and not on a continuous subinterval of \mathbb{R} .

In most ODE resolution methods, the approximation error depends on an integration step. We therefore introduce a discretization \mathcal{D}_h of the time interval $[0, T]$, which we assume, for simplicity, to admit a constant step $h > 0$:

$$\mathcal{D}_h = \{0 = \tau_0, \tau_1, \tau_2, \dots, \tau_J = T\}, \quad (9)$$

with $J > 0$ and $\tau_{j+1} - \tau_j = h$ for all $0 \leq j < J$.

For each parameter value $\lambda \in \mathbf{W}$, the chosen approximation method will compute an approximate solution to the ODE, which we denote y^λ . Recall that the initial condition $z_0 \in \mathbb{R}$ (\mathbb{R}^n in the general setting) has been fixed and that for any $\lambda \in \mathbf{W}$, the exact solution to eq. (1) such that $z(0) = z_0$ is written z^λ .

For the sake of measuring the approximation error between y^λ and z^λ , we use a finer notion of distance than the one proposed above. Indeed, standard

resolution methods provide guarantees that depend on the integration step in the sense that choosing a finer integration step enhances the quality of the approximation. Our aim here is to be able to take advantage of this fact, which could not be captured if we used the distance d from eq. (8).

Definition 1 (Global approximation error). *Let $h > 0$ be the integration step of the chosen resolution method. The global approximation error $\epsilon_h(\boldsymbol{\lambda})$ between the approximate solution y^λ and the exact solution z^λ is defined as follows:*

$$\epsilon_h(\boldsymbol{\lambda}) = \max_{\tau \in \mathcal{D}_h} |z^\lambda(\tau) - y^\lambda(\tau)|. \quad (10)$$

In the rest of the paper, we make two important assumptions on the approximation method. First, we assume that the set \mathcal{T} of time points given by eq. (4), at which the observation data γ are recorded, satisfies $\mathcal{T} \subset \mathcal{D}_h$. This assumption is quite natural as there are a finite number of experimental data, therefore a sufficiently small h can always be chosen accordingly. Our second assumption is that the approximation method is *convergent*, which guarantees that for all $\boldsymbol{\lambda} \in \mathbf{W}$, the global approximation error $\epsilon_h(\boldsymbol{\lambda})$ converges to 0 when h gets smaller. This latter assumption is directly satisfied for usual approximation methods (such as, e.g., Runge-Kutta; see for instance [3]).

3.2 Monte-Carlo method

We now move to our main result, i.e. providing an estimation of the probability that the original ODE system, with a given parameter value $\boldsymbol{\lambda}^*$, agrees with the experimental data with statistical guarantees. For the sake of simplicity, we focus in this section on standard BLTL properties as introduced in section 2.2. We then show in section 3.3 how these results can be extended to reward functions.

Let $\boldsymbol{\lambda}^* \in \mathbf{W}$ be a parameter value. In order to take the internal variability of our system into account, we will consider that $\boldsymbol{\lambda}^*$ can slightly vary. In order to do this, we set a constant $\rho > 0$ and define the open ball

$$B(\boldsymbol{\lambda}^*, \rho) = \{\boldsymbol{\lambda} \in \mathbb{R}^m \mid \|\boldsymbol{\lambda} - \boldsymbol{\lambda}^*\| < \rho\}, \quad (11)$$

where $\|\cdot\|$ is the Euclidian norm defined in section 2.1.

We start by recalling the Monte Carlo procedure for estimation. This procedure aims at taking advantage of the Central Limit Theorem and the Law of Large Numbers. In order to estimate the probability that our system (where $\boldsymbol{\lambda}$ can vary inside of $B(\boldsymbol{\lambda}^*, \rho)$) satisfies the given BLTL property Ψ^* (see eq. (5)), we will generate a set of n samples of values for $\boldsymbol{\lambda}$ inside of $B(\boldsymbol{\lambda}^*, \rho)$, and use these values to provide n solutions to the ODE system. Each solution will be evaluated, yielding a score of 1 if it satisfies Ψ^* and 0 otherwise. Informally, the Central Limit Theorem (Theorem 1) states that the mean value of the samples \hat{p} is a good estimator for the probability p that our system (i.e. the ODE system, where the parameter value is set to $\boldsymbol{\lambda}^*$, with internal variability) satisfies Ψ^* . Moreover, it also provides a confidence interval that solely depends on the number of samples—provided this number is large enough—and the variance of the initial distribution.

Theorem 1 (Central Limit Theorem [19]). *Let X_1, X_2, \dots be a sequence of independent and identically distributed random variables of mean μ and variance σ^2 . Then, the distribution of $\frac{\sum_{i=1}^n X_i - n\mu}{\sigma\sqrt{n}}$ tends to the standard normal distribution as $n \rightarrow \infty$. That is, for any $a \in \mathbb{R}$,*

$$\lim_{n \rightarrow \infty} \mathbb{P} \left(\frac{\sum_{i=1}^n X_i - n\mu}{\sigma\sqrt{n}} \leq a \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^a e^{-x^2/2} dx.$$

Because we cannot evaluate the exact solutions of the ODE system but instead have to rely on approximate solutions, we will define two auxiliary properties φ_1^ε and φ_2^ε (not expressed in BLTL) that take into account the global approximation error defined above, use the Monte Carlo procedure to estimate two probabilities \hat{p}_1^ε and \hat{p}_2^ε using those properties and the approximate solutions, and finally propose an estimation of \hat{p} that relies on \hat{p}_1^ε and \hat{p}_2^ε . We will finally use \hat{p} in order to rate the chosen (central) parameter value λ^* .

Let \mathcal{T} be a set of time points as described earlier. Let γ be the set of experimental data and $\delta > 0$ be a precision (tolerance) w.r.t. γ . Let $\lambda^* \in \mathbf{W}$ be a parameter value, let $\rho > 0$ be a variability setting. Consider the ball $\mathcal{B}_{\lambda^*} = B(\lambda^*, \rho)$ and let \mathbb{P}^{λ^*} be the uniform distribution on this ball.

Given a function $g \in F_{\mathcal{T}}$, we write $\varphi(g) := d(g, \gamma) \leq \delta$ the property that means “the distance between g and γ is less than δ ”. Note that this property can easily be written in BLTL (see eq. (5) above). For convenience, if y^λ is an approximate solution to eq. (1) induced by the parameter $\lambda \in \mathcal{B}_{\lambda^*}$, we will identify $\varphi(\lambda)$ to $\varphi(y^\lambda)$.

Given $\varepsilon > 0$, we introduce the properties:

$$\begin{aligned} \varphi(z^\lambda) &:= d(z^\lambda, \gamma) \leq \delta, \\ \varphi_1^\varepsilon(y^\lambda) &:= d(y^\lambda, \gamma) + \varepsilon \leq \delta, \quad \varphi_2^\varepsilon(y^\lambda) := d(y^\lambda, \gamma) - \varepsilon \leq \delta. \end{aligned}$$

The translation of φ in BLTL is the property of interest Ψ^* defined in eq. (5). Our aim is to provide an estimation \hat{p} for $\mathbb{P}^{\lambda^*}(\Psi^*)$. For convenience, we write \mathbb{P} for \mathbb{P}^{λ^*} in the rest of this section.

In order to do that, we show in Lemma 1 that for a small enough integration step h , we have $\epsilon_h(\lambda) \leq \varepsilon$ for all $\lambda \in \mathcal{B}_{\lambda^*}$, and therefore

$$\varphi_1^\varepsilon(y^\lambda) \Rightarrow \varphi(z^\lambda) \Rightarrow \varphi_2^\varepsilon(y^\lambda). \quad (12)$$

Lemma 1. *Let $(h_i)_{i \in \mathbb{N}} \in \mathbb{R}^+$ be a sequence of integration steps, such that $\lim_{i \rightarrow \infty} h_i = 0$. Then for all $\varepsilon > 0$, there exists $i^* > 0$ such that*

$$\epsilon_{h_i}(\lambda) < \varepsilon, \quad \forall i \geq i^*, \forall \lambda \in \mathcal{B}_{\lambda^*}. \quad (13)$$

In other words, the global error $\epsilon_h(\lambda)$ can be uniformly bounded in the closure $\overline{\mathcal{B}_{\lambda^*}}$ of the open ball \mathcal{B}_{λ^*} . The proof of this lemma is given in appendix A, along with a method to compute h_{i^*} .

Now, we define the probabilities

$$p = \mathbb{P}(\varphi(z_{\lambda^*})), \quad p_1^\varepsilon = \mathbb{P}(\varphi_1^\varepsilon(y_{\lambda^*})), \quad p_2^\varepsilon = \mathbb{P}(\varphi_2^\varepsilon(y_{\lambda^*})). \quad (14)$$

Note that p, p_1, p_2 implicitly depend on δ . However, we omit this dependence in order to lighten our notations. Next, it is straightforward that

$$p_1^\varepsilon \leq p \leq p_2^\varepsilon, \quad \forall \varepsilon > 0. \quad (15)$$

Estimators $\hat{p}_1^\varepsilon, \hat{p}_2^\varepsilon$ of the probabilities p_1^ε and p_2^ε respectively can be determined using the Monte-Carlo procedure, involving a precision α and a risk θ . Our main result, given in Theorem 2 below, establishes a statistical guarantee on the probability p of interest with respect to these estimators $\hat{p}_1^\varepsilon, \hat{p}_2^\varepsilon$.

Theorem 2 (Main theorem).

Let $\lambda^* \in \mathbf{W}$, $\rho > 0$, $\delta > 0$, $\varepsilon > 0$. For any risk $\xi \in (0, 1)$, we define $\theta = 1 - \sqrt{1 - \xi}$. Then, for any precision $\alpha > 0$, the probabilities p_1^ε and p_2^ε defined in eq. (14) satisfy

$$\mathbb{P}(p_1^\varepsilon \in [\hat{p}_1^\varepsilon - \alpha, \hat{p}_1^\varepsilon + \alpha]) \geq 1 - \theta, \quad \mathbb{P}(p_2^\varepsilon \in [\hat{p}_2^\varepsilon - \alpha, \hat{p}_2^\varepsilon + \alpha]) \geq 1 - \theta, \quad (16)$$

where the estimators \hat{p}_1^ε and \hat{p}_2^ε can each be determined after performing a number $N' = \frac{\log(2/\theta)}{2\alpha^2}$ (and hence a total number $N = 2 \times \frac{\log(2/\theta)}{2\alpha^2}$) of simulations of eq. (1) induced by parameter values λ sampled in \mathcal{B}_{λ^*} .

Furthermore, there exist $\varepsilon_0 > 0$ and $h_0 > 0$ sufficiently small such that, for any integration step $h \leq h_0$ and any $\varepsilon < \varepsilon_0$, the following statements hold:

- the probability p defined in eq. (14) satisfies the estimation

$$\mathbb{P}(p \in [\hat{p}_1^\varepsilon - \alpha, \hat{p}_2^\varepsilon + \alpha]) \geq 1 - \xi, \quad (17)$$

- the distance between \hat{p}_1^ε and \hat{p}_2^ε satisfies:

$$\mathbb{P}(|\hat{p}_1^\varepsilon - \hat{p}_2^\varepsilon| \leq 3\alpha) \geq 1 - \xi. \quad (18)$$

We emphasize that estimations (17) and (18) imply a confidence interval of width 5α for p and require a number of samples $N = 2 \times \frac{\log(2/\theta)}{2\alpha^2}$. If the analysis was performed directly on the exact solutions of the ODE, we would have a confidence interval of width 2α and only require $\frac{\log(2/\xi)}{2\alpha^2}$ samples.

The proof of Theorem 2, given in appendix B, is divided in three main steps. First, using the Central Limit Theorem and the Law of Large Numbers, we determine estimators \hat{p}_1^ε and \hat{p}_2^ε of p_1^ε and p_2^ε , respectively. Then, eq. (14) and the independence of simulations lead to the confidence interval of p . Finally, Lemma 1 guarantees that proper values of h and ε can be found, in order to control the distance between \hat{p}_1^ε and \hat{p}_2^ε . It is worth noting that, for some resolution methods (such as Runge-Kutta 4 for example), a value for h can be explicitly determined to guarantee Lemma 1 for a given ε and therefore eq. (17). However, the convergence speed of $|\hat{p}_1^\varepsilon - \hat{p}_2^\varepsilon|$ is not known in general, therefore we can only guarantee the existence of a sufficiently small value for ε to ensure eq. (18) but not compute it.

3.3 Model checking extension through reward functions

As explained in section 2.3, our method can be extended to non-Boolean reward functions. Indeed, these functions may provide not only qualitative results—“does the property hold?”—but also *quantitative* ones—“*how well* does the property hold?”. In our case, this allows to distinguish the good parameters that induce a suitable solution from the best ones that induce the solutions closest to the data.

To use such a real-valued reward function r , some conditions are required. First, it must be assumed that two other reward functions r_1 and r_2 can be found, such that the following estimation holds for any $\lambda \in \mathcal{B}_{\lambda^*}$:

$$r_1(\lambda) \leq r(\lambda) \leq r_2(\lambda). \quad (19)$$

Second, the law of the unconscious statistician must be applicable to these lower and upper reward functions, i.e. the computation of the expected value² must be applicable, so that estimators \hat{r}_1 and \hat{r}_2 of r_1 and r_2 respectively, can be computed.

Moreover, and most importantly, the reward function must be *compatible* with the global error defined in eq. (10). Indeed, since we compute score based on approximated solutions, said computations must take this approximation into account to provide any significance to the resulting score. It is worth noting that these conditions are satisfied by all the reward functions we have considered in this work, such as the total accumulated/maximal/average distance to γ or the number of time points where γ is not respected.

Similarly to eq. (18), the distance between \hat{r}_1 and \hat{r}_2 must be controlled. Depending on the order of the approximation method used to compute approximate solutions to the ODEs, this may be easy to ensure. For instance, in our case the integration method Runge-Kutta 4 ensures that the approximation error—and thus, the global error as defined in eq. (10)—is of order 5: all derivatives of the integration functions converge at most linearly w.r.t. h^5 , where h is the integration step.

4 Case studies

In this section, we apply our method to two case studies [17, 22] taken from the literature to show its potential. After presenting the studies and their results, we will display our results and discuss them. We implemented our technique in C++ to validate the approach. The experiments were realized on a 2.1 GHz Intel Xeon Silver 4216 processor, running g++ version 7.5.0 on Ubuntu 18.04. The code is available at https://gitlab.com/davidjulien/smc_for_ode.git, and the experiments can be reproduced using the right branches, i.e. `compute_aurelia` to run the experiment from section 4.1 and `compute_prey` to run the experiment from section 4.2. We used the Runge-Kutta 4 method to compute approximate solutions, a SMC precision $\alpha = 0.05$ and a risk $\xi = 0.05$.

First, we briefly recall the experiment. After discretizing the value space \mathbf{W} defined in eq. (2) for the parameter λ , we will grade every value in order to

² See eq. (27) in appendix B.

select the best ones w.r.t. the experimental data γ . In order to take the internal variability of the model into account, each chosen parameter value λ^* is associated with the open ball \mathcal{B}_{λ^*} as defined in eq. (11). Once the SMC parameters α and ξ , as well as a small enough value for ε are chosen, we can compute an integration step h , as well as a required number N of samples such that theorem 2 holds. Then, we sample N values $\lambda \in \mathcal{B}_{\lambda^*}$, compute the approximated solutions to the induced ODEs, and compare them with the experimental data γ . For each $\lambda^* \in \mathbf{W}$, we thus estimate the probabilities \hat{p}_1^ε and \hat{p}_2^ε defined in the previous section, and use them to define $\text{grade}(\lambda^*) = \frac{\hat{p}_1^\varepsilon + \hat{p}_2^\varepsilon}{2}$. In order to better discriminate the best parameter values, we also estimate the expected value of the reward function $r : \lambda \mapsto d(z_\lambda, \gamma)$ that measures the distance between the ODE simulations and the experimental data.

4.1 Case study 1: a study on *Aurelia Aurita* population growth [17]

In 2014, Melica et al. [17] published a paper studying the growth of *Aurelia Aurita*, a species of jellyfish that is very common in Adriatic Sea. In this paper, they compared experimental data, resulting from the culture of *Aurelia Aurita* polyps, to simulation models based on the following ODE:

$$x'(t) = ax(t)(1 - x(t)/b) \quad (20)$$

where t is time, x is the population density, a is the maximum rate of population growth, and b is the positive equilibrium. The authors show that the dynamics of a *Aurelia Aurita* polyps population can, indeed, be modeled by the density-dependent, or Verhulst [23], ODE presented above and compute the values for a and b that ensure the best fitting w.r.t. the experimental data. These values are recalled in table 1.

	HD	LD
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 1. Estimation of parameters of the logistic curve fitting the laboratory experimental data [17].

Remark 1. HD and LD represent the studies for *High* and *Low Density*, respectively, which were both ran by the original authors. Here, we focused on the *High Density* case.

In order to illustrate our method, we applied it to the same case study, using eq. (20) as the ODE system. We evaluated parameter values in the ranges $a \in [0, 3]$, $b \in [0, 9]$, and discretized this space with a parameter step of 0.01. We set the internal variability of the parameters $\rho = 0.005$ and performed $N = 874$ simulations for each parameter value on the discretized space, therefore ensuring a statistical precision of $\alpha = 0.05$ and risk of $\xi = 0.05$.

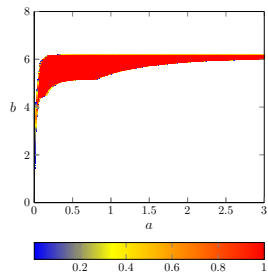


Fig. 3. Heatmap of the score of the parameters.

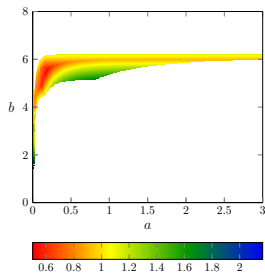


Fig. 4. Heatmap of the distance to γ .

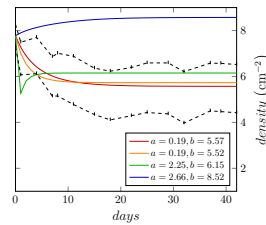


Fig. 5. Solution to the ODE from eq. (20).

In fig. 3, we represent the score of the best parameter values, where the white zones are zones where $\text{grade}(\lambda^*) = 0$. One can see that there is a small gradient in the area where the score is positive, but this is not enough to discriminate between the parameter values in this zone. In order to refine the result, we present in fig. 4 the estimation of the expected value of the reward function $r : \lambda \mapsto d(z_\lambda, \gamma)$. Figure 4 shows a tighter area of values that induce solutions that are very close to the data (down to 0.50 polyps on average), plotted in red, which contains the parameter value estimated by [17]: it comforts us in saying that our method provides tangible results. The best parameter found using our method is the pair $(a, b) = (0.19, 5.57)$. It induces the red curve in fig. 5.

4.2 Case study 2: a prey-predator model for lynx and hares [22]

In 2010, Restrepo and Sánchez [22] published a paper describing a genetic algorithm, which aimed at estimating the best parameters for prey-predator models. The first model, which we will study in the following, is a basic prey-predator interaction model defined by the following ODE system:

$$P' = aP - bPD, \quad D' = -cD + dPD \quad (21)$$

where t is time, P and D are the two time-dependent variables representing the quantity of individuals in each group: $P(t)$ for prey and $D(t)$ for predators; a , b , c , d are positive constants, a and c indicating the birth rate of prey and death rate of predators respectively, and b and d representing the rates of predation and reproduction of predators. Note that even if the model is a standard way to describe, on first approximation, such an interaction between two populations, its simplicity might make it imprecise – which is why other, more complex models are studied in [22]. The best values for parameters (a, b, c, d) w.r.t. experimental data are given in [22]: $(a, b, c, d) = (0.55, 0.027, 0.83, 0.026)$.

Again, we applied our method to this case study, using eq. (21) as the ODE system. We evaluated parameter values in the ranges $a \in [0.48, 0.68]$, $b \in [0.015, 0.04]$, $c \in [0.78, 0.9]$, $d \in [0.01, 0.05]$, and discretized this space with a parameter step of 0.001. We set the internal variability of the parameters

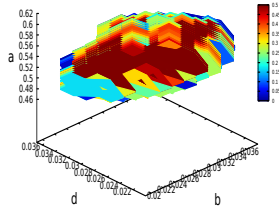


Fig. 6. Heatmap of the score (for $c = 0.89$).

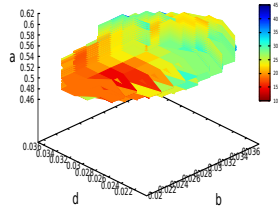


Fig. 7. Heatmap of the distance to γ (for $c = 0.89$).

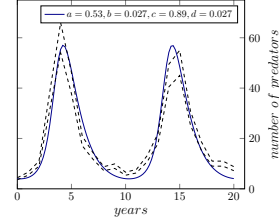


Fig. 8. Best solution to eq. (21).

$\rho = 0.0005$ and performed $N = 874$ simulations for each parameter value on the discretized space, therefore ensuring a statistical precision of $\alpha = 0.05$ and risk of $\xi = 0.05$. Note that the parameter ranges have been tightened according to the paper results, since cyclic models can be very sensitive to parameter values.

We encountered two issues with this study. First, the adequacy of the model with the data was improvable, thus computing a fitting solution was challenging. We had to loosen the property we verify: instead of Ψ^* , which enforces the solution to always stay in the tunnel, we verified a property Ψ^\dagger , which allows the solutions to step out of the tunnel a total of 5 times (for a total of 22 time points) before rejecting them. This explains why the best solution displayed in fig. 8 does not perfectly fit inside of the tunnel. Remark that Ψ^\dagger can easily be expressed in BLTL, and is therefore compatible with our theory. Second, since system eq. (21) involves 4 parameters, displaying the results with heatmaps is more difficult than in case study 1 (section 4.1). Nonetheless, locking a parameter value (here, $c = 0.89$) allows the plot of a 3-dimensional heatmap.

Figure 6 shows a local subset of solutions fitting the data with a certain quality. Notice that the score only goes up to 0.5; because of the internal variability we impose and the sensitivity of the model, very few simulations stayed in the inner tunnel (corresponding to φ_1^ε in section 3), yielding $\hat{p}_1^\varepsilon = 0$ in most cases. That said, fig. 7 displays the distance for the same subset. We notice that some solutions are, at most, at a 10 individuals distance from the data. The subset contains our best candidate $(a, b, c, d) = (0.52, 0.027, 0.89, 0.027)$, whose corresponding curve is displayed in fig. 8. We see that the general shape of the curve is satisfying but does not perfectly fit inside of the tunnel. This may be explained by the fact that cyclic ODE systems like prey-predator models can be very sensitive to the non-linear terms, i.e. bPD and dPD . Again, our goal here was to prove the concept rather than describe a phenomenon with the upmost precision: while this is satisfying as far as we are concerned, a more thorough study of the parameters, along with a better quality of the data (with e.g. several observations for each time point t , allowing for more robust data and observable data tunnels) would help getting results closer to the actual experiments.

5 Conclusion

In this paper, we have proposed a statistical method for synthesizing the best parameter values w.r.t. given experimental data for an ODE system with internal variability, while providing formal statistical guarantees that for the first time (to the best of our knowledge) take into account the approximation error introduced through the numerical resolution of the ODEs. To do that, we discretize the parameter space and define balls around the resulting (finite) set of parameter values to take internal variability into account. We then use the Monte-Carlo technique to estimate the probability that exact solutions of the ODE system are close to the experimental data for each resulting parameter ball, and use the result of this estimation to select the best (central) parameter values. Our main contribution is Theorem 2 which guarantees the precision of our estimation despite the fact that it is performed using numerical resolution techniques that do not give us access to exact solutions of the ODE system. In contrast with other existing works on parameter estimation for ODE systems, like [15], where this problem is left aside, we show that the number of simulations required for a given precision and risk of the statistical estimation is (more than) twice the one needed when working with exact solutions. We also show that an upper bound on the integration step of the chosen integration technique exists (and can be computed for standard integration techniques) in order to make sure that a given statistical precision and risk are respected.

One of the limitations of our work is that, in order to prove our results and perform parameter synthesis in practice, we rely on a setting ε that represents the maximal admissible distance between exact and approximate solutions to the ODE system. Although it is possible, for most integration techniques³, to compute an integration step that will guarantee that a given value for ε is respected, our results only show the existence of a suitable value for ε for any statistical setting, but do not provide any method to compute this value in practice. This is due to our lack of guarantees on the convergence speed of the distance between the two estimators \hat{p}_1^ε and \hat{p}_2^ε that appear in eq. (17) and eq. (18). What we do in practice is that we set small values for ε , perform experiments and then estimate the value of $|\hat{p}_1^\varepsilon - \hat{p}_2^\varepsilon|$. If the resulting value is too large, then we start over the experiment with a smaller value for ε .

Although the only BLTL property that we verify in this paper is the property Ψ^* defined in eq. (5), we believe that our reasoning can be easily extended to other BLTL property following the definition given in section 2.2. This is, in our opinion, a straightforward extension that we will address in the near future.

As said in the introduction, our results are generic and could therefore be combined with any exploration strategy for the parameter space. The global exploration we perform in this paper is obviously costly but yields global information that is precious when analysing a complex system. In the future, we plan on combining a coarse global exploration to identify interesting zones in the parameter space with more efficient and detailed search algorithms (such as the one from [15]) limited to those zones.

³ See remark 2 in appendix A, and [16].

Appendices

In these appendices, we provide the complete proofs of Lemma 1 and Theorem 2.

A Proof of Lemma 1

First, we recall the definition of *stability* of an approximation method.

Definition 2 (Method stability). *We say that the approximation method determined by eq. (7) is stable if there exists a constant $\mathcal{K} > 0$, called stability constant, such that, for any two sequences $(y_k)_{0 \leq k \leq J}$ and $(\tilde{y}_k)_{0 \leq k \leq J}$ defined as $y_{k+1} = y_k + h \Phi(\tau_k, y_k, \boldsymbol{\lambda}, h)$ and $\tilde{y}_{k+1} = \tilde{y}_k + h \bar{\Phi}(\tau_k, \tilde{y}_k, \boldsymbol{\lambda}, h) + \eta_k$ respectively, ($0 \leq k < J$), with $\boldsymbol{\lambda} \in \mathbf{W}$ and $\eta_k \in \mathbb{R}$, we have*

$$\max_{0 \leq k \leq J} |y_k - \tilde{y}_k| \leq \mathcal{K}(|y_0 - \tilde{y}_0| + \sum_{0 \leq k \leq J} |\eta_k|). \quad (22)$$

It is well-known that if Φ is κ -Lipschitz w.r.t. y , i.e. $\forall t \in [0, T], \forall y, \tilde{y} \in \mathbb{R}, \forall \boldsymbol{\lambda} \in \mathbf{W}$ and $\forall h \in \mathbb{R}, |\Phi(t, y, \boldsymbol{\lambda}, h) - \Phi(t, \tilde{y}, \boldsymbol{\lambda}, h)| \leq \kappa |y - \tilde{y}|$, then stability is ensured (see for instance [3] or [4]).

Now, we fix $\boldsymbol{\lambda}^* \in \mathbf{W}$ and $\boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \mathcal{B}_{\boldsymbol{\lambda}^*}$, and we consider the approximate solutions $y^{\boldsymbol{\lambda}_1}, y^{\boldsymbol{\lambda}_2}$ to eq. (1) relative to $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ and starting from z_0 .

$$\begin{cases} y_0^{\boldsymbol{\lambda}_1} &= z_0, \\ y_{k+1}^{\boldsymbol{\lambda}_1} &= y_k^{\boldsymbol{\lambda}_1} + h \Phi(t_k, y_k^{\boldsymbol{\lambda}_1}, \boldsymbol{\lambda}_1, h), \end{cases} \quad \begin{cases} y_0^{\boldsymbol{\lambda}_2} &= z_0, \\ y_{k+1}^{\boldsymbol{\lambda}_2} &= \tilde{y}_k + h \Phi(t_k, y_k^{\boldsymbol{\lambda}_2}, \boldsymbol{\lambda}_2, h). \end{cases}$$

We recall that the exact solutions to eq. (1) relative to $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$ and starting from z_0 are denoted $z^{\boldsymbol{\lambda}_1}$ and $z^{\boldsymbol{\lambda}_2}$ respectively. For $i \in \{1, 2\}$ and $0 \leq k \leq J$, we introduce the consistency error on $y^{\boldsymbol{\lambda}_i}$ at step k :

$$\epsilon_{h,k}(\boldsymbol{\lambda}_i) = |z^{\boldsymbol{\lambda}_i}(\tau_k) - y^{\boldsymbol{\lambda}_i}(\tau_k)|. \quad (23)$$

The consistency errors satisfy $\epsilon_h(\boldsymbol{\lambda}_i) = \max_{0 \leq k \leq J} \epsilon_{h,k}(\boldsymbol{\lambda}_i)$, for $i \in \{1, 2\}$, where $\epsilon_h(\boldsymbol{\lambda}_i)$ is the global approximation error (defined by eq. (10)). The proof of Lemma 1 can be derived from the following theorem.

Theorem 3 (Stability with respect to consistency error). *Assume that the function Φ defined in eq. (7) is κ_1 -Lipschitz w.r.t. $\boldsymbol{\lambda}$ 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*

$$\forall \boldsymbol{\lambda}_1, \boldsymbol{\lambda}_2 \in \mathcal{B}_{\boldsymbol{\lambda}^*}, \max_{0 \leq k \leq J} |\epsilon_{h,k}(\boldsymbol{\lambda}_1) - \epsilon_{h,k}(\boldsymbol{\lambda}_2)| \leq \mathcal{K} \|\boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2\|, \quad (24)$$

where $\|\cdot\|$ is the Euclidean norm defined in section 2.1.

Proof (of Theorem 3). By assumption, Φ is κ_1 -Lipschitz continuous w.r.t. λ :

$$\forall t, y, h \in \mathbb{R}, \forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, |\Phi(t, y, \lambda_1, h) - \Phi(t, y, \lambda_2, h)| \leq \kappa_1 \|\lambda_1 - \lambda_2\|.$$

It follows that

$$\begin{aligned} |y_{k+1}^{\lambda_1} - y_{k+1}^{\lambda_2}| &\leq |y_k^{\lambda_1} - y_k^{\lambda_2}| + h|\Phi(t_k, y_k^{\lambda_1}, \lambda_1, h) - \Phi(t_k, y_k^{\lambda_2}, \lambda_2, h)| \\ &\leq |y_k^{\lambda_1} - y_k^{\lambda_2}| + h|\Phi(t_k, y_k^{\lambda_1}, \lambda_1, h) - \Phi(t_k, y_k^{\lambda_1}, \lambda_2, h)| \\ &\quad + h|\Phi(t_k, y_k^{\lambda_1}, \lambda_2, h) - \Phi(t_k, y_k^{\lambda_2}, \lambda_2, h)| \\ &\leq (1 + h\kappa_2)|y_k^{\lambda_1} - y_k^{\lambda_2}| + h\kappa_1\|\lambda_1 - \lambda_2\|, \end{aligned}$$

for $0 \leq k \leq J$. We write $|y_k^{\lambda_1} - y_k^{\lambda_2}| = \Delta_{y,k}$ and $\|\lambda_1 - \lambda_2\| = \Delta_\lambda$, and we get

$$\Delta_{y,k+1} \leq (1 + h\kappa_2)\Delta_{y,k} + h\kappa_1\Delta_\lambda. \quad (25)$$

Applying the discrete Gronwall lemma (see for instance [7], VIII.2.3), we deduce

$$\max_{0 \leq k \leq J} \Delta_{y,k} \leq e^{\kappa_2 T} (\Delta_{y,0} + \sum_{0 \leq j \leq k-1} h\kappa_1 \Delta_\lambda)$$

which leads to

$$\max_{0 \leq k \leq J} |y_k^{\lambda_1} - y_k^{\lambda_2}| \leq e^{\kappa_2 T} T \kappa_1 \|\lambda_1 - \lambda_2\|,$$

since $y_0^{\lambda_1} = y_0^{\lambda_2} = z_0$ and $hJ = T$.

Furthermore, it is proved in [4] that if Φ is Lipschitz continuous w.r.t. λ , then the exact solution z_λ is also Lipschitz continuous w.r.t. λ that is, there exists $\kappa_3 > 0$ such that

$$\forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, \forall t \in [0, T], |z^{\lambda_1}(t) - z^{\lambda_2}(t)| \leq \kappa_3 \|\lambda_1 - \lambda_2\|. \quad (26)$$

Finally, we have

$$\begin{aligned} |\epsilon_{h,k}(\lambda_1) - \epsilon_{h,k}(\lambda_2)| &\leq |z^{\lambda_1}(\tau_k) - y^{\lambda_1}(\tau_k) - z^{\lambda_2}(\tau_k) - y^{\lambda_2}(\tau_k)| \\ &\leq |z^{\lambda_1}(\tau_k) - z^{\lambda_2}(\tau_k)| + |y^{\lambda_1}(\tau_k) - y^{\lambda_2}(\tau_k)| \\ &\leq \mathcal{K} \|\lambda_1 - \lambda_2\|, \end{aligned}$$

with $\mathcal{K} = \kappa_3 + T\kappa_1 e^{\kappa_2 T}$, which completes the proof of Theorem 3. \square

It remains to show that Theorem 3 implies Lemma 1.

Proof (of Lemma 1). Let $(h_i)_{i \geq 0}$ be a sequence of discretization steps such that $\lim_{i \rightarrow \infty} h_i = 0$. Since the approximation method given by (7) is assumed to be convergent, each function $\epsilon_{h_i}(\cdot)$ defined in eq. (23) is pointwise convergent to 0.

Furthermore, we recall that Φ is Lipschitz continuous w.r.t. $\lambda \in \mathbf{W}$. Hence, Theorem 3 implies that the functions $(\epsilon_{h_i}(\cdot))_{i \geq 0}$ defined in eq. (23) are also Lipschitz continuous, with uniform Lipschitz constant \mathcal{K} :

$$|\epsilon_{h_i}(\lambda_1) - \epsilon_{h_i}(\lambda_2)| \leq \mathcal{K} \|\lambda_1 - \lambda_2\|, \quad \forall \lambda_1, \lambda_2 \in \mathcal{B}_{\lambda^*}, \quad \forall i \in \mathbb{N}.$$

Consequently, the functions $(\epsilon_{h_i}(\cdot))_{i \geq 0}$ are uniformly equicontinuous. Hence, Arzelà–Ascoli Theorem [8] implies that the sequence $(\epsilon_{h_i}(\cdot))_{i \geq 0}$ converges uniformly to 0 on \mathcal{B}_{λ^*} , thus $\forall \varepsilon > 0, \exists i^* \in \mathbb{N}, \forall i \geq i^*, \forall \lambda \in \mathcal{B}_{\lambda^*}, \epsilon_{h_i}(\lambda) < \varepsilon$, and Lemma 1 is proved. \square

Remark 2 (Computation of a sufficiently small integration step). We emphasize that Lemma 1 can be supplemented by an explicit choice of a sufficiently small integration step h , provided the integration method comes with appropriate estimates of their global error. Notably, the accuracy of the Runge-Kutta 4 method, which we use for the numerical treatment of our case studies, has been thoroughly studied (see [16] for instance), and it is known that its inherent error can be bounded in terms of the successive derivatives of the function f involved in eq. (1), up to order 4.

B Proof of Theorem 2

First step. We begin the proof of Theorem 2 by showing how to compute an estimator \hat{p}_1^ε of the probability p_1^ε defined in (14).

Let $(\lambda_i)_{\mathbb{N}}$ be a sequence of values in the ball \mathcal{B}_{λ^*} . We write B_i the random variable corresponding to the test “ $\varphi_1^\varepsilon(\lambda_i)$ holds”: all the B_i are i.i.d. variables and follow a Bernoulli’s law of parameter p_1^ε . We write b_i the evaluation of B_i . We introduce the transfer function $g_1 : \mathcal{B}_{\lambda^*} \rightarrow \{0, 1\}$ corresponding to the test regarding $\varphi_1^\varepsilon(\lambda_i)$, defined by $g_1(\lambda_i) = 1$ if $\varphi_1^\varepsilon(\lambda_i)$ holds, 0 otherwise. Next, we consider

$$G = \mathbb{E}(g_1(X)) = \int_{\mathcal{B}_{\lambda^*}} g_1(x) f_X(x) dx, \quad (27)$$

where f_X is defined by a uniform distribution, that is, $f_X(x) = \frac{1}{|\mathcal{B}_{\lambda^*}|}$, $x \in \mathcal{B}_{\lambda^*}$. We produce a sample (x_1, x_2, \dots, x_N) of the variable X in \mathcal{B}_{λ^*} , and use it to compute the Monte-Carlo estimator G . By virtue of the Law of Large Numbers, the sample mean satisfies: $\bar{g}_N = \frac{1}{N} \sum_{i=1}^N g_1 x_i$. The Central Limit Theorem states that the variable $Z = \frac{\bar{g}_N - G}{\sigma_{\bar{g}_N}}$ approximately follows a Standard Normal Distribution $\mathcal{N}(0, 1)$; hence, for a risk θ , we can bound the error $|\alpha_N|$ of swapping G with \bar{g}_N by building confidence intervals:

$$\mathbb{P}\left(|\alpha_N| \leq \chi_{1-\frac{\theta}{2}} \frac{\sigma_{g_1}}{\sqrt{N}}\right) = 1 - \theta, \quad (28)$$

where $\chi_{1-\frac{\theta}{2}}$ is the quantile of the Standard Normal Distribution $\mathcal{N}(0, 1)$ and σ_{g_1} is the variance of g_1 .

Since we are interested in finding p_1^ε with a certain confidence, we can perform this process after setting the desired target error α and risk θ , knowing how many simulations must be ran using Hoeffding’s inequality [12]:

$$\theta = \mathbb{P}(\bar{g}_N \notin [p_1^\varepsilon - \alpha, p_1^\varepsilon + \alpha]) \leq 2 \exp(-2\alpha^2 N),$$

or equivalently $N \geq \frac{\log(2/\theta)}{2\alpha^2}$. Here, it is worth emphasizing that N can be chosen independently of ε .

Further, the variance of \bar{g}_N can be expressed with the variance of $g_1(X)$:

$$\sigma_{g_1}^2 = \mathbb{E}([g_1(X) - \mathbb{E}(g_1(X))]^2) = \int_{\mathcal{B}_{\lambda^*}} (g_1(x))^2 f_X(x) dx - G^2.$$

We consider i.i.d. samples, hence $\sigma_{g_1}^2$ can be estimated with the variance $S_{g_1}^2$:

$$\sigma_{g_1}^2 \simeq S_{g_1}^2 = \frac{1}{N} \sum_{i=1}^N (g_1(\lambda_i)^2 - \bar{g}_N^2).$$

It follows that σ_{g_1} can be estimated with its empirical counterpart $\hat{\sigma}_{g_1} = \sqrt{S_{g_1}^2}$, which shows that the error displays a $1/\sqrt{N}$ convergence.

Finally, after estimating σ_{g_1} , we can find \hat{p}_1^ε using the variance of Bernoulli's law $\hat{\sigma}_{g_1}^2 = \hat{p}_1^\varepsilon \times (1 - \hat{p}_1^\varepsilon)$. We conclude that the probability that $\varphi_1^\varepsilon(\lambda)$ holds is estimated by $\hat{p}_1^\varepsilon = \frac{1}{2} \left(1 \pm \sqrt{1 - 4\hat{\sigma}_{g_1}^2} \right)$, with an error α and a risk θ , provided we perform $N \geq \frac{\log(2/\theta)}{2\alpha^2}$ simulations. It follows that

$$\mathbb{P}(p_1^\varepsilon \in [\hat{p}_1^\varepsilon - \alpha, \hat{p}_1^\varepsilon + \alpha]) \geq 1 - \theta. \quad (29)$$

Similarly, we determine an estimator \hat{p}_2^ε of p_2^ε by running $N \geq \frac{\log(2/\theta)}{2\alpha^2}$ additional simulations, and obtain a confidence interval satisfying

$$\mathbb{P}(p_2^\varepsilon \in [\hat{p}_2^\varepsilon - \alpha, \hat{p}_2^\varepsilon + \alpha]) \geq 1 - \theta. \quad (30)$$

Second step. Now, let us show how a confidence interval for the probability p can be derived from the confidence intervals given in (29), (30), involving the estimators \hat{p}_1^ε and \hat{p}_2^ε respectively. The independence of the samples used to determine the estimators \hat{p}_1^ε , \hat{p}_2^ε guarantees that

$$\mathbb{P}(p \in [\hat{p}_1^\varepsilon - \alpha, \hat{p}_2^\varepsilon + \alpha]) = \mathbb{P}(\{p \geq \hat{p}_1^\varepsilon - \alpha\}) \times \mathbb{P}(\{p \leq \hat{p}_2^\varepsilon + \alpha\}).$$

By virtue of (29), we have $\mathbb{P}(p_1^\varepsilon \geq \hat{p}_1^\varepsilon - \alpha) \geq 1 - \theta$. Next, the estimate (15) implies $\mathbb{P}(p \geq \hat{p}_1^\varepsilon - \alpha) \geq \mathbb{P}(p_1^\varepsilon \geq \hat{p}_1^\varepsilon - \alpha) \geq 1 - \theta$. Similarly, we have $\mathbb{P}(p \leq \hat{p}_2^\varepsilon + \alpha) \geq 1 - \theta$, and finally $\mathbb{P}(p \in [\hat{p}_1^\varepsilon - \alpha, \hat{p}_2^\varepsilon + \alpha]) \geq (1 - \theta)^2 = 1 - \xi$, since $\theta = 1 - \sqrt{1 - \xi}$.

Third step. Finally, let us prove how Lemma 1 guarantees that proper values of h and ε can be found, in order to control the distance between \hat{p}_1 and \hat{p}_2 .

Indeed, the continuity of the probability measure \mathbb{P} ensures that there exists $\varepsilon_0 > 0$ such that $|p_1^\varepsilon - p_2^\varepsilon| \leq \alpha$, for $\varepsilon < \varepsilon_0$. Next, we write

$$|\hat{p}_1^\varepsilon - \hat{p}_2^\varepsilon| \leq |\hat{p}_1^\varepsilon - p_1^\varepsilon| + |\hat{p}_2^\varepsilon - p_2^\varepsilon| + |p_1^\varepsilon - p_2^\varepsilon|,$$

hence we have, for $\varepsilon < \varepsilon_0$:

$$\begin{aligned} \mathbb{P}(|\hat{p}_1^\varepsilon - \hat{p}_2^\varepsilon| \leq 3\alpha) &\geq \mathbb{P}(|\hat{p}_1^\varepsilon - p_1^\varepsilon| \leq \alpha) \times \mathbb{P}(|\hat{p}_2^\varepsilon - p_2^\varepsilon| \leq \alpha) \times \mathbb{P}(|p_1^\varepsilon - p_2^\varepsilon| \leq \alpha) \\ &\geq (1 - \theta)^2 \times 1 = 1 - \xi. \end{aligned}$$

In parallel, Lemma 1 guarantees that for h sufficiently small, the global stability error can be uniformly bounded on \mathcal{B}_{λ^*} by ε_0 . The proof is complete. \square

References

1. Baier, C., de Alfaro, L., Forejt, V., Kwiatkowska, M.: Model checking probabilistic systems. In: Handbook of Model Checking, pp. 963–999. Springer (2018). https://doi.org/10.1007/978-3-319-10575-8_28
2. Bao, R., Attiogbe, C., Delahaye, B., Fournier, P., Lime, D.: Parametric statistical model checking of UAV flight plan. In: International Conference on Formal Techniques for Distributed Objects, Components, and Systems. pp. 57–74. Springer (2019). https://doi.org/10.1007/978-3-030-21759-4_4
3. Butcher, J.C.: Numerical methods for ordinary differential equations. John Wiley & Sons (2016). <https://doi.org/10.1002/9781119121534>, 3rd Edition
4. Crouzeix, M., Mignot, A.L.: Analyse numérique des équations différentielles, vol. 1. Masson (1984)
5. Daws, C.: symbolic and parametric model checking of discrete-time Markov chains. In: International Colloquium on Theoretical Aspects of Computing. pp. 280–294. Springer (2004). https://doi.org/10.1007/978-3-540-31862-0_21
6. Dehnert, C., Junges, S., Jansen, N., Corzilius, F., Volk, M., Bruintjes, H., Katoen, J.P., Ábrahám, E.: Prophesy: A probabilistic parameter synthesis tool. In: International Conference on Computer Aided Verification. pp. 214–231. Springer (2015). https://doi.org/10.1007/978-3-319-21690-4_13
7. Demailly, J.: Analyse numérique et équations différentielles. Grenoble Sciences, EDP Sciences (2012)
8. Dunford, N., Schwartz, J.: Linear operators. 1. General theory. A Wiley Interscience Publication, Interscience Publishers (1967)
9. Gainer, P., Hahn, E.M., Schewe, S.: Accelerated model checking of parametric Markov chains. In: International Symposium on Automated Technology for Verification and Analysis. pp. 300–316. Springer (2018). https://doi.org/10.1007/978-3-030-01090-4_18
10. Gyori, B.M., Liu, B., Paul, S., Ramanathan, R., Thiagarajan, P.: Approximate probabilistic verification of hybrid systems. In: International Workshop on Hybrid Systems Biology. pp. 96–116. Springer (2015). https://doi.org/10.1007/978-3-319-26916-0_6
11. Han, T., Katoen, J.P., Mereacre, A.: Approximate parameter synthesis for probabilistic time-bounded reachability. In: 2008 Real-Time Systems Symposium. pp. 173–182. IEEE (2008). <https://doi.org/10.1109/RTSS.2008.19>
12. Hoeffding, W.: Probability inequalities for sums of bounded random variables. Journal of the American Statistical Association **58**(301), 13–30 (1963). <https://doi.org/10.1080/01621459.1963.10500830>
13. Katoen, J.P.: The probabilistic model checking landscape. In: Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science. pp. 31–45 (2016). <https://doi.org/10.1145/2933575.2934574>
14. Legay, A., Delahaye, B., Bensalem, S.: Statistical model checking: An overview. In: International Conference on Runtime Verification. pp. 122–135. Springer (2010). https://doi.org/10.1007/978-3-642-16612-9_11
15. Liu, B., Gyori, B.M., Thiagarajan, P.S.: Statistical model checking-based analysis of biological networks, pp. 63–92. Springer International Publishing (2019). <https://doi.org/10.48550/arXiv.1812.01091>
16. Lotkin, M.: On the accuracy of Runge-Kutta’s method. Mathematical Tables and Other Aids to Computation **5**(35), 128–133 (1951). <https://doi.org/10.1090/S0025-5718-1951-0043566-3>

17. Melica, V., Invernizzi, S., Caristi, G.: Logistic density-dependent growth of an *Aurelia aurita* polyps population. *Ecological Modelling* **291**, 1–5 (2014). <https://doi.org/10.1016/j.ecolmodel.2014.07.009>
18. Perko, L.: *Differential equations and dynamical systems*. Texts in Applied Mathematics, Springer New York (2013). <https://doi.org/10.1007/978-1-4613-0003-8>
19. Petrov, V.V.: *Sums of independent random variables*. De Gruyter (2022). <https://doi.org/10.1515/9783112573006>
20. Ramondenc, S., Eveillard, D., Guidi, L., Lombard, F., Delahaye, B.: Probabilistic modeling to estimate jellyfish ecophysiological properties and size distributions. *Scientific Reports* **10**(1), 1–13 (2020). <https://doi.org/10.1038/s41598-020-62357-5>
21. Ramsay, J.O., Hooker, G., Campbell, D., Cao, J.: Parameter estimation for differential equations: A generalized smoothing approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **69**(5), 741–796 (2007). <https://doi.org/10.1111/j.1467-9868.2007.00610.x>
22. Restrepo, J.G., Sánchez, C.M.V.: Parameter estimation of a predator-prey model using a genetic algorithm. In: 2010 IEEE ANDESCON. pp. 1–4 (Sep 2010). <https://doi.org/10.1109/ANDESCON.2010.5633365>
23. Vandermeer, J.H., Goldberg, D.E.: *Population ecology*. In: *Population Ecology*. Princeton University Press (2013)
24. Varah, J.M.: A spline least squares method for numerical parameter estimation in differential equations. *SIAM Journal on Scientific and Statistical Computing* **3**(1), 28–46 (1982). <https://doi.org/10.1137/0903003>