Statistical model checking for ODE models: challenges and applications to Environment modeling

G. Ardourel, G. Cantin, **Benoît Delahaye**, D. Julien, F. Khochman . . .
16/09/2024 Day on Probabilities

Outline

- Context: Natural systems, ODEs, ...

Outline

- Context: Natural systems, ODEs, ...
- SMC for ODE models

Outline

- Context: Natural systems, ODEs, ...
- SMC for ODE models
- Parameterization and stability analysis

Outline

- Context: Natural systems, ODEs, . . .
- SMC for ODE models
- Parameterization and stability analysis
- Abstraction of ODE systems

Outline

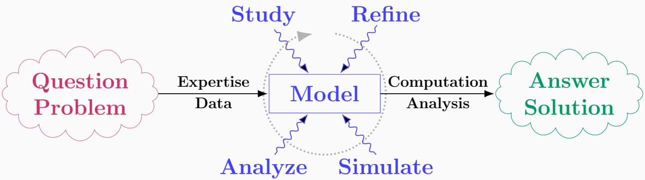
- Context: Natural systems, ODEs, ...
- SMC for ODE models
- Parameterization and stability analysis
- Abstraction of ODE systems
- Applications

Outline

- Context: Natural systems, ODEs, . . .
- SMC for ODE models
- Parameterization and stability analysis
- Abstraction of ODE systems
- Applications
- Perspectives and future work

Context

Science of Models

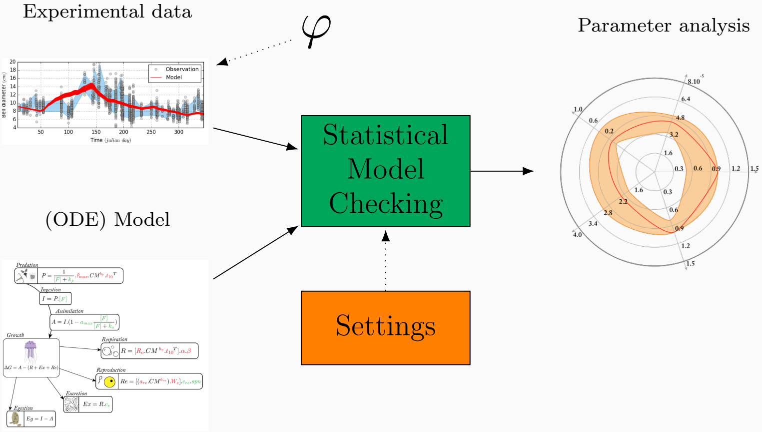


Motivations

Provide mathematical **guarantees** and **tools** for building and analyzing models

- usable by any scientist;
- answering real-life questions.

Example: Calibration of a jellyfish model[1]



[1] [Ramondec et al. 2020] Probabilistic modeling to estimate jellyfish ecophysiological properties and size distribution. *Scientific Reports*

Formal verification of ODE models

System

Formal verification of ODE models

System

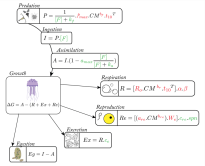
φ

Property

Formal verification of ODE models

System

(ODE) Model



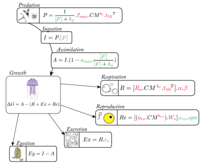
φ

Property

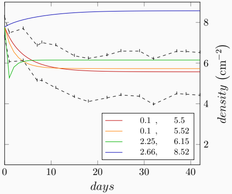
Formal verification of ODE models

System

(ODE) Model



Set of traces



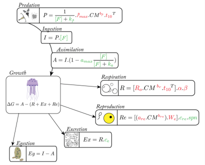
φ

Property

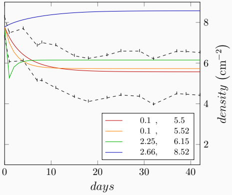
Formal verification of ODE models

System

(ODE) Model



Set of traces



Some models may be verified directly (automata, graphs...).

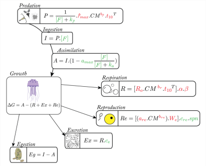
φ

Property

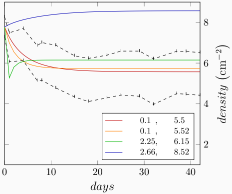
Formal verification of ODE models

System

(ODE) Model



Set of traces



Sometimes we can only analyze traces of the model.

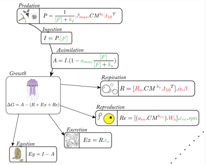
φ

Property

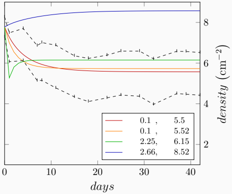
Formal verification of ODE models

System

(ODE) Model



Set of traces

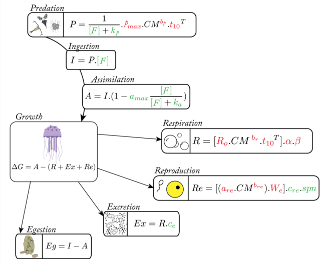


If Model ∼ Traces
⇒ Checking φ on the traces is equivalent to checking it on the model.

φ

Property

Why Probabilistic models?



Uncertainty and variability

- Several experiments

  $\Rightarrow$ data uncertainty.
- Family of systems

  $\Rightarrow$ parameters variability.

Statistical Model Checking for ODE models

SMC: answering “how good is a model?”

The Monte-Carlo procedure:

1. Randomly generate N samples $(\sigma_1, \ldots, \sigma_N)$ from the model M.
2. Check whether the sample i satisfies the property φ.
   $X_i = 1 \Leftrightarrow \sigma_i \models \varphi$
3. Compute the estimator $\hat{p} = \frac{\sum X_i}{N}$.

SMC: answering "how good is a model?"

The Monte-Carlo procedure:

1. Randomly generate N samples $(\sigma_1, \ldots, \sigma_N)$ from the model M.
2. Check whether the sample i satisfies the property φ. $X_i = 1 \Leftrightarrow \sigma_i \models \varphi$
3. Compute the estimator $\hat{p} = \frac{\sum X_i}{N}$.

Central Limit Theorem

- $\hat{p} \sim \mathbb{E}(X)$ $\Rightarrow \hat{p}$ is a good estimation of $p = \mathbb{P}(M \models \varphi)$.
- **precision α, risk θ.**

SMC: answering "how good is a model?"

The Monte-Carlo procedure:

1. Randomly generate N samples $(\sigma_1, \dots, \sigma_N)$ from the model M.
2. Check whether the sample i satisfies the property φ.
   $X_i = 1 \Leftrightarrow \sigma_i \models \varphi$
3. Compute the estimator $\hat{p} = \frac{\sum X_i}{N}$.

Central Limit Theorem

- $\hat{p} \sim \mathbb{E}(X)$
  $\Rightarrow \hat{p}$ is a good estimation of $p = \mathbb{P}(M \models \varphi)$.
- **precision α, risk θ.**

$$\Rightarrow \mathbb{P}(p \in [\hat{p} - \alpha, \hat{p} + \alpha]) \geq 1 - \theta$$

SMC: answering "how good is a model?"

The Monte-Carlo procedure:

1. Randomly generate N samples $(\sigma_1, \ldots, \sigma_N)$ from the model M.
2. Check whether the sample i satisfies the property φ.
   $X_i = 1 \Leftrightarrow \sigma_i \models \varphi$
3. Compute the estimator $\hat{p} = \frac{\sum X_i}{N}$.

Central Limit Theorem

- $\hat{p} \sim \mathbb{E}(X)$
  $\Rightarrow \hat{p}$ is a good estimation of $p = \mathbb{P}(M \models \varphi)$.
- **precision α, risk θ.**

$$\Rightarrow \mathbb{P}(p \in [\hat{p} - \alpha, \hat{p} + \alpha]) \geq 1 - \theta$$

Application to ODE models

[Liu et al. 2019] Statistical Model Checking-Based Analysis of Biological Networks.

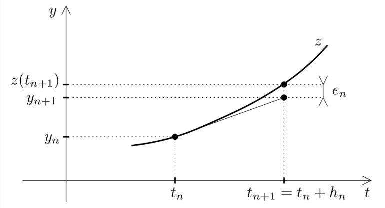
Automated Reasoning for Systems Biology and Medecine

Parameter analysis of ODE models with variability.

Problem: Approximations

Model $\not\approx$ Traces

Approximation error stacks up at each step: $\varepsilon = \max_n e_n$
$\Rightarrow$ SMC estimation does not apply to the original ODE model.

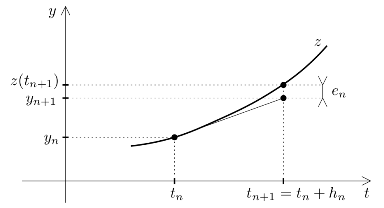


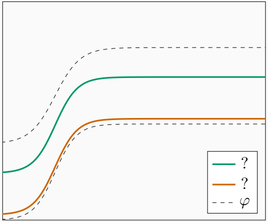
Problem: Approximations

Model $\not\approx$ Traces

Approximation error stacks up at each step: $\varepsilon = \max_n e_n$

$\Rightarrow$ SMC estimation does not apply to the original ODE model.



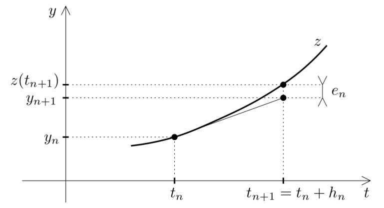


Problem: Approximations

Model ≁ Traces

Approximation error stacks up at each step: $\varepsilon = \max_n e_n$

$\Rightarrow$ SMC estimation does not apply to the original ODE model.



Solution: Safety margins

Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

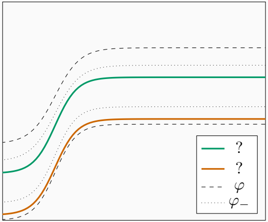
Solution: Safety margins

Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$\sigma \models \varphi_- \Rightarrow M \models \varphi$$



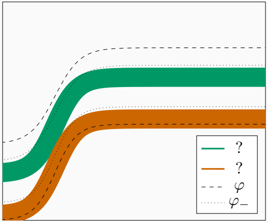
Solution: Safety margins

Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$\sigma \models \varphi_- \Rightarrow M \models \varphi$$



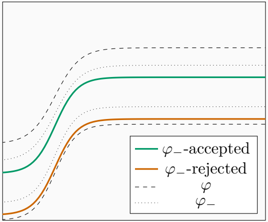
Solution: Safety margins

Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$\sigma \models \varphi_- \Rightarrow M \models \varphi \qquad \Rightarrow \qquad p_- \leq p$$



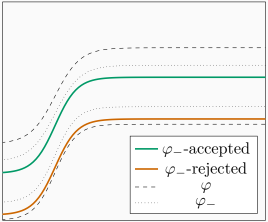
Solution: Safety margins

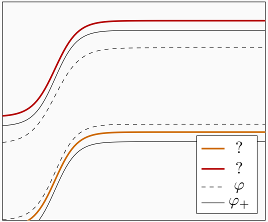
Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$M \models \varphi \Rightarrow \sigma \models \varphi_+$$





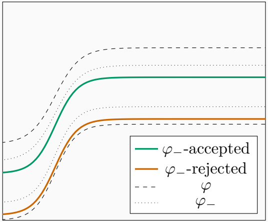
Solution: Safety margins

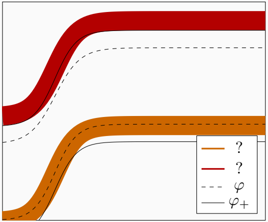
Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$M \models \varphi \Rightarrow \sigma \models \varphi_+$$





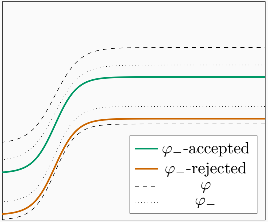
Solution: Safety margins

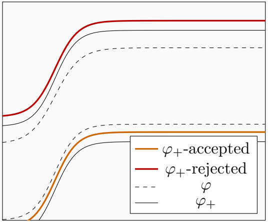
Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$M \models \varphi \Rightarrow \sigma \models \varphi_+ \quad \Rightarrow \qquad p \leq p_+$$





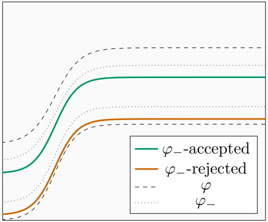
Solution: Safety margins

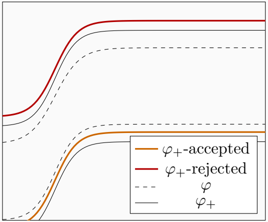
Proposition

If we can bound the approximation error ε on the value space, then we can define new properties $\varphi_- =$ "$\varphi - \varepsilon$" and $\varphi_+ =$ "$\varphi + \varepsilon$" to check.

$\Rightarrow$ Given a trace σ generated from the model M,

$$\sigma \models \varphi_- \Rightarrow M \models \varphi \Rightarrow \sigma \models \varphi_+ \quad \Rightarrow \quad p_- \leq p \leq p_+$$





Guarantees: global risk ξ and precision α

Usual SMC

- $n = \frac{\log(2/\xi)}{2\alpha^2}$ simulations.

$\Rightarrow \mathbb{P}(p \in [\hat{p} - \alpha, \hat{p} + \alpha]) \geq 1 - \xi$

In our case (for each property)

- SMC risk $\theta = 1 - \sqrt{1 - \xi} < \xi$
- $n' = \frac{\log(2/\theta)}{2\alpha^2} > n$ simulations

$$\Rightarrow \mathbb{P}(p_- \in [\hat{p}_- - \alpha, \hat{p}_- + \alpha]) \geq 1 - \theta, \quad \mathbb{P}(p_+ \in [\hat{p}_+ - \alpha, \hat{p}_+ + \alpha]) \geq 1 - \theta$$

Guarantees: global risk ξ and precision α

Usual SMC

- $n = \frac{\log(2/\xi)}{2\alpha^2}$ simulations.

$\Rightarrow \mathbb{P}(p \in [\hat{p} - \alpha, \hat{p} + \alpha]) \geq 1 - \xi$

In our case (for each property)

- SMC risk $\theta = 1 - \sqrt{1 - \xi} < \xi$
- $n' = \frac{\log(2/\theta)}{2\alpha^2} > n$ simulations

$$\Rightarrow \mathbb{P}(p_- \in [\hat{p}_- - \alpha, \hat{p}_- + \alpha]) \geq 1 - \theta, \quad \mathbb{P}(p_+ \in [\hat{p}_+ - \alpha, \hat{p}_+ + \alpha]) \geq 1 - \theta$$

Main Theorem

After performing $N = 2 \times n'$ simulations, the following statements hold:

- $\mathbb{P}(p \in [\hat{p}_- - \alpha, \hat{p}_+ + \alpha]) \geq 1 - \xi$;
- $\mathbb{P}(|\hat{p}_- - \hat{p}_+| \leq 3\alpha) \geq 1 - \xi$.

Bonus: extension to reward functions.

Parameterization and stability analysis

Parameterization

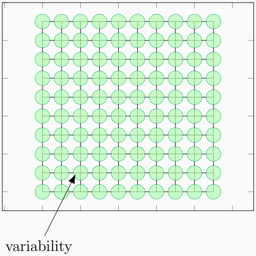
- Parameterization: Find parameter values λ for a generic model.

- Goal: Find good values for λ w.r.t. score $(\mathbb{E}(r))$ of φ-satisfaction.

- Any algorithm:
  - Local search: low execution time / superficial search;
  - **Global** search: more informative / higher execution time;
  - ...

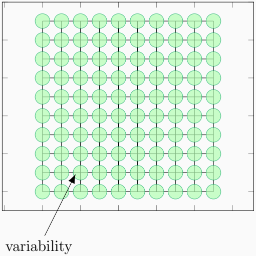
What we do

1. Compute the grid of parameters.
2. Compute the score of each value.
3. Select the best value.



What we do

1. Compute the grid of parameters.
2. Compute the score of each value.
3. Select the best value.

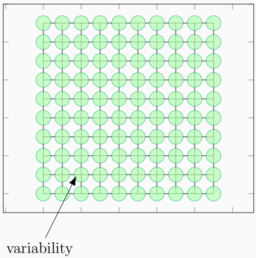


Reminder

It only works if we can bound the error ε!

What we do

1. Compute the grid of parameters.
2. Compute the score of each value.
3. Select the best value.



Reminder

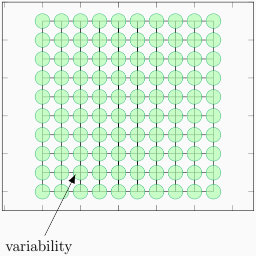
It only works if we can bound the error ε!

Proposition

When solving an ODE, the error ε is bounded by a function of the integration step h.

What we do

1. Compute the grid of parameters.
2. Compute the score of each value.
3. Select the best value.



Reminder

It only works if we can bound the error ε!

Lemma 1

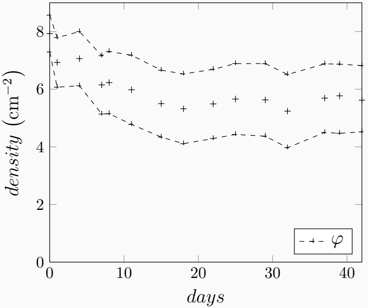
For any arbitrary $\varepsilon > 0$, there exists an integration step h such that

$$0 < \varepsilon_h < \varepsilon, \quad \forall \lambda.$$

POC: Aurelia Aurita[2]

Jellyfish species from the Adriatic Sea.

- $x'(t) = a \cdot x(t) \cdot \left(1 - \frac{x(t)}{b}\right)$
- $\varphi =$ data $\pm$ standard error

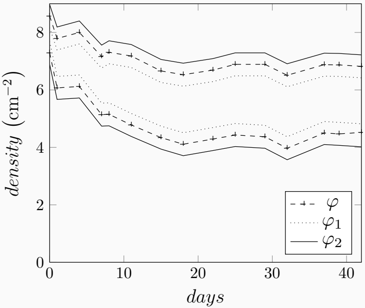


[2][Melica et al. 2014] Logistic density-dependent growth of an aurelia aurita polyps population. *Ecological Modeling*.

POC: Aurelia Aurita[2]

Jellyfish species from the Adriatic Sea.

- $x'(t) = a \cdot x(t) \cdot \left(1 - \frac{x(t)}{b}\right)$
- $\varphi =$ data $\pm$ standard error
- $\varphi_- =$ "$\varphi - \varepsilon$"
- $\varphi_+ =$ "$\varphi + \varepsilon$"



[2] [Melica et al. 2014] Logistic density-dependent growth of an aurelia aurita polyps population. *Ecological Modeling*.

Parameter analysis

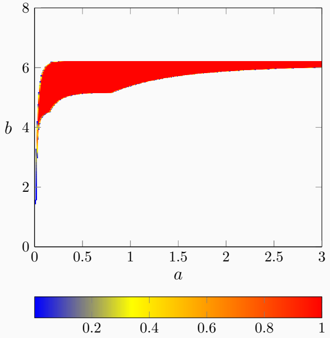


Fig. 1: Probability of staying in the tunnel.

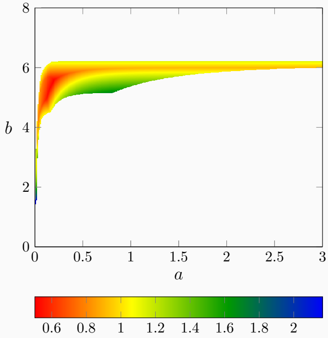


Fig. 2: Expected distance to data.

Stability

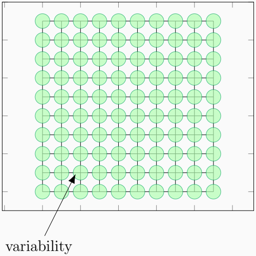
- Stability analysis: Find whether the trajectories of a generic model deviate from its initial condition x_0.

- Goal: Find good values for x_0 w.r.t. score $(\mathbb{E}(r))$ of stability.

- Any algorithm:
  - Local search: low execution time / superficial search;
  - **Global** search: more informative / higher execution time;
  - ...

What we do

1. Grid of **initial conditions**.
2. Compute the score of each value.

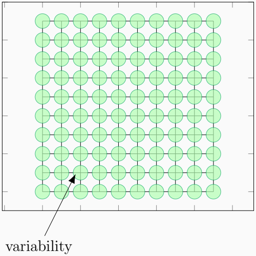


What we do

1. Grid of **initial conditions**.
2. Compute the score of each value.

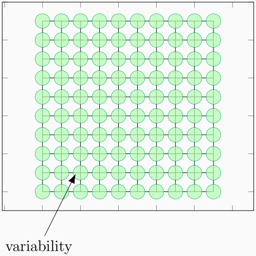
Reminder (again)

It only works if we can bound the error ε!



What we do

1. Grid of **initial conditions**.
2. Compute the score of each value.



Reminder (again)

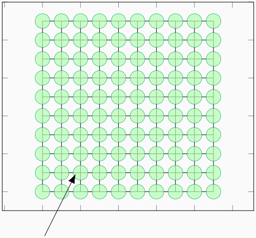
It only works if we can bound the error ε!

Proposition

When solving an ODE, the error ε is bounded by a function of the integration step h.

What we do

1. Grid of **initial conditions**.
2. Compute the score of each value.



Reminder (again)

It only works if we can bound the error ε!

Lemma

For any arbitrary $\varepsilon > 0$, there exists an integration step h such that

$$0 < \varepsilon_h < \varepsilon, \quad \forall \lambda.$$

POC: Duffing's oscillator[3]

Non-linear second order equation used to model damped and driven oscillators.

- $\begin{cases} \frac{\mathrm{d}x_1(t)}{\mathrm{d}t} = x_2(t), \\ \frac{\mathrm{d}x_2(t)}{\mathrm{d}t} = -bx_1(t) - ax_2(t) - dx_1(t)^3 cx_1(t)^2 x_2(t). \end{cases}$
- $\varphi =$ "Stability around x_e."

Stability

Lyapunov: For every $\delta > 0$, there exists $\rho > 0$ such that $\| x(0) - x_e \| < \rho \Rightarrow \| x(t) - x_e \| < \delta$, for all $t > 0$.

[3][Holmes et al. 1980] Phase portraits and bifurcation of the non-linear oscillator $\ddot{x} + \alpha\dot{x} + \gamma x^2 \dot{x} + \beta x + \delta x^3 = 0$. *International Journal of Non-Linear Mechanics.*

POC: Duffing's oscillator[3]

Non-linear second order equation used to model damped and driven oscillators.

- $\begin{cases} \frac{\mathrm{d}x_1(t)}{\mathrm{d}t} = x_2(t), \\ \frac{\mathrm{d}x_2(t)}{\mathrm{d}t} = -bx_1(t) - ax_2(t) - dx_1(t)^3 cx_1(t)^2 x_2(t). \end{cases}$
- $\varphi =$ "**(ρ, δ)-stability around x_e.**"

Relaxed stability

(ρ, δ)-stability: $\| x(0) - x_e \| < \rho \Rightarrow \| x(t) - x_e \| < \delta$, for all $t \in [T_1, T_2]$.

[3] [Holmes et al. 1980] Phase portraits and bifurcation of the non-linear oscillator $\ddot{x} + \alpha\dot{x} + \gamma x^2 \dot{x} + \beta x + \delta x^3 = 0$. *International Journal of Non-Linear Mechanics.*

POC: Duffing's oscillator[3]

Non-linear second order equation used to model damped and driven oscillators.

- $\begin{cases} \frac{\mathrm{d}x_1(t)}{\mathrm{d}t} = x_2(t), \\ \frac{\mathrm{d}x_2(t)}{\mathrm{d}t} = -bx_1(t) - ax_2(t) - dx_1(t)^3 cx_1(t)^2 x_2(t). \end{cases}$
- $\varphi =$ "**(ρ, δ)-stability around x_e.**"
- $\varphi_- =$ "$\varphi - \varepsilon$"
- $\varphi_+ =$ "$\varphi + \varepsilon$"

Relaxed stability

(ρ, δ)-stability: $\| x(0) - x_e \| < \rho \Rightarrow \| x(t) - x_e \| < \delta$, for all $t \in [T_1, T_2]$.

[3] [Holmes et al. 1980] Phase portraits and bifurcation of the non-linear oscillator $\ddot{x} + \alpha\dot{x} + \gamma x^2 \dot{x} + \beta x + \delta x^3 = 0$. *International Journal of Non-Linear Mechanics.*

Stability analysis

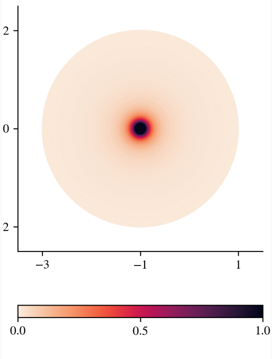


Fig. 3: Stability of $x_e = (-1, 0)$, $\delta = 0.5$.

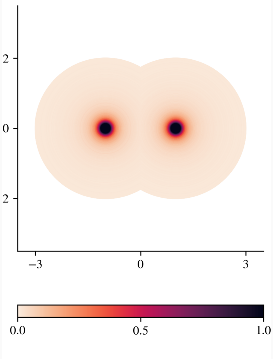


Fig. 4: Stability of $x_{e,1} = (-1, 0)$ and $x_{e,2} = (1, 0)$.

Basins of attraction

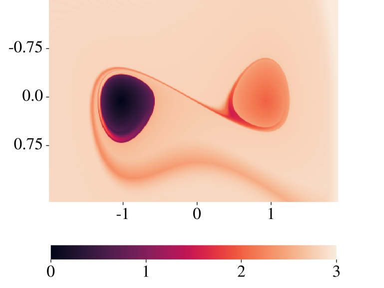


Fig. 5: Basin of attraction of $x_{e,1}$.

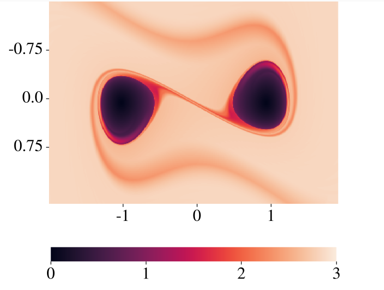


Fig. 6: Basin of attraction of $x_{e,1}$ and $x_{e,2}$.

Abstraction of ODE systems

Context

Formal verification for hybrid systems

- Existing formalisms not adapted for real life systems
- Need for discrete control of ODE models
- Need for formal verification of such models

Context

Formal verification for hybrid systems

- Existing formalisms not adapted for real life systems
- Need for discrete control of ODE models
- Need for formal verification of such models

Context

Formal verification for hybrid systems

- Existing formalisms not adapted for real life systems
- Need for discrete control of ODE models
- Need for formal verification of such models

Objective

New discrete abstraction for ODE models, that can be combined with discrete control

▶ The hybrid model $(\mathcal{H})$ determines a piecewise deterministic dynamical system, with random jumps:

$$(\mathcal{H})\begin{cases}\dfrac{dU}{dt}=f(U,\lambda_k), k\tau\leq t\leq (k+1)\tau, k\geq 0, \lambda_k\in\Lambda, U\in\Phi,\\ U\big((k+1)\tau^-\big)\mapsto U\big((k+1)\tau\big),\\ \lambda_k\mapsto\lambda_{k+1},\\ (U_0,\lambda_0)\in\Phi\times\Lambda.\end{cases}$$

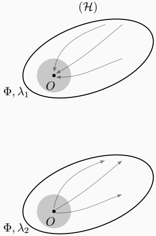


Fig. 7: Trajectories of the hybrid model.

Trajectories of a hybrid model

▶ The hybrid model $(\mathcal{H})$ determines a piecewise deterministic dynamical system, with random jumps:

$$(\mathcal{H}) \begin{cases} \dfrac{dU}{dt} = f(U, \lambda_k), k\tau \leq t \leq (k+1)\tau, k \geq 0, \lambda_k \in \Lambda, U \in \Phi, \\ U\big((k+1)\tau^-\big) \mapsto U\big((k+1)\tau\big), \\ \lambda_k \mapsto \lambda_{k+1}, \\ (U_0, \lambda_0) \in \Phi \times \Lambda. \end{cases}$$

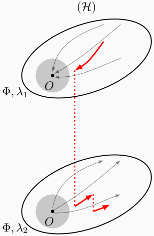


Fig. 7: Trajectories of the hybrid model.

Verification of dynamical properties

▶ Our aim is to verify, using (statistical) Model Checking, several dynamical properties of the hybrid model $(\mathcal{H})$:

- invariance of a neighborhood of the persistence equilibrium,
- reachability of a neighborhood of the persistence equilibrium,
- oscillations between persistence and extinction,
- emergent and chaotic behaviors.

▶ We construct an abstraction of the hybrid model as a discrete Markov decision process.

Design and properties of the abstraction

▶ Abstraction ($\mathcal{M}_{\psi,\tau,\Lambda}$) of the hybrid model as a Markov decision process
Depends on property ψ, time step τ, parameter space Λ.

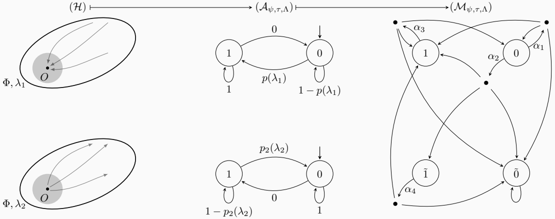


Fig. 8: Abstraction of the hybrid model.

▶ We can use (S)MC on ($\mathcal{M}_{\psi,\tau,\Lambda}$).

▶ Questions on the abstraction $\mathcal{H} \mapsto \mathcal{M}_{\psi,\tau,\Lambda}$: well-defined? computable? complexity? one-to-one? bisimulation? image of a property? invertible?

Example: avoid the extinction

- We consider the property "Avoid **O** after time **T**"

$$\psi_O(\rho, T) = \{\mathbb{U}(t, U_0, \lambda_0) \notin B(O, \rho), t \geq T\}.$$

$\rightsquigarrow$ Its image becomes $\tilde{\psi}_O(\rho, T) = \{\pi(t_s) \notin (1) \cup (\tilde{1}), t_s \geq T\}$.

Proposition / Intuition

Let $\rho^* > \rho$. Assume that $\mathbb{P}\big(\{\mathcal{M}_{\psi,\tau,\Lambda} \models \tilde{\psi}_O(\rho^*, T)\}\big) = 1$. Assume moreover the interpolation condition:

$$\forall k \geq 0, \quad U(t_k) \notin B(O, \rho^*) \wedge U(t_{k+1}) \notin B(O, \rho^*)$$
$$\Rightarrow U(t) \notin B(O, \rho), t_k \leq t \leq t_{k+1}.$$

Then we have

$$\mathbb{P}\big(\{\mathcal{H} \models \psi_O(\rho, T)\}\big) = 1.$$

Example: avoid the extinction

- We consider the property "Avoid **O** after time **T**"

$$\psi_O(\rho, T) = \{\mathbb{U}(t, U_0, \lambda_0) \notin B(O, \rho), t \geq T\}.$$

$\rightsquigarrow$ Its image becomes $\tilde{\psi}_O(\rho, T) = \{\pi(t_s) \notin (1) \cup (\tilde{1}), t_s \geq T\}.$

Proposition / Intuition

Let $\rho^* > \rho$. Assume that $\mathbb{P}\big(\{\mathcal{M}_{\psi,\tau,\Lambda} \models \tilde{\psi}_O(\rho^*, T)\}\big) \geq \xi$. Assume moreover the interpolation condition:

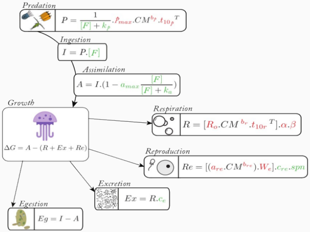
$$\forall k \geq 0, \quad U(t_k) \notin B(O, \rho^*) \wedge U(t_{k+1}) \notin B(O, \rho^*) \\ \Rightarrow U(t) \notin B(O, \rho), t_k \leq t \leq t_{k+1}.$$

Then we have

$$\mathbb{P}\big(\{\mathcal{H} \models \psi_O(\rho, T)\}\big) \geq \tilde{\xi}.$$

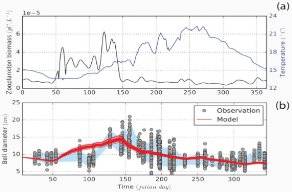
Applications

Goal : Parameterize a growth model of *Pelagia Noctiluca* that fits experimental data

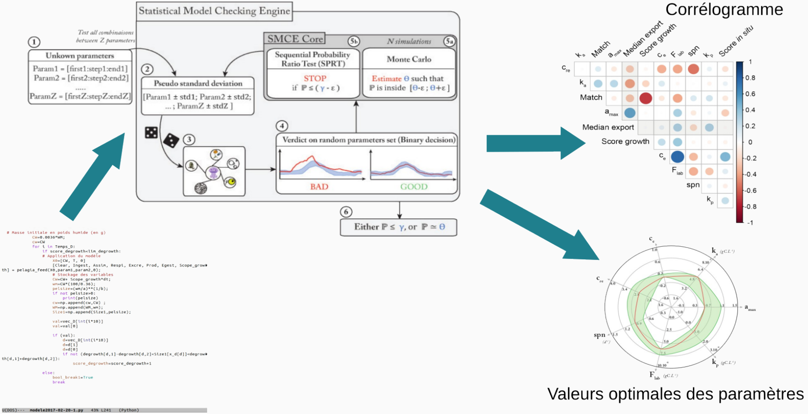


+ Données expérimentales

```
# Masse initiale en poids humide (en g)
            CW=0.0036*WM;
            cw=CW
            for i in Temps_D:
                if score_degrowth<lim_degrowth:
                # Application du modèle
                    X0=[CW, T, 0]
                    [Clear, Ingest, Assim, Respi, Excre, Prod, Egest, Scope_grow
th] = pelagia_feed(X0,param1,param2,0);
                    # Stockage des variables
                    CW=CW+ Scope_growth*dt;
                    wm=CW*(100/0.36);
                    pelsize=(wm/a)**(1/b);
                    if not pelsize>0:
                        print(pelsize)
                    cw=np.append(cw,CW) ;
                    WM=np.append(WM,wm);
                    Size1=np.append(Size1,pelsize);

                    val=vec_D[int(i*10)]
                    val=val[0]

                    if (val):
                        d=vec_D[int(i*10)]
                        d=d[1]
                        d=d[0]
                        if not (degrowth[d,1]-degrowth[d,2]<Size1[x_d[d]]<degrow
th[d,1]+degrowth[d,2]):
                            score_degrowth=score_degrowth+1

                else:
                    bool_break1=True
                    break

U(DOS)---  modele2017-02-20-1.py   43% L241   (Python)
```



Pelagia Noctiluca 2/2

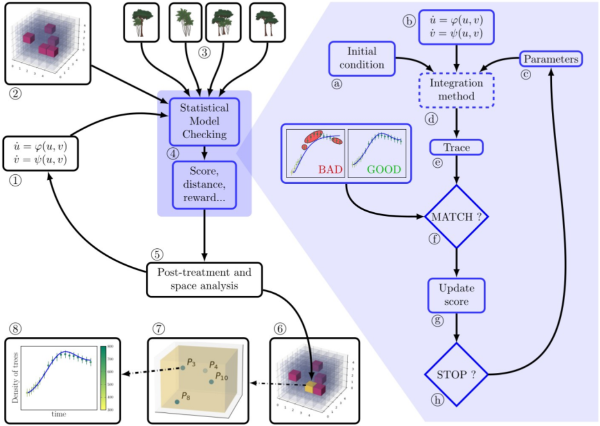


▶ S. Ramondenc, D. Eveillard, L. Guidi, F. Lombard, B. Delahaye. *Probabilistic modeling to estimate jellyfish ecophysiological properties and size distributions.* In Scientific Reports, Vol 10-1, pp 1-13, 2020.

Paracou forest regrowth 1/2

Computational assessment of Amazon forest patches regrowth capacity under strong spatial variability

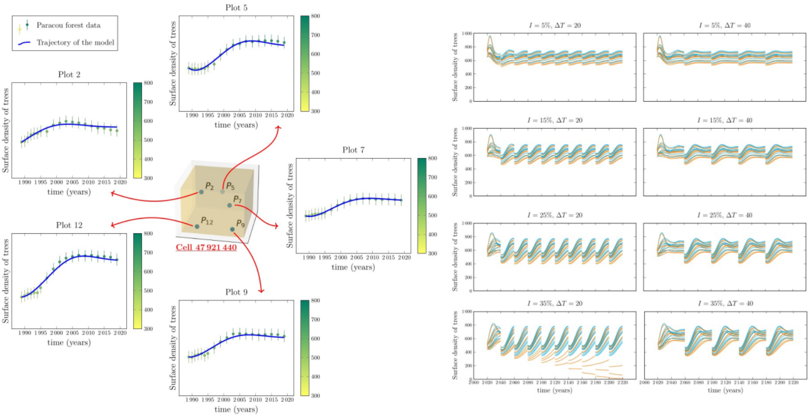
Authors (by alphabetical order): Gilles Ardourel[1], Guillaume Cantin[1,2] Benoît Delahaye[1], Géraldine Derroire[3], Beatriz M.Funatsu[4], David Julien[1].



Paracou forest regrowth 2/2

Computational assessment of Amazon forest patches regrowth capacity under strong spatial variability

Authors (by alphabetical order): Gilles Ardourel[1], Guillaume Cantin[1,2] Benoît Delahaye[1], Géraldine Derroire[3], Beatriz M.Funatsu[4], David Julien[1].



Perspectives and future work

Perspectives and future work

Lots of things to do...

- Dynamic discretization of the parameter grids
- Choice of the integration step
- Guarantees w.r.t. abstraction + coupling with discrete control
- Application to genomic synthesis / epidemiology
- ...

Perspectives and future work

Lots of things to do...

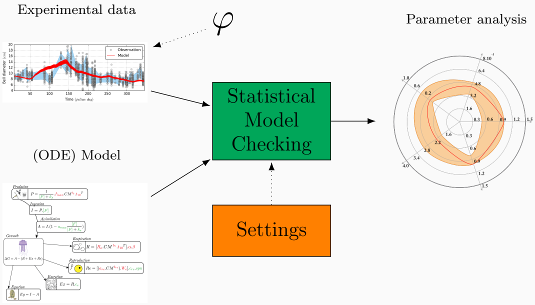
- Dynamic discretization of the parameter grids
- Choice of the integration step
- Guarantees w.r.t. abstraction + coupling with discrete control
- Application to genomic synthesis / epidemiology
- ...

Going further...

- Link/Combination with NN to speed up sampling
- Formal guarantees?
- Link/Combination with agent models
- ...

End

Thanks!



Experimental data
φ
Statistical Model Checking
(ODE) Model
Settings
Parameter analysis

Supplementary materials

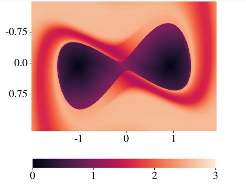


Fig. 9:
$\lambda = (-0.8, -1, 1, 1)$.

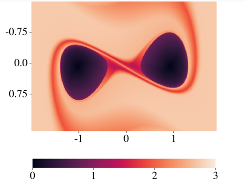


Fig. 10:
$\lambda = (-0.9, -1, 1, 1)$.

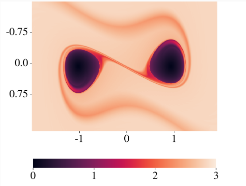


Fig. 11:
$\lambda = (-1, -1, 1, 1)$.

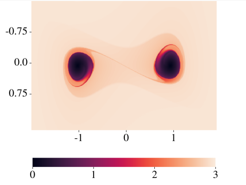


Fig. 12:
$\lambda = (-1.1, -1, 1, 1)$.

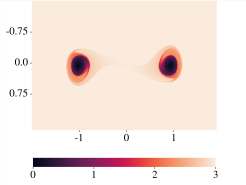


Fig. 13:
$\lambda = (-1.2, -1, 1, 1)$.