

Computing with real numbers: Robustness in computability and complexity over the reals

Manon Blanc



université
PARIS-SACLAY

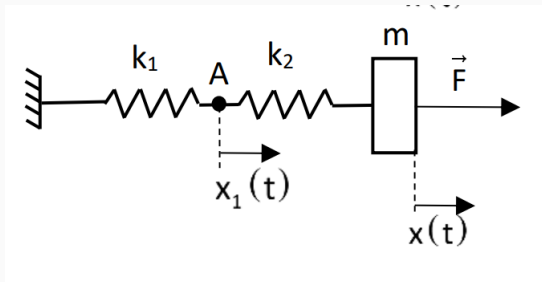


Introduction

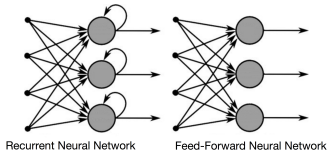
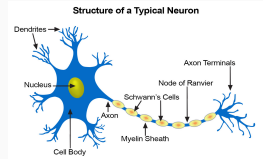
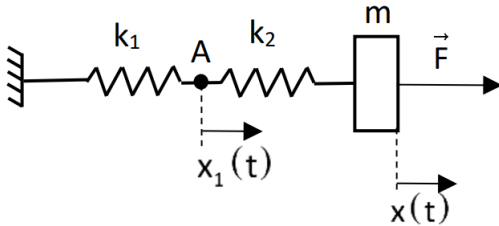
- Why algebraic characterisations?

- Why algebraic characterisations?
- We already do it for computability classes
- Computation over the reals : Framework of computable analysis

Introduction



Introduction



- Framework of **implicit complexity** (Cobham, Bellantoni, Cook, Levant, Marion, de Naurois...)
- **Time = Length:**
 - Continuous ODEs : Bournez, Graça, Pouly (ICALP 2017)
 - Discrete ODEs : Bournez, Durand (MFCS 2019)

**A type of ODE for algebraic
characterisations: PTIME**

Discrete derivation

$$\frac{\delta f(\mathbf{x}, \mathbf{y})}{\delta \mathbf{x}} = f(\mathbf{x} + 1, \mathbf{y}) - f(\mathbf{x}, \mathbf{y})$$

→ It is a derivation with a step of one

“Length ODE”?

Definition (Length ODE)

A function \mathbf{f} is “length-ODE” definable (from u , g and h) if it is a solution of:

$$f(0, \mathbf{y}) = \mathbf{g}(\mathbf{y}) \quad \text{and} \quad \frac{\delta \mathbf{f}(x, \mathbf{y})}{\delta \ell} = \mathbf{u}(\mathbf{f}(x, \mathbf{y}), \mathbf{h}(x, \mathbf{y}), x, \mathbf{y}). \quad (1)$$

“Length ODE”?

Definition (Length ODE)

A function \mathbf{f} is “length-ODE” definable (from u , g and h) if it is a solution of:

$$f(0, \mathbf{y}) = \mathbf{g}(\mathbf{y}) \quad \text{and} \quad \frac{\delta \mathbf{f}(x, \mathbf{y})}{\delta \ell} = \mathbf{u}(\mathbf{f}(x, \mathbf{y}), \mathbf{h}(x, \mathbf{y}), x, \mathbf{y}). \quad (1)$$

Formal synonym for right-hand side of (1):

$$\mathbf{f}(x + 1, \mathbf{y}) = \mathbf{f}(x, \mathbf{y}) + (\ell(x + 1) - \ell(x)) \cdot \mathbf{u}(\mathbf{f}(x, \mathbf{y}), \mathbf{h}(x, \mathbf{y}), x, \mathbf{y})$$

“Length ODE”?

$$\mathbf{f}(x + 1, \mathbf{y}) = \mathbf{f}(x, \mathbf{y}) + (\ell(x + 1) - \ell(x)) \cdot \mathbf{u}(\mathbf{f}(x, \mathbf{y}), \mathbf{h}(x, \mathbf{y}), x, \mathbf{y})$$

Derivation with respect to the length = change of variable.

Variation when:

$$\ell(x + 1) - \ell(x) \neq 0$$

Inspired by:

$$\frac{\delta f(x, \mathbf{y})}{\delta x} = \frac{\delta \ell(x)}{\delta x} \cdot \frac{\delta f(x, \mathbf{y})}{\delta \ell(x)}.$$

Motivation behind Length-ODEs



Fig. 5. A continuous system before and after an exponential speed-up.

θ solution of:

$$y' = f(y)$$

$$f : \mathbb{R} \rightarrow \mathbb{R}$$

ϕ solution of:

$$z = z'$$

$$y' = f(y)z$$

Motivation behind Length-ODEs



Fig. 5. A continuous system before and after an exponential speed-up.

$$\text{Re-scaling: } \phi_1(t) = \theta(e^t)$$

Motivation behind Length-ODEs

Now “time-complexity” is measured by the **length** of the solution curve of the ODE.

Invariance by rescaling

Example of length-ODE

$$f(0) = 2$$

$$\frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$

Example of length-ODE

$$f(0) = 2$$

$$\frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$

Unique solution : $f(x) = 2^{2^{\ell(x)}}$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$
$$f(x+1) - f(x) = (\ell(x+1) - \ell(x))(f(x)f(x) - f(x))$$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$
$$f(x+1) - f(x) = (\ell(x+1) - \ell(x))(f(x)f(x) - f(x))$$

$$z = \ell(x)$$

$$F(z) = 2^{2^z}$$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$
$$f(x+1) - f(x) = (\ell(x+1) - \ell(x))(f(x)f(x) - f(x))$$

$$z = \ell(x)$$

$$F(z) = 2^{2^z}$$

- We have $f(x) = F(z)$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$
$$f(x+1) - f(x) = (\ell(x+1) - \ell(x))(f(x)f(x) - f(x))$$

$$z = \ell(x)$$

$$F(z) = 2^{2^z}$$

- We have $f(x) = F(z)$
- $F(z+1) = 2^{2^z+2^z} = F(z)F(z)$

Example of length-ODE

$$f(0) = 2; \quad \frac{\delta f}{\delta \ell}(x) = f(x) \cdot f(x) - f(x)$$
$$f(x+1) - f(x) = (\ell(x+1) - \ell(x))(f(x)f(x) - f(x))$$

$$z = \ell(x)$$

$$F(z) = 2^{2^z}$$

- We have $f(x) = F(z)$
- $F(z+1) = 2^{2^{z+1}} = F(z)F(z)$
- Then, $F(z+1) = F(z) + (z+1 - z)(F(z)F(z) - F(z))$

u is essentially linear in f iff:

$$u = A(\dots, \sigma(f))f(x) + B(\dots, \sigma(f))$$

with A and B that may depend on a sigmoid over f .

Characterisation of PTIME (the one you are used to)

We consider

$$\text{LDL} = [0, 1, \pi_i^k, \ell(x), +, -, \times, \text{cond}(x);$$

composition, linear length ODE]

Theorem (Bournez & Durand, '19)

$$\text{LDL} \cap \mathbb{N}^{\mathbb{N}} = \text{PTIME} \cap \mathbb{N}^{\mathbb{N}}$$

Some basic definitions of computable analysis

Definitions (from computable analysis): intuition

- What is a **computable real number** x ?
 - we can compute a representation of x .

Definitions (from computable analysis): intuition

- What is a **computable real number** x ?
→ we can compute a representation of x .

Example

e, π are computable. $\sum_{i \geq 1} 2^{-BB(i)}$, where BB is the Busy Beavers function is not.

Definitions (from computable analysis): intuition

- What is a **computable real number** x ?
→ we can compute a representation of x .

Example

e, π are computable. $\sum_{i \geq 1} 2^{-BB(i)}$, where BB is the Busy Beavers function is not.

- What is a **computable function** $f : \mathbb{R} \rightarrow \mathbb{R}$?
→ on a representation of $x \in \mathbb{R}$, we produce a representation of $f(x)$.

Definitions (from computable analysis): intuition

- What is a **computable real number** x ?
→ we can compute a representation of x .

Example

e, π are computable. $\sum_{i \geq 1} 2^{-BB(i)}$, where BB is the Busy Beavers function is not.

- What is a **computable function** $f : \mathbb{R} \rightarrow \mathbb{R}$?
→ on a representation of $x \in \mathbb{R}$, we produce a representation of $f(x)$.
- How do we define **complexity over the reals**?

More formally

- What is a **computable real number** x ?

There exists a Cauchy sequence $\phi : \mathbb{N} \rightarrow \mathbb{D}$ such that for all $n \in \mathbb{N}$, $|\phi(n) - x| \leq 2^{-n}$.

More formally

- What is a **computable real number** x ?

There exists a Cauchy sequence $\phi : \mathbb{N} \rightarrow \mathbb{D}$ such that for all $n \in \mathbb{N}$, $|\phi(n) - x| \leq 2^{-n}$.

- What is a **computable function** $f : \mathbb{R} \rightarrow \mathbb{R}$?

There exists an oracle Turing machine M such that, on a Cauchy sequence ϕ converging to x , M queries the oracle to have m such that $|\phi(m) - x| \leq 2^{-m}$ and computes $d \in \mathbb{D}$ such that $|d - f(x)| \leq 2^{-n}$.

PTIME over the reals with Discrete ODEs

First step: real sequences

Definition

$f : \mathbb{R} \rightarrow \mathbb{R}$ is an **effective limit** of $\bar{f} : \mathbb{R} \times \mathbb{N} \rightarrow \mathbb{R}$ if

$$|f(x) - \bar{f}(x, 2^n)| \leq 2^{-n}$$

Key observation : \bar{f} computable in polynomial time $\Rightarrow f$ computable in polynomial time.

Algebraic characterisation of PTIME for sequences

We consider:

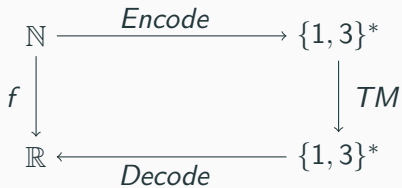
$$\overline{\text{LDL}}^\bullet = [\mathbf{0}, \mathbf{1}, \pi_i^k, \ell(x), +, -, \times, \overline{\text{cond}}(x), \frac{x}{2};$$

composition, linear length ODE, effective limit]

Theorem (B., Bournez, MCU22)

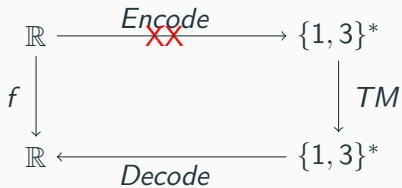
$$\text{FPTime} \cap \mathbb{R}^{\mathbb{N}} = \overline{\text{LDL}}^\bullet \cap \mathbb{R}^{\mathbb{N}}$$

“Proof”



Second step: functions over the reals with discrete ODEs

What we cannot do



FPTIME for computable functions over the reals

We consider:

$$\overline{\text{LDL}}^\circ = [\mathbf{0}, \mathbf{1}, \pi_i^k, \ell(x), +, -, \tanh, \frac{x}{2}, \frac{x}{3};$$

composition, linear length ODE, effective limit]

Theorem (B., Bournez, MFCS23)

$$\overline{\text{LDL}}^\circ \cap \mathbb{R}^{\mathbb{R}} = \text{FPTIME} \cap \mathbb{R}^{\mathbb{R}}$$

Proof ideas (\supseteq)

- We give *continuous approximations* of some basic functions (integer/fractional part, modulo 2, ...);

Proof ideas (\supseteq)

- We give *continuous approximations* of some basic functions (integer/fractional part, modulo 2, ...);
- We *encode each step of the transition of the Turing machine* into the algebra, using the previous functions : Next;

Proof ideas (\supseteq)

- We give *continuous approximations* of some basic functions (integer/fractional part, modulo 2, ...);
- We *encode each step of the transition of the Turing machine* into the algebra, using the previous functions : Next;
- We obtain a *linear length ODE* encoding the execution of the Turing machine: $State(t + 1) = Next(State(t))$ so

$$\frac{\delta \overline{State}(2^{t+1})}{\delta \ell} = \overline{Next}(\overline{State}(2^t))$$

PSPACE with Discrete ODEs

Definition (Robust linear ODE)

A bounded function \mathbf{f} is a robustly linear ODE definable from g and u (with \mathbf{u} *essentially linear* in $\mathbf{f}(x, \mathbf{y})$) if:

1. it is a solution of

$$\mathbf{f}(0, \mathbf{y}) = \mathbf{g}(\mathbf{y}) \quad \text{and} \quad \frac{\delta \mathbf{f}(x, \mathbf{y})}{\delta x} = \mathbf{u}(\mathbf{f}(x, \mathbf{y}), \mathbf{h}(x, \mathbf{y}), x, \mathbf{y}),$$

2. the ODE is (polynomially) **numerically stable**.

FPSPACE for computable functions over the reals

We consider:

$$\overline{\text{RLD}}^\circ = [\mathbf{0}, \mathbf{1}, \pi_i^k, \ell(n), +, -, \tanh, \frac{x}{2}, \frac{x}{3};$$

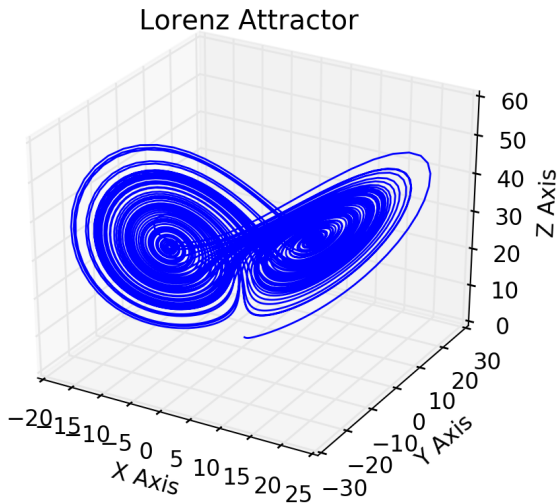
composition, robust linear ODE, ELim]

Theorem (B., Bournez, MFCS23)

$$\overline{\text{RLD}}^\circ \cap \mathbb{R}^{\mathbb{R}} = \text{FPSPACE} \cap \mathbb{R}^{\mathbb{R}}$$

Functions over the reals with continuous ODEs

Why it is not easy to solve ODEs



Why it is not easy to solve ODEs

$$\mathbf{f}(0, \mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{and} \quad \frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial t} = \mathbf{u}(\mathbf{f}(t, \mathbf{x}), t, \mathbf{x}),$$

- Non-computable in the **general case**, for \mathbf{u} computable (Pour-El, Richards)

Why it is not easy to solve ODEs

$$\mathbf{f}(0, \mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{and} \quad \frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial t} = \mathbf{u}(\mathbf{f}(t, \mathbf{x}), t, \mathbf{x}),$$

- Non-computable in the **general case**, for \mathbf{u} computable (Pour-El, Richards)
- Computable for \mathbf{u} **computable** and unique solution (Collins, Graça, Ruohonen)

Why it is not easy to solve ODEs

$$\mathbf{f}(0, \mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{and} \quad \frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial t} = \mathbf{u}(\mathbf{f}(t, \mathbf{x}), t, \mathbf{x}),$$

- Non-computable in the **general case**, for \mathbf{u} computable (Pour-El, Richards)
- Computable for \mathbf{u} **computable** and unique solution (Collins, Graça, Ruohonen)
- PSPACE-completeness on a **bounded domain** for \mathbf{u} computable in polynomial-time (Kawamura, Ko)

Robust Continuous ODEs (1/2)

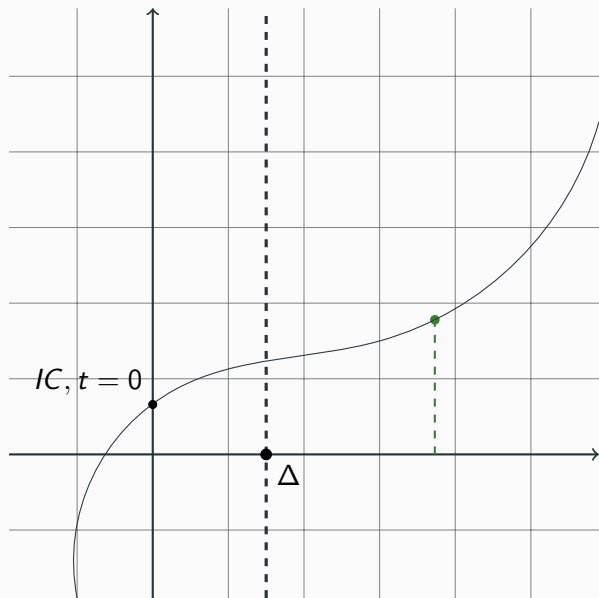
A function $\mathbf{f} : \mathbb{R} \rightarrow \mathbb{R}$ is robustly ODE definable (from initial condition \mathbf{g} , and dynamic \mathbf{u}) if

- it corresponds to the solution of the following continuous ODE:

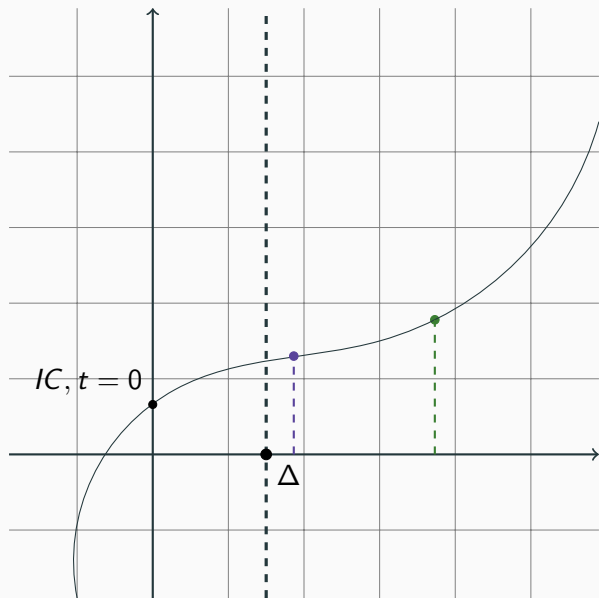
$$\mathbf{f}(0, \mathbf{x}) = \mathbf{g}(\mathbf{x}) \quad \text{and} \quad \frac{\partial \mathbf{f}(t, \mathbf{x})}{\partial t} = \mathbf{u}(\mathbf{f}(t, \mathbf{x}), t, \mathbf{x}), \quad (2)$$

- And polynomially numerically stable (to be defined : next slides)
→ this allows computation by dichotomy.

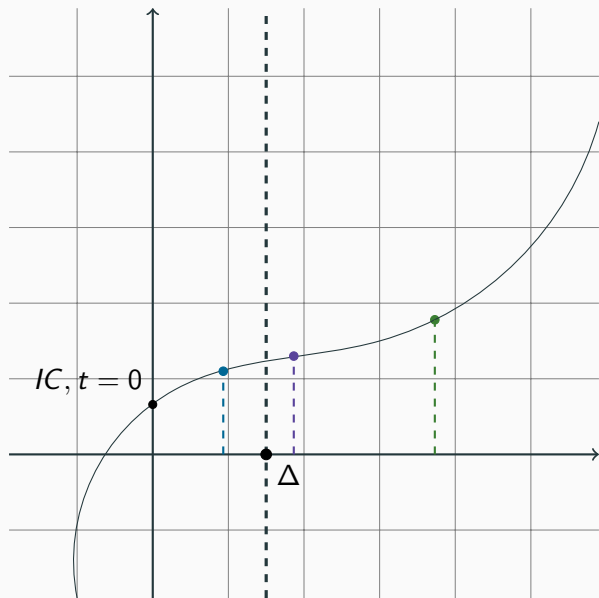
An unusual way of solving an ODE



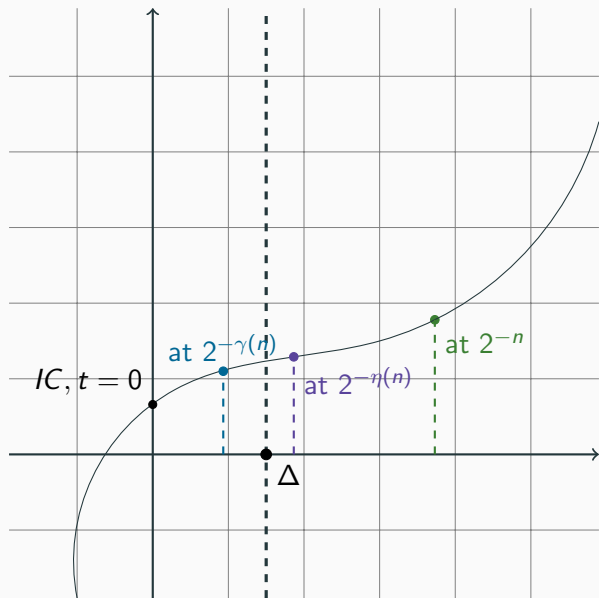
An unusual way of solving an ODE



An unusual way of solving an ODE



An unusual way of solving an ODE



An unusual way of solving an ODE

- Computation at $t = x$ at precision 2^{-n} , with initial condition $t = 0$
- Computation at $t = t_0 + \frac{x}{2}$ at precision $2^{-\eta(n)}$, with initial condition $t_0 = 0$ and $t_0 = \frac{x}{2}$
- Computation at $t = t_0 + \frac{x}{4}$ at precision $2^{-\gamma(n)}$, with initial condition $t_0 = 0$ and $t_0 = \frac{x}{4}$ and $t_0 = \frac{x}{2}$ and $t_0 = \frac{3x}{4}$

Robust Continuous ODEs (2/2)

More “formally” :

- $\exists \Delta \in \mathbb{Q}_+^*$, such that the previous ODE is (polynomially spaced) **solvable** on $[0, \Delta]$.
- For $t \geq \Delta$, we can compute $\mathbf{f}(t, \mathbf{x})$ at 2^{-n} by computing some approximation $\widetilde{\mathbf{f}(t/2, \mathbf{x})}$ of $\mathbf{f}(t/2, \mathbf{x})$ at precision $2^{-\eta(n)}$.

Characterisation of PSPACE with continuous ODEs

We consider :

$$\overline{\text{RCD}} = [\mathbf{0}, \mathbf{1}, \pi_i^k, +, -, \times, \tanh, \cos, \pi, \frac{x}{2}, \frac{x}{3};$$

composition, robust continuous ODE, ELim]

Theorem (B., Bournez, ICALP24)

$$\overline{\text{RCD}} \cap \mathbb{R}^{\mathbb{R}} = \text{FPSPACE} \cap \mathbb{R}^{\mathbb{R}}$$

$$\overline{\text{RLD}^\circ} \cap \mathbb{R}^{\mathbb{R}} = \overline{\text{RCD}} \cap \mathbb{R}^{\mathbb{R}}$$

$$\overline{\text{RLD}^\circ} \cap \mathbb{R}^{\mathbb{R}} = \overline{\text{RCD}} \cap \mathbb{R}^{\mathbb{R}}$$

- **Discrete** \Rightarrow **Continuous settings:**

Adaptation of the a trick due to Branicky ('95)

$$\overline{\text{RLD}^\circ} \cap \mathbb{R}^{\mathbb{R}} = \overline{\text{RCD}} \cap \mathbb{R}^{\mathbb{R}}$$

- **Discrete** \Rightarrow **Continuous settings:**

Adaptation of the a trick due to Branicky ('95)

- **Continuous** \Rightarrow **Discrete settings:**

From our definitions of robustness of ODEs

Partial Conclusion

Parital Conclusion

- We have an algebraic characterisations of PTIME and PSPACE over the reals using *discrete* and *continuous ODEs*.
- The last algebra might not be minimal
- Further work :
 - talking about NP (non-determinism)?
 - talking about probabilistic classes
 - talking about distributions

More results on robustness: study of dynamical systems

Introduction: robustness in Dynamical Systems

- Motivated by the field of verification.
- Informal conjecture: undecidability in verification does not happen for robust systems.

Introduction: robustness in Dynamical Systems

- Approach: Asarin and Bouajjani (LICS '01).
- Our goal:
 - go from computability to complexity,

Introduction: robustness in Dynamical Systems

- Approach: Asarin and Bouajjani (LICS '01).
- Our goal:
 - go from computability to complexity,
 - quantifying the robustness to characterise PSPACE,

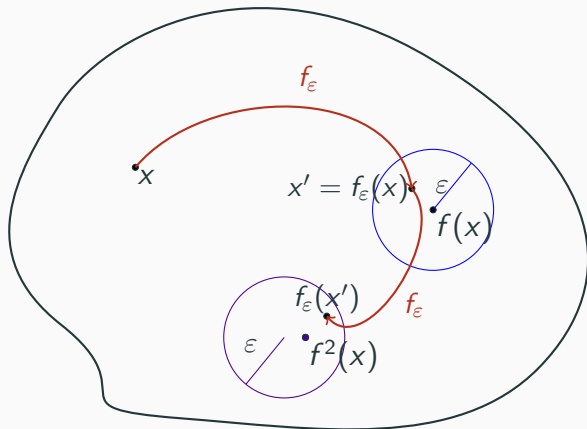
Introduction: robustness in Dynamical Systems

- Approach: Asarin and Bouajjani (LICS '01).
- Our goal:
 - go from computability to complexity,
 - quantifying the robustness to characterise PSPACE,
 - having a notion of robustness to characterise PTIME,

Introduction: robustness in Dynamical Systems

- Approach: Asarin and Bouajjani (LICS '01).
- Our goal:
 - go from computability to complexity,
 - quantifying the robustness to characterise PSPACE,
 - having a notion of robustness to characterise PTIME,
 - “applying” those notions to tiling.

Introduction: robustness in Dynamical Systems



It is not necessarily the same notion
of robustness as before

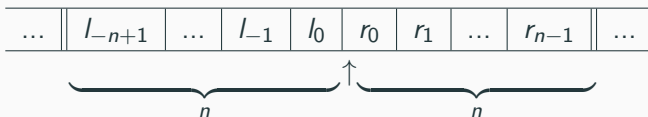
Introduction: robustness in Dynamical Systems

Frame : We want to talk about **discrete- and continuous-time dynamical systems**. We first use the framework of Turing machines
(seen as a particular dynamical system).

Space-perturbation of a Dynamical System

n-perturbation of a TM

Let \mathcal{M}_n , the *n-space-perturbed version* of TM \mathcal{M} : the idea is that the *n*-perturbed version of the machine \mathcal{M} is unable to remain correct at a distance more than *n* from the head of the machine.



Some Properties (from Asarin and Bouajjani '01)

Let $L_n(\mathcal{M})$ be the n -perturbed language of \mathcal{M} .

- By definition:
 - a word accepted by \mathcal{M} is also recognised by all the \mathcal{M}_n 's.

Some Properties (from Asarin and Bouajjani '01)

Let $L_n(\mathcal{M})$ be the n -perturbed language of \mathcal{M} .

- By definition:
 - a word accepted by \mathcal{M} is also recognised by all the \mathcal{M}_n 's.
 - $L_{n+1}(\mathcal{M}) \subseteq L_n(\mathcal{M})$.

Some Properties (from Asarin and Bouajjani '01)

Let $L_n(\mathcal{M})$ be the n -perturbed language of \mathcal{M} .

- By definition:
 - a word accepted by \mathcal{M} is also recognised by all the \mathcal{M}_n 's.
 - $L_{n+1}(\mathcal{M}) \subseteq L_n(\mathcal{M})$.
 - $L_n(\mathcal{M})$ is co-c.e.

Some Properties (from Asarin and Bouajjani '01)

Let $L_n(\mathcal{M})$ be the n -perturbed language of \mathcal{M} .

- By definition:
 - a word accepted by \mathcal{M} is also recognised by all the \mathcal{M}_n 's.
 - $L_{n+1}(\mathcal{M}) \subseteq L_n(\mathcal{M})$.
 - $L_n(\mathcal{M})$ is co-c.e.
- So we have
 - $L_\omega(\mathcal{M}) = \bigcap_n L_n(\mathcal{M})$: this is the set of words accepted by \mathcal{M} when subject to arbitrarily “small” perturbations.

Some Properties (from Asarin and Bouajjani '01)

Let $L_n(\mathcal{M})$ be the n -perturbed language of \mathcal{M} .

- By definition:
 - a word accepted by \mathcal{M} is also recognised by all the \mathcal{M}_n 's.
 - $L_{n+1}(\mathcal{M}) \subseteq L_n(\mathcal{M})$.
 - $L_n(\mathcal{M})$ is co-c.e.
- So we have
 - $L_\omega(\mathcal{M}) = \bigcap_n L_n(\mathcal{M})$: this is the set of words accepted by \mathcal{M} when subject to arbitrarily “small” perturbations.
 - $L(\mathcal{M}) \subseteq L_\omega(\mathcal{M}) \subseteq \dots \subseteq L_2(\mathcal{M}) \subseteq L_1(\mathcal{M})$.

Theorem (Perturbed reachability is co-c.e., from Asarin-Bouajjani 01)

$$L_\omega(\mathcal{M}) \in \Pi_1^0.$$

Say L is robust if $L = L_\omega$:

L robust $\Rightarrow L$ decidable.

- Definition of the notion of perturbation of a TM
- Some properties

A Characterisation of PSPACE

Definition

For $f : \mathbb{N} \rightarrow \mathbb{N}$, we write $L_{\{f\}}(\mathcal{M})$ the set of words accepted by \mathcal{M} with a "space-perturbation" f :

$$L_{\{f\}}(\mathcal{M}) = \{w \mid w \in L_{f(\ell(w))}(\mathcal{M})\}.$$

Theorem (CSL24, B., Bournez)

$L \in \text{PSPACE}$ iff for some \mathcal{M} and some polynomial p ,
 $L = L(\mathcal{M}) = L_{\{p\}}(\mathcal{M})$.

- Characterisation of PSPACE with space-perturbated TMs

Reachability Relations

Reachability Relation for Dynamical Systems

To each rational discrete-time dynamical system \mathcal{P} is associated its reachability relation $R^{\mathcal{P}}(\cdot, \cdot)$ on $\mathbb{Q}^d \times \mathbb{Q}^d$.

→ two rational points \mathbf{x} and \mathbf{y} , $R^{\mathcal{P}}(\mathbf{x}, \mathbf{y})$ holds iff there exists a trajectory of \mathcal{P} from \mathbf{x} to \mathbf{y} .

Reachability Relation With Small Perturbations

- Consider now a discrete-time dynamical system \mathcal{P} with function \mathbf{f} Lipschitz on a compact domain:
For $\varepsilon > 0$ we consider the ε -perturbed system \mathcal{P}_ε .
- Its trajectories are defined as sequences \mathbf{x}_t satisfying $d(\mathbf{x}_{t+1}, \mathbf{f}(\mathbf{x}_t)) < \varepsilon$ for all t .

Reachability Relation With Small Perturbations

- We denote reachability in the system \mathcal{P}_ϵ by $R_\epsilon^{\mathcal{P}}(\cdot, \cdot)$.
- $R_\omega^{\mathcal{P}} = \bigcap_\epsilon R_\epsilon^{\mathcal{P}}(\cdot, \cdot)$ is co-c.e..
- Say $R^{\mathcal{P}}$ is robust when $R_\omega^{\mathcal{P}} = R^{\mathcal{P}}$:
 $R^{\mathcal{P}}$ robust $\Rightarrow R^{\mathcal{P}}$ computable.

Proposition (Robust \Leftrightarrow reachability true or ϵ -far from being true)

We have $R_\omega^{\mathcal{P}} = R^{\mathcal{P}}$ iff for all $\mathbf{x}, \mathbf{y} \in \mathbb{Q}^d$, either $R^{\mathcal{P}}(\mathbf{x}, \mathbf{y})$ is true or $R^{\mathcal{P}}(\mathbf{x}, \mathbf{y})$ is false and there exists $\epsilon > 0$ such that it is ϵ -far from being true.

\Rightarrow relation with δ -reachability (Gao, Kong, Chen, Clarke '06).

Towards real numbers : Ball decision problem

Input:

- A point $\mathbf{x} \in \mathbb{Q}$
- A set B
- A dynamic given by the function $\mathbf{f} \in \mathbb{R}^{\mathbb{R}}$
- The promise that the dynamics starting from \mathbf{x} never ends up on the border of B .

Towards real numbers : Ball decision problem

Input:

- A point $\mathbf{x} \in \mathbb{Q}$
- A set B
- A dynamic given by the function $\mathbf{f} \in \mathbb{R}^{\mathbb{R}}$
- The promise that the dynamics starting from \mathbf{x} never ends up on the border of B .

Output: Does the dynamics starting from \mathbf{x} reach B ?

Polynomially robust to precision \Rightarrow PSPACE:

Theorem (CSL24, B., Bournez)

Take a Lipschitz system on a compact, with \mathbf{f} poly. time computable, whose domain X is a closed rational box, and that for all rational \mathbf{x} , $R^{\mathcal{P}}(\mathbf{x})$ is closed and $R^{\mathcal{P}}(\mathbf{x}) = R^{\mathcal{P}}_{\{p\}}(\mathbf{x})$ for a polynomial p . Then the ball decision problem is in PSPACE.

Complexity of reachability relations

Polynomially robust to precision \Leftrightarrow PSPACE-completeness

Theorem (CSL24, B., Bournez)

Any PSPACE language is reducible to PAM's reachability relation:

$$R^{\mathcal{P}} = R^{\mathcal{P}}_{\{p\}}, \text{ for some polynomial } p.$$

- Perturbated reachability relation for rational dynamical systems
- Characterisation of PSPACE in that framework (with additionnal properties)

Alternative view: Drawability and extension to the reals

But we could also see it as a relation over the reals and use the framework of CA, regarding subsets of $\mathbb{R}^d \times \mathbb{R}^d$.

Definition

A closed subset of \mathbb{R}^d is said *c.e. closed* if we can effectively enumerate the rational open balls intersecting it.

From the statements of CA, the following holds:

Theorem

Consider a computable discrete time system \mathcal{P} whose domain is a computable compact. For all \mathbf{x} , $\text{cls}(R^{\mathcal{P}}(\mathbf{x})) \subseteq \mathbb{R}^d$ is a c.e. closed subset.

And With Reachability Properties

Theorem (Perturbed reachability is co-c.e.)

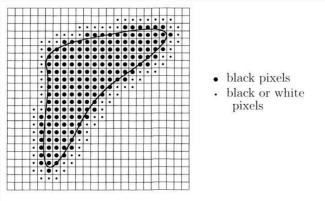
Consider a dynamical system, with \mathbf{f} locally Lipschitz, computable, whose domain is a computable compact, then, for all \mathbf{x} ,

$\text{cls}(R_{\omega}^{\mathcal{P}}(\mathbf{x})) \subseteq \mathbb{R}^d$ is a co-c.e. closed subset.

Say $R^{\mathcal{P}}$ is robust when $R_{\omega}^{\mathcal{P}} = R^{\mathcal{P}}$: $R^{\mathcal{P}}$ robust $\Rightarrow R^{\mathcal{P}}$ computable.

Plot twist

From CA, Computable \Leftrightarrow can be plotted.



Theorem

$R^{\mathcal{P}}$ closed and can be plotted in a name of $\mathbf{f} \Leftrightarrow$ the system is robust, i.e. $R_{\omega}^{\mathcal{P}} = R^{\mathcal{P}}$.

- We have a nice extension to geometric properties : what can be drawn is robust

Time-perturbation

Another type of perturbation

Definition

Given $f : \mathbb{N} \rightarrow \mathbb{N}$, we write $L^{\{f\}}(\mathcal{M})$ for the set of words accepted by \mathcal{M} with time perturbation f :

$$L^{\{f\}}(\mathcal{M}) = \{w \mid w \in L^{f(\ell(w))}(\mathcal{M})\}.$$

Another Characterisation of PTIME with length

Theorem (Polynomially robust to time \Leftrightarrow PTIME)

- A language $L \in \text{PTIME}$ iff for some \mathcal{M} and some polynomial p , $L = L(\mathcal{M}) = L^{\{p\}}(\mathcal{M})$.
- Any PTIME language is reducible to PAM's reachability: $R^{\mathcal{P}} = R^{\mathcal{P},(p)}$ for some polynomial p .

A Characterisation of PTIME

Theorem (Polynomially length robust \Rightarrow PTIME)

Assume distance d is time metric and $R^{\mathcal{P}} = R^{\mathcal{P},(p)}$ for some polynomial p . Then $R^{\mathcal{P}} \in \text{PTIME}$.

- Definition of another type of perturbation
- Characterisation of PTIME

Partial Conclusion

Partial Conclusion

- Characterisation of FPSPACE with the suitable notion of perturbation.
- Characterisation of PTIME with the suitable notion of perturbation.
- Extension to drawability.

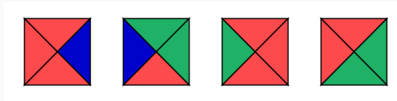
Extending robustness: Towards tilings

Still not necessarily the same notion
of robustness as before

- Applying our notion of robustness to tilings :
what makes the problem of tiling *undecidable*?
- What can make the *problem of some tiling* decidable?

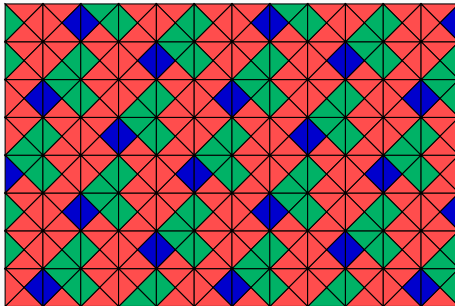
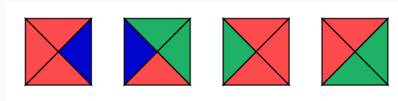
Wang Tileset

- A set of tiles with compatibility rules



Wang Tileset

- A set of tiles with compatibility rules



Theorem (Easy inclusion, compacity argument)

*Given a set of tiles, the problem of knowing whether it tiles the plan is *co-computably enumerable*.*

Conjecture (Wang '61) : all the tiling of the plan are *periodic*.

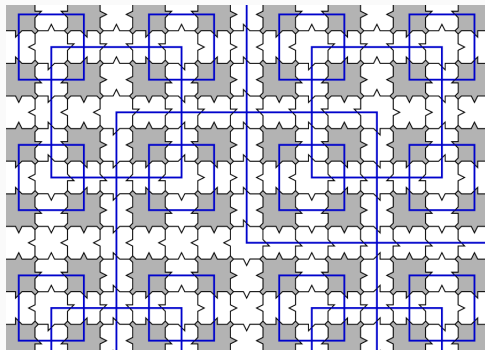
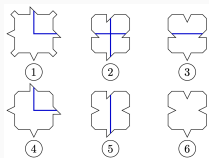
Theorem (Easy inclusion, compactness argument)

*Given a set of tiles, the problem of knowing whether it tiles the plan is *co-computably enumerable*.*

Conjecture (Wang '61) : all the tiling of the plan are *periodic*.

FALSE!!

Counter-example : Robinson tileset '78

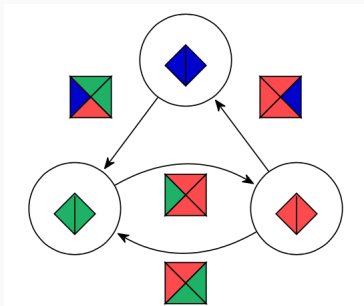


Theorem (Berger '64)

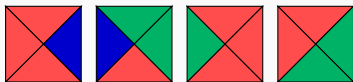
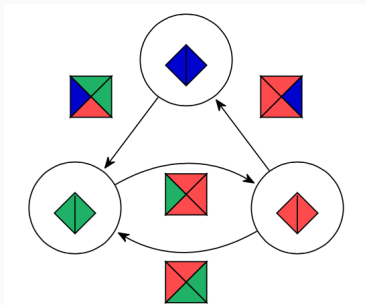
*Given a set of tiles, the problem of knowing whether it tiles the plane is **undecidable**.*

The formalism of transducers

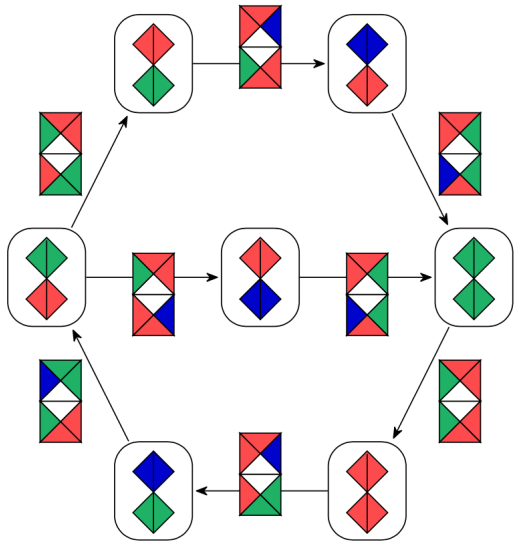
Transducers



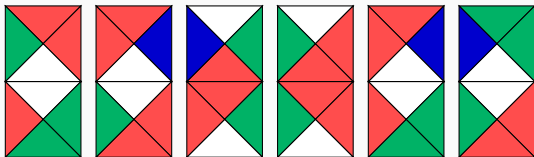
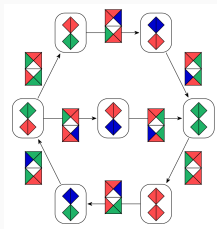
Example



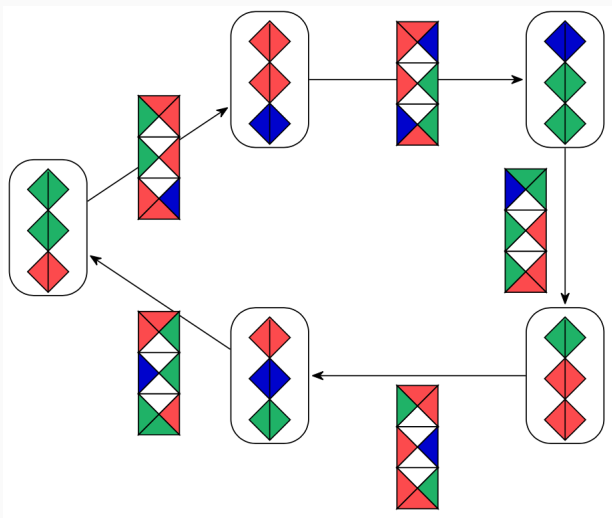
(Meta-)Transducers



Example



(Meta-)Transducers



The notion of robustness

Some notations

Definition (Notation $\mathcal{T} \succ_{\varepsilon} \mathcal{T}'$)

Given two transducers \mathcal{T} and \mathcal{T}' , we write $\mathcal{T} \succ_{\varepsilon} \mathcal{T}'$ iff every edge of the transducer \mathcal{T}' is an edge of \mathcal{T} .

Some notations

Definition (Notation $\mathcal{T} \succ_{\circ} \mathcal{T}'$)

Given two transducers \mathcal{T} and \mathcal{T}' , we write $\mathcal{T} \succ_{\circ} \mathcal{T}'$ iff every edge of the transducer \mathcal{T}' is an edge of \mathcal{T} .

Definition (Composition pattern)

Consider a finite set S of transducers. Consider the signature made of the symbols of S (with arity 0) and the symbol \circ (of arity 2). A *composition pattern* F over S is a term over this signature.

Definition (Robustness)

A tileset τ is *provability robust* if there exists a family of transducers $(\mathcal{T}_n)_n$, of respective heights $(g(n))_n$, with $g : \mathbb{N}^* \rightarrow \mathbb{N}^*$ injective with $g(1) = 1$ and some (fixed) composition pattern $F = F(\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n)$ over $\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n$ for some integer k , such that:

1. Initialisation: $\tau \succ_{\circ} \mathcal{T}_1, \dots, \tau^{g(k+1)} \succ_{\circ} \mathcal{T}_{k+1}$

Definition (Robustness)

A tiling τ is *provability robust* if there exists a family of transducers $(\mathcal{T}_n)_n$, of respective heights $(g(n))_n$, with $g : \mathbb{N}^* \rightarrow \mathbb{N}^*$ injective with $g(1) = 1$ and some (fixed) composition pattern $F = F(\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n)$ over $\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n$ for some integer k , such that:

1. Initialisation: $\tau \succ \mathcal{T}_1, \dots, \tau^{g(k+1)} \succ \mathcal{T}_{k+1}$
2. F provides an invariant:

$$\forall n, \quad F(\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n) \succ \mathcal{T}_{n+1} \quad (3)$$

Definition (Robustness)

A tiling τ is *provability robust* if there exists a family of transducers $(\mathcal{T}_n)_n$, of respective heights $(g(n))_n$, with $g : \mathbb{N}^* \rightarrow \mathbb{N}^*$ injective with $g(1) = 1$ and some (fixed) composition pattern $F = F(\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n)$ over $\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n$ for some integer k , such that:

1. Initialisation: $\tau \succ \mathcal{T}_1, \dots, \tau^{g(k+1)} \succ \mathcal{T}_{k+1}$
2. F provides an invariant:

$$\forall n, \quad F(\mathcal{T}_1, \mathcal{T}_{n-k}, \dots, \mathcal{T}_n) \succ \mathcal{T}_{n+1} \quad (3)$$

3. For all n , \mathcal{T}_n contains a loop.

Theorem (Aubrun, B., Bournez)

Let τ a tileset. τ is *provably robust* $\Leftrightarrow \tau$ admits a tiling.

→ the domino problem for provably robust tileset is computable

Partial Conclusion

Partial Conclusion

- We have a criterion to prove if a tileset can tile a plan.
- We applied it to some well-known tilesets, including Jeandel-Rao 2015 tileset and Robinson 1978 tileset.
- Current work : a criterion to prove if a tileset can *aperiodically* tile.

Conclusion

- We have algebraic characterisations of PTIME and PSPACE using *robust ODEs*;

Conclusion

- We have algebraic characterisations of PTIME and PSPACE using *robust ODEs*;
- We have studied *robustness in dynamical systems*, to have *Reach* in PSPACE and PTIME;

Conclusion

- We have algebraic characterisations of PTIME and PSPACE using *robust ODEs*;
- We have studied *robustness in dynamical systems*, to have *Reach* in PSPACE and PTIME;
- We extend *robustness in tilings*, to have a computability result.

Conclusion: further work

- Having PSPACE-completeness for the reachability relation (submitted);

Conclusion: further work

- Having PSPACE-completeness for the reachability relation (submitted);
- Adding complexity results in tiling theory.