Math. Struct. in Comp. Science (2007), *vol.* 17, *pp.* 247–259. © 2007 Cambridge University Press
doi:10.1017/S096012950700597X Printed in the United Kingdom

Dynamical systems: stability and simulability

MATHIEU HOYRUP

LIENS, CNRS – ENS, 45 rue d'Ulm, F-75230 Paris cedex 05, France
Email: hoyrup@di.ens.fr

Received 7 April 2006; revised 22 November 2006

Computers are used extensively to simulate continuous dynamical systems. However, different conceptual and mathematical structures underlie discrete machines and continuous dynamics, so the question arises as to the ability of the computer to simulate or, more generally, to check the properties of a continuous system.
We discuss and compare two notions of stability for a continuous dynamical system, *viz.* shadowing and robustness, and relate them to both the practical and theoretical computability of the system. We first discuss what we can learn from the stability of a system, using a finite-precision machine. We then show, following the work in Collins (2005), that shadowing fails but robustness succeeds in ensuring the checkability of a reachability property.

1. Introduction

Computational methods now play a huge role in scientific practice. In view of the place occupied by computer simulations in today's scientific methodology, we need to investigate the role assigned to them for several reasons: so that scientists become more conscious of the orientation induced in their work when using a computer or a computational model; so that computer experiments are meaningful and give informative answers; and, more generally, so that we gain a better understanding of the extent to which a computer can be of help in scientific research.

So, what place does a computer simulation take in the scientific method? Is it just the achievement of a mathematical development, the effective part of a modelling process, the tangible counterpart of a platonistic mathematical model? Or does it rather constitute a specific modelisation process, detached from the abstract mathematical model it is expected to simulate?

The implicit first part of a simulation of a model is the transposition of the model into a specific discrete space, with a change in the nature of its interactions. Indeed, structures (topology, measure, symplectic structures, and so on) that are of great importance for the relevance of the model disappear or are distorted when injected in a computer. Thus, carrying out a computer simulation really is different from putting the initial model into practice; we are, instead, resorting to an autonomous model with its own space and structures.

1.1. *Simulation abilities of computers*

Now let us briefly discuss what computers are good at, and where they may reach their limits.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

The computer has from its beginnings, which may be viewed either as the abstract logical machine invented by Turing in 1936 or its physical realisation around 1950, always been thought of as an arithmetical machine. The discrete internal structure and the exactness of its data-access make computers very efficient at carrying out exact computations on finite entities that can be exactly represented, such as integers, graphs and other sorts of discrete object (within the machine's storage capacity). It is the ideal tool for dealing with discrete mathematics. If the program written is correct, you can run it and be sure that the result will be the correct one: computations are robust and repeatable (they will produce the same results on every computer every time), and everything in a computer is done to obtain absoluteness. We can understand the fact that computer science seems to be integrated within discrete mathematics because of the natural ability of the latter to be handled with algorithms.

All this makes the computer a 'natural' simulator of discrete models, because of the match between the discrete space of the model and the computer data set. But the range of applications of the computer has widened considerably to include all scientific disciplines, going far beyond discrete mathematics, despite the fact that the above-mentioned rationale for using a computer remains confined to the discrete world. Thus, how can we justify the use of a computer to handle a continuous model? Speed and automatic processing are good practical reasons, but there is no general theoretical setting investigating this question. The reliability of simulations is a very tricky problem, as numerous notable studies in numerical analysis prove. This strikingly technical work gives positive results that ensure that discrete resolution methods manage to solve classes of problems on the continuum. They enable us to use computers to produce rigorous solutions to many problems in physics, engineering, biology, and so on.

However, it is, in general, difficult to simulate a continuous system with complete accuracy, and the ability of computers to deal with continuous systems is anything but clear – there is still much to understand at the border between discrete computational models and continuous systems.

1.2. *Stability assures computability*

In the practical world of scientists using finite-precision machines, what makes a system simulable is its stability to rounding errors. But stability can have several aspects, depending on what observations are made of the system. In this paper we will look at two notions of stability that appear in the literature.

The *shadowing* property, which was introduced by Anosov a few decades ago in the theory of dynamical systems, is a global notion of stability: it requires that the complete set of orbits of the system is stable to perturbations.

The *robustness* property (which is stronger than the one proposed in Asarin and Bouajjani (2001) and Collins (2005) in different contexts) is a stronger requirement: it is a local notion of stability, which holds if every orbit is stable to perturbations (see below).

1.3. *Structure of the paper*

Section 2 introduces, compares and discusses the two notions of stability for a dynamical system: the shadowing and robustness properties. Section 3 presents an investigation of

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

the influence of these notions of stability on the algorithmic checkability of key properties. We show through theorems that robustness, which is *a priori* necessary only for practical simulations with finite-precision, is even required for checking some properties of a system with the exact computation model of effective analysis. Section 4 applies the previous notions and results to the logistic maps family. We first illustrate, through the example of the logistic map and the tent map, that shadowing does not prevent distortion of the qualitative behaviour of the system through simulation, due to the internal representation of numbers in a computer, and then we apply results from Section 3. Finally, in Section 5 we take some first steps in a study of the critical logistic map from a computability point of view and show that it is computable (that is, its parameter is computable).

2. Stability of dynamics: shadowing and robustness

2.1. *Shadowing*

Given a discrete time dynamical system $f : X \to X$ where X is a compact metric space, simulating it on a computer is achieved, in practice, by dividing the space X into discrete cells, each of them being represented in the computer by a floating number. Then the function implemented in the machine is an approximation of f, *viz.* a perturbation of f to fit it to the grid. We can formalise this with the following definition.

Definition 1. Given $\delta > 0$, we define a *δ-pseudo-orbit* of f to be a sequence $(x_i)_{i \geqslant 0}$ such that for all i

$$|f(x_i) - x_{i+1}| \leqslant \delta.$$

δ is a majoration of the rounding of the computer, and the result of a simulation is just a δ-pseudo-orbit. It is well known that nonlinear dynamics are unstable with respect to perturbations, and thus not susceptible to approximate computations: in general a pseudo-orbit is unrelated to the real orbit (that is, on the continuum of the real numbers) starting from the same initial condition. Nevertheless, one might expect the computer to give some information about the dynamics of the system simulated. The shadowing property expresses the fact that every pseudo-orbit can be approached (or *shadowed*) by a real orbit.

We say two sequences (x_i), (y_i) are ϵ-close if for every i, we have $|x_i - y_i| \leqslant \epsilon$.

Definition 2. A discrete-time dynamical system $f : X \to X$ has the *shadowing property* if

$$\forall \epsilon > 0, \exists \delta > 0 \text{ such that } \forall \delta\text{-pseudo-orbit}, \exists \text{ a real orbit } \epsilon\text{-close to it.}$$

The shadowing property ensures that the simulation produces plausible behaviours of the system, in fact, orbits that are real orbits up to the precision ϵ. Notice the order in which the statements appear here: for all *δ-pseudo-orbit*, there exists a *real* orbit ϵ-close to it. This is of practical importance: given a discrete grid of step δ (the floating numbers), every effective simulation on this grid is ϵ-approached by a real orbit of the system, but it is not the case, in general, that any real orbit can be approximated by a δ-pseudo-orbit on this grid (for example, uncomputable sequences are not observable). This will be illustrated in Section 4 using the tent map.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

The shadowing property is a *global* stability property, since it requires the whole set of orbits to be stable to perturbation; or, more precisely, given the oriented Hausdorff distance from a set A to a set B as

$$h(A,B) = \sup_{a\in A}\left\{\inf_{b\in B} d(a,b)\right\},$$

the shadowing property requires that for every δ there is an ϵ such that

$$h(\{\delta\text{-perturbed orbits}\},\{\text{real orbits}\}) < \epsilon.$$

2.2. *Robustness: a stronger property*

We will now present a stronger notion of robustness.

Definition 3. Let $f : X \to X$ be a continuous map of a compact metric space X. f is *robust* if

$$\forall\epsilon, \exists\delta \text{ such that } \forall\delta\text{-pseudo-orbits}, \forall i, |x_i - f^i(x_0)| \leqslant \epsilon.$$

Robustness looks like the shadowing property, but *is much stronger*. It requires every pseudo-orbit to stay close to the orbit starting from the same point. It it a *local* stability property in the sense that it expects *every single orbit* to be stable to perturbations. It thus enables practical, effective (using a computer with rounding) predictions of the system (a system with negative Lyapunov exponents is a good example of a robust system).

Collins (2005) defines a notion of robustness based on the reachable set (a similar approach can be found in Asarin and Bouajjani (2001)). This is expressed in topological terms, using open coverings of the space. We will present a metric formulation, which is equivalent on a compact metric space.

Definition 4. Let $f : X \to X$ be a continuous map of a compact metric space X. Given X_0 compact, the *reachable set* is defined as

$$Reach(f, X_0) := \bigcup_{n\geqslant 0} f^n(X_0).$$

Definition 5. Let $f : X \to X$ be a continuous map of a compact metric space X. Given X_0 compact, the *chain-reachable set* is defined as

$$ChainReach(f, X_0) := \bigcap_{\delta>0}\{x|\exists \text{ a } \delta\text{-pseudo-orbit from } X_0 \text{ to } x\}.$$

δ-pseudo-orbits correspond to $\mathscr{V}$-chains as defined in Collins (2005) in a topological context.

Using $\overline{Reach}(f, X_0)$ to denote the topological closure of $Reach(f, X_0)$, it is easy to see that:

— $ChainReach(f, X_0)$ is compact for all X_0 compact.
— $\overline{Reach}(f, X_0) \subseteq ChainReach(f, X_0)$.

Definition 6. A continuous map $f : X \to X$ of a compact metric space X is *reach-robust* iff $\overline{Reach}(f, X_0) = ChainReach(f, X_0)$ for all X_0 compact.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

Proposition 1. Any robust system is reach-robust.

Proof. Let X_0 be a compact subset, $x \in ChainReach(f, X_0)$ and $\epsilon > 0$. We want to show that x is at distance at most ϵ from $Reach(f, X_0)$. As f is robust, some δ is associated to ϵ. Consider a δ-pseudo-orbit $(x_i)_{1 \leqslant i \leqslant n}$ from X_0 to x (which exists, because $x \in ChainReach(f, X_0)$). This is ϵ close to the real orbit starting from the same point:

$$\forall i, \quad d(x_i, f^i(x_0)) \leqslant \epsilon.$$

In particular, $d(x_n, f^n(x_0)) \leqslant \epsilon$. But $x_n = x$ and $f^n(x_0) \in Reach(f, X_0)$. □

The following result is in agreement with the intuition, but is disastrous for the prediction of chaotic systems (the compounding ingredients of chaos are the ones identified in Devaney (1989): the density of the periodic set, topological transitivity and sensitivity to initial conditions).

Proposition 2. Let X be a compact metric space with an infinity of elements. Then a chaotic system $f : X \to X$ is not reach-robust, and so, not robust.

Proof. Take $X_0 = \{p\}$ where p is a periodic point. $ChainReach(f, X_0) = X$ because there is a dense orbit coming near p, but $\overline{Reach}(f, X_0)$ is a finite cycle. □

Note that in most of the scientific literature, the space X is considered to be infinite (as a set).

3. Computability over the real numbers: theoretical simulations

Generally, a discretisation of a system makes it hard to formulate precise statements about that system, simply because of an incompleteness phenomenon: a finite approximation gives only a part of the information on the state of the system, so prediction is impossible in general. Problems appear with non-linear dynamics, as the chaotic logistic map illustrates.

The framework of effective analysis, and, more particularly, type-two effectivity, enables us to represent the system exactly and to carry out exact computations. But handling elements from a continuous space with a discrete computation model (the Turing machine) cannot be achieved without problems. If a real number can be represented exactly, it cannot be known entirely in a finite time.

3.1. *Recursive analysis*

We will concentrate on the real numbers – see Weihrauch (2000) for classical definitions and theory.

Through some encodings, we define the usual *Cauchy representation* of the real numbers $\rho_C :\subseteq \Sigma^\omega \to \mathbb{R}$ were Σ is an alphabet. $\mathrm{dom}(\rho)$ is the set of sequences representing real numbers. A *type-two Turing machine* (TTM) works on infinite sequences on its input(s) tape(s), and stacks symbols on its output tape. If the execution of the machine does not halt, it may produce in the limit an infinite output sequence. If φ is a TTM and $\omega \in \Sigma^\omega$, we use $\varphi(\omega)$ to denote the limit output sequence if it is infinite (otherwise, it is undefined).

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

A function $f : \mathbb{R} \to \mathbb{R}$ is computable if there is a TTM φ such that $f \circ \rho = \rho \circ \varphi$ on $\mathrm{dom}(f \circ \rho)$.

Definition 7. A property P on $\mathbb{R}$ is said to be *semi-decidable* if there is a TTM φ such that for all $\omega \in \mathrm{dom}(\rho)$, $\varphi(\omega)$ halts if and only if $P(\rho(\omega))$

One of the results of effective analysis is given by the following proposition.

Proposition 3. The characteristic set of any semi-decidable property is open for the Euclidian topology.

The intuition is very clear: if φ halts on the representation ω of x, it 'knows' x with only finite precision, so it will also halt on every representation of numbers close to x up to this precision.

Computability on sets has been deeply investigated (see, for example, Brattka and Presser (2003), Brattka and Weihrauch (1999) and Collins (2005)). Three main notions of computability have now emerged: recursivity, recursive enumerability and co-recursive enumerability, which are the generalisations of the classical notions on the integers. We will focus on the latter two.

Definition 8. With I ranging over compact intervals with rational endpoints and J over open intervals with rational endpoints:

— An open set U is r.e. if the set $\{I | I \subseteq U\}$ is recursively enumerable.
— An open set U is co-r.e. if the set $\{J | J \not\subseteq U\}$ is recursively enumerable.
— A closed set A is r.e. if the set $\{J | J \cap A \neq \varnothing\}$ is recursively enumerable.
— A closed set A is co-r.e. if the set $\{I | I \cap A = \varnothing\}$ is recursively enumerable.

Remark. Intervals with rational endpoints can be encoded into integers and recursive enumerability is the classical notion from computability on the integers.

Proposition 4. If U is open, the following are equivalent:

— Membership of U is semi-decidable.
— U is an (open) r.e. set.
— $U^{\complement}$ is a (closed) co-r.e. set.

3.2. *Application to dynamical systems*

The computability of the reachable and chain reachable set are studied in Collins (2005). The following results are specific instances of more general theorems given in a broader context. They will be used in the following section.

Proposition 5. Given a computable function $f : X \to X$ on a compact metric space:

— $\overline{Reach}(f, \{x\})$ is a (closed) r.e. set, effectively in x;
— $ChainReach(f, \{x\})$ is a (closed) co-r.e. set, effectively in x;
— if f is robust, $\overline{Reach}(f, \{x\})$ is a (closed) co-r.e. set, effectively in x;

where *effectively in* x means that x can be passed as an argument, keeping the r.e. or co-r.e. character of the set.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

Theorem 1. Given a *chaotic* computable dynamical system, its non-periodic set (the set of non-periodic points) is not semi-decidable.

Proof. We show that the non-periodic set is not open. It is not empty because of the existence of a dense orbit. And it contains no open interval because of the density of the periodic set. □

We will see in Section 4 that for a robust system the non-periodic set may be semi-decidable, unlike the case for a chaotic system.

3.3. *Reachability*

Proposition 6. Given a computable dynamical system (X, f), its *reaching* relation

$$R(x, U) :\iff \exists n, f^n(x) \in U$$

is semi-decidable (as a relation over $\mathbb{R} \times \{\text{rational open intervals}\}$).

Remark. Some arguments of the relation are rational numbers: they are encoded into integers in the usual way, and placed on an extra input tape of a TTM, which treats them in the same way as a classical Turing machine.

The algorithm is simple: compute the iterates of x and test (using a dove-tail scheme) whether they belong to U (a semi-decidable test).

This reaching relation can be expressed another way:

$$R(x, U) \iff Reach(f, \{x\}) \cap U \neq \varnothing \iff \overline{Reach}(f, \{x\}) \cap U \neq \varnothing.$$

This is semi-decidable because $\overline{Reach}(f, \{x\})$ is r.e., effectively in x (this result is mentioned in Collins (2005)).

We will now introduce a notion that has a clear intuitive meaning and is slighty different from the complement of the reachability relation. Moreover, it effectively discriminates between chaos and robustness.

Definition 9. Given any computable dynamical system (X, f), we define the *keep away* relation:

$$\begin{aligned} KA(x, I) &:\iff \text{the orbit of } x \text{ keeps away from } I \\ &\iff \inf_{n \in \mathbb{N}} d(f^n(x), I) > 0 \end{aligned}$$

where x ranges over $\mathbb{R}$ and I over rational compact intervals.

Theorem 2. Given a *chaotic* computable dynamical system, its *keep away* relation is not semi-decidable.

Proof. For each periodic point p there is I with non-empty interior at non-zero distance from the orbit of p. But I is reached by any dense orbit starting arbitrary close to p, so there is no open set containing p on which the property holds. So the characteristic set is not open. □

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

This property can be expressed another way:

$$KA(x, I) \iff \overline{Reach}(f, \{x\}) \cap I = \varnothing.$$

As a result, we have the following theorem.

Theorem 3. Given a *robust* computable system, its *keep away* relation is semi-decidable.

Proof. Indeed, for a robust computable system, $\overline{Reach}(f, \{x\})$ is a co-r.e. set, effectively in x (put in this way, the result can be deduced from a result in Collins (2005)). □

3.4. *Interpretation*

One can see recursive analysis as a framework formalising what computations can be achieved in a situation located at the border between practice and theory: only finite approximations are handled by the computer, but with arbitrary precision. There is no limitation on the precision used, but in finite time only finite approximations are available.

So, through a representation such as the one used here, a real number can never be known entirely, contrary to an enumeration of the rational numbers, which enables a complete knowledge of a rational number in a finite amount of time. For example, having access to the number $3/5$ through an infinite sequence converging effectively to $3/5$ is weaker than knowing 'that this number is $3/5$'. What one could know, by a mathematical proof, about the orbit of $3/5$ may be of no help when the input is only a *representation* of $3/5$: there is no way of knowing that it is $3/5$ that is represented. Formally, equality to $3/5$ is not semi-decidable.

In this sense, only 'robust' properties can be algorithmically checked, that is, properties that can be checked with arbitrary but finite precision. Topologically, robustness implies openness of the characteristic set. As shown by the last two theorems, the robustness of a property can depend on the robustness of the system.

4. An application to the logistic maps family

We will focus, as an illustration of a dynamical system, on the well-known family of logistic maps, which exhibits a widespread range of dynamical features.

Definition 10. The logistic maps family is defined by

$$\begin{aligned} F_\lambda : [0,1] &\to [0,1] \\ x &\mapsto \lambda x(1-x) \end{aligned}$$

where λ is a parameter in $[0, 4]$.

If λ is less than the threshold $\lambda_\infty \approx 3.5768\ldots$ (see Section 5), every orbit of the system is attracted to a periodic cycle, and the system is robust. If $\lambda = 4$, the system is chaotic.

Theorem 4. The chaotic logistic map F_4 has the shadowing property.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

This is a consequence of the following lemmas and facts:

— The tent map

$$\begin{aligned} T : [0,1] &\to [0,1] \\ x &\mapsto 2x \text{ if } x \leqslant 1/2 \\ &\quad\; 2(1-x) \text{ if } x \geqslant 1/2 \end{aligned}$$

has the shadowing property (see Alligood and Sauer (1996) for a sketch of the proof, or Chow and Palmer (1991)).

— F_4 and T are topologically conjugate, in other words, they have the 'same behaviour' up to a homeomorphic renaming of the state (see Devaney (1989)).

— The shadowing property is preserved through topological conjugacy on compact metric spaces.

Now let us run simulations of the tent map and the logistic map on a conventional computer. These topologically conjugate maps should show close behaviours. But every simulation of the tent map quickly stabilises on 0 whereas simulations of the logistic map seem erratic. This is due to the internal representation of numbers, more precisely to the specificity of the floating numbers among real numbers. Indeed, the grid formed by these floating points is not neutral and interferes with the dynamics of the system.

Proposition 7. The set of points of the tent map reaching 0 coincides with the set of dyadic numbers.

Proof. The binary expansion of a dyadic number is finite and becomes strictly shorter by application of the tent map.

For the opposite direction, the reverse image of a dyadic number x is a pair of dyadic numbers ($x/2$ and $1 - x/2$). As 0 is a dyadic number, the result follows. □

Therefore, the grid of floating numbers is contained in the set of points reaching 0.

Furthermore, the simulation of the tent map is *exact*: multiplications by 2 are achieved without rounding, simply by shifts of bit strings.

To sum up, on the one hand (the chaotic logistic map), rounding errors are introduced in computations, but the sequences observed *seem* common, representative. On the other hand (the tent map), the computer carries out exact computations but shows particular orbits, which are unrepresentative of the general behaviour of the system. So the improvement in exactness comes together with a strong restriction on the observable behaviours.

This example shows that the shadowing property is not sufficient to assure the reliability of the observations on a screen: the pseudo-orbits are plausible, but may not be representative of the generic behaviour of the system. This is a major, and typical, problem that one encounters with discrete state machines.

4.1. *Semi-decidability of two properties*

We will now give a a simple illustration, on the logistic maps, of the effect of the stability of the dynamics on the semi-decidability of some properties.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

4.1.1. *Non-periodic set*

Theorem 5. Let $\lambda < \lambda_\infty$ be a computable real number. For the robust logistic map F_λ, non-periodicity is semi-decidable.

Proof. Indeed, non-periodicity is equivalent to being in the complement of a finite set of computable points. □

Theorem 1 implies that for the chaotic logistic map F_4, non-periodicity is not semi-decidable.

4.1.2. *Keep-away relation* For $\lambda < \lambda_\infty$ computable, the keep-away relation of F_λ is semi-decidable because of the robustness of the function (Theorem 3).

But the keep-away relation of the chaotic map F_4 is not semi-decidable, according to Theorem 2.

5. Computability of λ_∞

In the previous section we studied links between computability and dynamics for two of the three possible behaviours of the logistic map: the stable case ($\lambda < \lambda_\infty$) and the chaotic case ($\lambda = 4$). We will not study the critical case ($\lambda = \lambda_\infty$) in full here, but will overcome the first hurdle required for this study, which is the computability of the critical logistic map itself.

The aim of this section is to show that λ_∞ is a computable real number.

5.1. *Renormalisation of a dome-shaped map*

Definition 11. The Schwarzian derivative of a C^3 map $f : I \to I$ is defined as

$$Sf(x) = \frac{f'''(x)}{f'(x)} - \frac{3}{2}\left(\frac{f''(x)}{f'(x)}\right)^2.$$

Definition 12. A C^3 map $f : I \to I$ on a compact interval I will be said to have a *dome shape* if it is unimodal, has negative Schwarzian derivative, and satisfies $f'(0) > 1$.

This kind of map has interesting regularity properties (for instance, it has at most 2 fixed points). Details can be found in Devaney (1989).

Let f have a dome shape. Then f has two fixed points 0 and $p > 0$. Furthermore, if $p > c$ (c is the critical point), p has an antecedent $\hat{p} < c$. A nice and simple observation is that if $[\hat{p}, p]$ is invariant under f^2 (which is equivalent to $f^2(c) \geqslant \hat{p}$), and if $f'(p) < -1$, then the map $f^2 : [\hat{p}, p] \to [\hat{p}, p]$ also has a dome shape.

Definition 13. A dome-shaped map f is renormalisable if its second fixed point p satisfies $f'(p) < -1$ and $f^2(c) \geqslant \hat{p}$.

Proposition 8. A dome-shaped map f is renormalisable if and only if $f(c) > c$ and $f^4(c) \leqslant f^3(c)$.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

Remark. This characterisation is interesting because it does not use p.

Proof. it is easy to see that f has a fixed point $p > c$ if and only if $f(c) > c$. Now, if this is the case, as $f([0,\hat{p})) = [0,p)$ and $f([\hat{p},p)) = [p,f(c)]$, and $f^2(c) < p$, we have

$$f^2(c) \geqslant \hat{p} \iff f^2(c) \in [\hat{p},p) \iff f^3(c) \geqslant p \iff f^4(c) \leqslant f^3(c). \qquad \square$$

Given a renormalisable map f, we can investigate f^2 on $[\hat{p},p]$ by *renormalising* it, that is, stretching this interval to $[0,1]$. So, we define L_f to be the linear decreasing map sending p to 0 and $\hat{p}$ to 1.

Definition 14. The *renormalisation* of a renormalisable map f is the map defined by

$$Rf := L_f \circ f^2 \circ L_f^{-1}.$$

Rf has a dome shape. It may then be renormalisable, as f was. If this is the case, $R^2 f$ denotes the second renormalisation of f. And this may be iterated. We call the greatest integer n such that $R^n f$ exists the degree of renormalisability of f.

The logistic map f_λ has a dome shape for every parameter λ. Its degree of renormalisability varies with λ. It can be proved that there is a nested sequence of intervals $[\lambda_i, \mu_i]$ with $1 < \lambda_1 < \lambda_2 < \ldots < \mu_2 < \mu_1 < 4$ such that f_λ has degree i for $\lambda \in [\lambda_i, \mu_i]$. The sequences (λ_i) and (μ_i) are adjacent and converge to a critical parameter λ_∞, and f_{λ_∞} is infinitely renormalisable.

5.2. *Computability of Rf*

We will now look at the computable counterpart of this.

Proposition 9. Given a renormalisable computable dome-shaped map $f : I \to I$, its renormalisation Rf is computable, effectively in f.

This is a corollary of the following two results.

Lemma 1. Given a computable function $f : [a,b] \to [a,b]$ (where a and b are computable real numbers) with only one zero z, z is computable, effectively in f.

Remark. It is well known in computability theory that z is computable (see Pour-El and Richards (1989), for instance), but what is important here is to compute it effectively from f.

Proof. A simple dichotomy method may fail to compute z: if we cut the working interval at its middle point m, the tests $f(m) < 0$ and $f(m) > 0$ can diverge if $m = z$. So we carry out a trichotomy. From $a_0 = a, b_0 = b$, given a_n, b_n, we test in parallel the signs of $f(\frac{2a_n+b_n}{3})$ and $f(\frac{a_n+2b_n}{3})$. As at least one of them is not null (there is only one zero), one test stops, and we define a_{n+1} and b_{n+1} from this. We obtain $a_n < z < b_n$ and $b_n - a_n \leqslant \left(\frac{2}{3}\right)^n (b_0 - a_0)$, which proves that z can be computed from f. $\square$

Corollary 1. Given a dome-shaped map f for which the second fixed point p is greater than c, we have p and $\hat{p}$ are computable effectively in f.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

Proof. p is the unique zero of $f - id$ on $[c, 1]$ and $\hat{p}$ the unique zero of $f - p$ on $[0, c]$. □

5.3. *Finite degree of renormalisability*

Lemma 2. Knowing if a dome-shaped map has a finite degree of renomalisability is semi-decidable relative to an oracle furnishing an upper bound on $|f'|$.

Remark. One may wonder why the oracle is required. A computable C^1 map does not, in general, have a computable derivative. However, this is the case for a computable C^2 map, but not effectively in f: in general, f' cannot be computed from f. On the other hand, if we know an upper bound for $|f'|$ (which exists since f' is continuous on I compact), f' can be computed.

Furthermore, if f' is computable, the derivative of every $R^n f$ is computable, effectively in f. Indeed, $(Rf)' = L_f \circ f'.f' \circ f$, so $(Rf)'$ is computable from f and f'. Iterating the argument, $(R^{n+1}f)'$ is computable from $R^n f$ and $(R^n f)'$.

Proof. A useful remark is that $\max(c, p)$ is computable effectively in f. Indeed, the trichotomy algorithm computing p actually computes c if $p < c$.

We define a test $T(f) := (f(c) < c) \parallel (f^4(c) > f^3(c)) \parallel (|f'(\max(c, p))| < 1)$. If f is not renormalisable, then $f(c) < c$ or $f(c) = c$ or $f^4(c) > f^3(c)$. If $f(c) = c$, then $p = c$ and $f'(\max(c, p)) = 0$. Then the test stops.

If one of the first two tests stops first, f is not renormalisable. If the third test stops, then f may be either not renormalisable or renormalisable. In the latter case, $\max(c, p) = p$, so $(Rf)'(0) = (f'(p))^2 < 1$, and Rf is not renormalisable (it has only one fixed point 0).

Now run $T(f), T(Rf), T(R^2 f), \ldots$ in parallel (with interleaving) – if $R^i f$ is not renormalisable, the program intended to compute $R^{i+1}f$ does not, in fact, terminate, as $T(R^{i+1}f)$. If f is n times renormalisable and not $n + 1$ times, $T(R^n f)$ will stop. For the opposite direction, if $T(R^n f)$ for some n, $R^n f$ or $R^{n+1}f$ is not renormalisable.

Then the whole test stops if and only if f is only finitely renormalisable. □

Theorem 6. The relations $\lambda < \lambda_\infty$ and $\lambda > \lambda_\infty$ are semi-decidable. Hence λ_∞ is a computable real number.

Proof. For every $\lambda \in (1, 4]$, f_λ has a bell-shape and $|f'_\lambda| < \lambda \leqslant 4$, so finite renormalisability of f_λ, which is equivalent to $\lambda \neq \lambda_\infty$, is semi-decidable in λ. When a test $T(R^n f_\lambda)$ stops, if $R^n f_\lambda(c) < c$ or $|f'(\max(c, p))| < 1$, then $\lambda < \lambda_\infty$, and if $f^4(c) > f^3(c)$, then $\lambda > \lambda_\infty$. So $\lambda < \lambda_\infty$ and $\lambda > \lambda_\infty$ can be semi-decided. □

6. Conclusion

The shadowing property is actually a *global* stability property: stability of the whole set of orbits to perturbations of the system.

The robustness property is a *local* stability property: it requires the stability of every single orbit to perturbations of the system.

Robustness enables accurate computation of every orbit with a finite-precision, with rounding. In this sense, one might think that the use of effective analysis makes the

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press

robustness requirement too strong and not necessary, because exact computations are allowed through representations. But the theorems presented in this paper show that, when considering only chaotic or robust systems, robustness is also necessary in this framework in order to make key properties on dynamical systems semi-decidable.

So, we cannot hope to get predictive results, with approximate as well as exact computations, for maps that are non-robust. The fact that robustness is a very strong property means that this is very bad news. We have seen that the shadowing property was a weaker condition allowing a map to be handled with a computer. But questions such as the representativity of the (pseudo-)orbits observed are raised (an orbit could be said to be representative if it 'satisfies the ergodic theorem', for instance).

The issues we have hinted at here are fundamental to the relationship between the discrete universe of effective computations and continuous models. Effective notions of randomness may further help to fill the gap, as randomness is nicely expressed both by chaotic continuous dynamics and computability theory, *à la* Martin-Löf, typically (see Rojas (2006) for a recent survey and further indications of the direction of our work).

References

Alligood, K. T., Sauer, T. D. and Yorke, J. A (1996) *Chaos: An Introduction to Dynamical Systems*, Springer.

Asarin, E. and Bouajjani, A (2001) Perturbed Turing Machines and Hybrid Systems. *LICS'2001*.

Brattka, V. and Presser, G (2003) Computability on subsets of metric spaces. *Theoretical Computer Science* **305** 43–76.

Brattka, V. and Weihrauch, K (1999) Computability on subsets of Euclidean space I: Closed and compact subsets. *Theoretical Computer Science* **219** 65–93.

Chow, S. N. and Palmer, K. J (1991) On the numerical computation of orbits of dynamical systems: the one-dimensional case. *Dynamics and Differential Equations* **3** 361–380.

Collins, P (2005) Continuity and computability of reachable sets. *Theoretical Computer Science* **341**.

Devaney, R (1989) *An Introduction to Chaotic Dynamical Systems*, Perseus Publishing Co. (HarperCollins).

Pour-El, M. and Richards, I (1989) *Computability in Analysis and Physics*, Springer-Verlag.

Rojas, C (2006) Computability and Information in Models of Randomness and Chaos (to appear).

Weihrauch, K (2000) *Computable Analysis, An Introduction*, Springer-Verlag.

https://doi.org/10.1017/S096012950700597X Published online by Cambridge University Press