Logical Methods in Computer Science
Volume 16, Issue 1, 2020, pp. 28:1–28:37
https://lmcs.episciences.org/

Submitted Apr. 11, 2018
Published Feb. 28, 2020

A UNIVERSAL ORDINARY DIFFERENTIAL EQUATION

OLIVIER BOURNEZ [a] AND AMAURY POULY [b]

[a] Ecole Polytechnique, LIX, 91128 Palaiseau Cedex, France
e-mail address: bournez@lix.polytechnique.fr

[b] Max Planck Institute for Software Systems (MPI-SWS), Germany

Université de Paris, IRIF, CNRS
e-mail address: amaury.pouly@irif.fr

Abstract. An astonishing fact was established by Lee A. Rubel (1981): there exists a fixed non-trivial fourth-order polynomial differential algebraic equation (DAE) such that for any positive continuous function $\varphi(t)$ on the reals, and for any positive continuous function $\epsilon(t)$, it has a $\mathcal{C}^\infty$ solution with $|y(t)-\varphi(t)| < \epsilon(t)$ for all t. Lee A. Rubel provided an explicit example of such a polynomial DAE. Other examples of universal DAE have later been proposed by other authors. However, Rubel's DAE *never* has a unique solution, even with a finite number of conditions of the form $y^{(k_i)}(a_i) = b_i$.

The question whether one can require the solution that approximates $\varphi(t)$ to be the unique solution for a given initial data is a well known open problem [Rubel 1981, page 2], [Boshernitzan 1986, Conjecture 6.2]. In this article, we solve it and show that Rubel's statement holds for polynomial ordinary differential equations (ODEs), and since polynomial ODEs have a unique solution given an initial data, this positively answers Rubel's open problem. More precisely, we show that there exists a **fixed** polynomial ODE such that for any $\varphi(t)$ and $\epsilon(t)$ there exists some initial condition that yields a solution that is $\epsilon(t)$-close to $\varphi(t)$ at all times. In particular, the solution to the ODE is necessarily analytic, and we show that the initial condition is computable from the target function and error function.

1. Introduction

An astonishing result was established by Lee A. Rubel in 1981 [Rub81]. There exists a universal fourth-order algebraic differential equation in the following sense.

Theorem 1.1 [Rub81]**.** *There exists a non-trivial fourth-order implicit differential algebraic equation*

$$P(y', y'', y''', y'''') = 0 \tag{1.1}$$

Key words and phrases: Ordinary Differential Equations, Universal Differential Equations, Analog Models of Computation, Continuous-Time Models of Computation, Computability, Computational Analysis, Computational Complexity .

Olivier Bournez was partially supported by *ANR PROJECT RACAF*.

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.23638/LMCS-16(1:28)2020

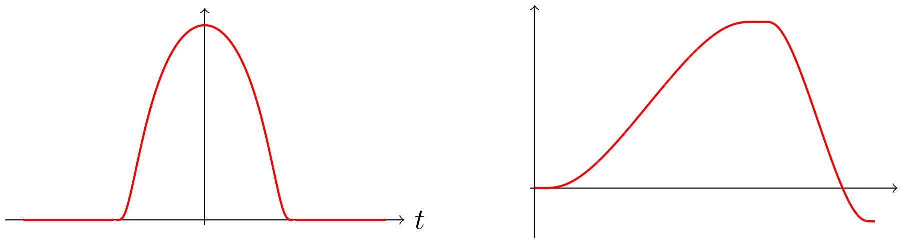

Figure 1: On left, graphical representation of function g. On right, two S-modules glued together.

where P is a polynomial in four variables with integer coefficients, such that for any continuous function φ on $(-\infty,\infty)$ and for any positive continuous function $\epsilon(t)$ on $(-\infty,\infty)$, there exists a $\mathcal{C}^\infty$ solution y to (1.1) *such that*

$$|y(t)-\varphi(t)|<\epsilon(t)$$

for all $t\in(-\infty,\infty)$.

Even more surprising is the fact that Rubel provided an explicit example of such a polynomial P that is particularly simple:

$$\begin{array}{ll} 3y'^4y''y''''^2 & -4y'^4y'''^2y'''' + 6y'^3y''^2y'''y'''' + 24y'^2y''^4y'''' \\ & -12y'^3y''y'''^3 - 29y'^2y''^3y'''^2 + 12y''^7 \qquad\qquad = 0. \end{array} \tag{1.2}$$

While this result looks very surprising at first sight, Rubel's proofs turns out to use basic arguments, and can be explained as follows. It uses the following classical trick to build $\mathcal{C}^\infty$ piecewise functions: let

$$g(t)=\begin{cases} e^{-1/(1-t^2)} & \text{for } -1<t<1 \\ 0 & \text{otherwise.}\end{cases}$$

It is not hard to see that function g is C^∞ and Figure 1 shows that g looks like a "bump". Since it satisfies

$$\tfrac{g'(t)}{g(t)} = -\frac{2t}{(1-t^2)^2},$$

then

$$g'(t)(1-t^2)^2+g(t)2t=0$$

and $f(t)=\int_0^t g(u)du$ satisfies the polynomial differential algebraic equation

$$f''(1-t^2)^2+f'(t)2t=0.$$

Since this equation is homogeneous, it also holds for $af+b$ for any a and b. The idea is then to obtain a fourth order DAE that is satisfied by every function $y(t)=\gamma f(\alpha t+\beta)+\delta$, for all $\alpha,\beta,\gamma,\delta$. After some computations, Rubel obtained the universal differential equation (1.2).

Functions of the type $y(t)=\gamma f(\alpha t+\beta)+\delta$ generate what Rubel calls S-modules: a function that values A at a, B at b, is constant on $[a,a+\delta]$, monotone on $[a+\delta,b-\delta]$, constant on $[b-\delta,b]$, by an appropriate choice of $\alpha,\beta,\gamma,\delta$. Summing S-modules corresponds to gluing then together, as is depicted in Figure 1. Note that finite, as well as infinite sums[1] of S-modules still satisfy the equation (1.2) and thus any piecewise affine function (and

[1]With some convergence or disjoint domain conditions.

hence any continuous function) can be approximated by an appropriate sum of S-modules. This concludes Rubel's proof of universality.

As one can see, the proof turns out to be frustrating because the equation essentially allows any behavior. This may be interpreted as merely stating that differential algebraic equations is simply too lose a model. Clearly, a key point is that this differential equation does not have a unique solution for any given initial condition: this is the core principle used to glue a finite or infinite number of S-modules and to approximate any continuous function. Rubel was aware of this issue and left open the following question in [Rub81, page 2].

> "It is open whether we can require in our theorem that the solution that approximates φ to be the unique solution for its initial data."

Similarly, the following is conjectured in [Bos86, Conjecture 6.2].

> "Conjecture. There exists a non-trivial differential algebraic equation such that any real continuous function on $\mathbb{R}$ can be uniformly approximated on all of $\mathbb{R}$ by its real-analytic solutions"

The purpose of this paper is to provide a positive answer to both questions. We prove that a fixed polynomial ordinary differential equations (ODE) is universal in above Rubel's sense. At a high level, our proofs are based on ordinary differential equation programming. This programming is inspired by constructions from our previous paper [BGP16a]. Here, we mostly use this programming technology to achieve a very different goal and to provide positive answers to these above open problems.

We also believe they open some lights on computability theory for continuous-time models of computations. In particular, it follows that concepts similar to Kolmogorov complexity can probably be expressed naturally by measuring the complexity of the initial data of a (universal-) polynomial ordinary differential equation for a given function. We leave this direction for future work.

The current article is an extended version of [BP17]: here all proofs are provided, and we extend the statements by proving that the initial condition can always be computed from the function in the sense of Computable Analysis.

1.1. **Related work and discussions.** First, let us mention that Rubel's universal differential equation has been extended in several papers. In particular, Duffin proved in [Duf81] that implicit universal differential equations with simpler expressions exists, such as

$$n^2 y^{''''} {y'}^2 + 3n(1-n)y^{'''}y^{''}y' + (2n^2-3n+1){y^{''}}^3 = 0$$

for any $n > 3$. The idea of [Duf81] is basically to replace the $\mathcal{C}^\infty$ function g of [Rub81] by some piecewise polynomial of fixed degree, that is to say by splines. Duffin also proves that considering trigonometric polynomials for function $g(x)$ leads to the universal differential equation

$$ny^{''''}{y'}^2 + (2-3n)y^{'''}y^{''}y' + 2(n-1){y''}^3 = 0.$$

This is done at the price of approximating function φ respectively by splines or trigonometric splines solutions which are $\mathcal{C}^n$ (and n can be taken arbitrary big) but not $\mathcal{C}^\infty$ as in [Rub81]. Article [Bri02] proposes another universal differential equation whose construction is based on Jacobian elliptic functions. Notice that [Bri02] is also correcting some statements of [Duf81].

All the results mentioned so far are concerned with approximations of continuous functions over the whole real line. Approximating functions over a compact domain seems to be a different (and somewhat easier for our concerns) problem, since basically by compactness, one just needs to approximate the function locally on a finite number of intervals. A 1986 reference survey discussing both approximation over the real line and over compacts is [Bos86]. Recently, over compact domains, the existence of universal ordinary differential equation $\mathcal{C}^\infty$ of order 3 has been established in [CJ16]: it is shown that for any $a < b$, there exists a third order $\mathcal{C}^\infty$ differential equation $y''' = F(y, y', y'')$ whose solutions are dense in $\mathcal{C}^0([a,b])$. Notice that this is not obtained by explicitly stating such an order 3 universal ordinary differential, and that this is a weaker notion of universality as solutions are only assumed to be arbitrary close over a compact domain and not all the real line. Order 3 is argued to be a lower bound for Lipschitz universal ODEs [CJ16].

Rubel's result has sometimes been considered to be the equivalent, for analog computers, of the universal Turing machines. This includes Rubel's paper motivation given in [Rub81, page 1]. We now discuss and challenge this statement.

Indeed, differential algebraic equations are known to be related to the General Purpose Analog Computer (GPAC) of Claude Shannon [Sha41], proposed as a model of the Differential Analysers [Bus31], a mechanical programmable machine, on which he worked as an operator. Notice that the original relations stated by Shannon in [Sha41] between differential algebraic equations and GPACs have some flaws, that have been corrected later by [PE74] and [GC03]. Using the better defined model of GPAC of [GC03], it can be shown that functions generated by GPAC exactly correspond to polynomial ordinary differential equations. Some recent results have established that this model, and hence polynomial ordinary differential equations can be related to classical computability [BCGH07] and complexity theory [BGP16a].

However, we do not really agree with the statement that Rubel's result is the equivalent, for analog computers, of the universal Turing machines. In particular, Rubel's notion of universality is completely different from those in computability theory. For a given initial data, a (deterministic) Turing machine has only one possible evolution. On the other hand, Rubel's equation does not dictate any evolution but rather some conditions that any evolution has to satisfy. In other words, Rubel's equation can be interpreted as the equivalent of an invariant of the dynamics of (Turing) machines, rather than a universal machine in the sense of classical computability.

Notice that while several results have established that (polynomial) ODEs are able to simulate the evolution of Turing machines (see e.g. [BCGH07, GCB08, BGP16a]), the existence of a universal ordinary differential equation does not follow from them. To understand the difference, let us restate the main result of [GCB08], of which [BGP16a] is a more advanced version for polynomial-time computable functions.

Theorem 1.2. *A function $f : [a,b] \to \mathbb{R}$ is computable (in the framework of Computable Analysis) if and only if there exists some polynomials $p : \mathbb{R}^n \to \mathbb{R}^n$, $p_0 : \mathbb{R} \to \mathbb{R}$ with computable coefficients and $\alpha_1, \dots, \alpha_{n-1}$ computable reals such that for all $x \in [a,b]$, the solution $y : \mathbb{R}_{\geqslant 0} \to \mathbb{R}^n$ to the Cauchy problem*

$$y(0) = (\alpha_1, \dots, \alpha_{n-1}, p_0(x)), \qquad y' = p(y)$$

satisfies that for all $t \geqslant 0$ that

$$|f(x) - y_1(t)| \leqslant y_2(t) \quad \text{and} \quad \lim_{t\to\infty} y_2(t) = 0.$$

Since there exists a universal Turing machine, there exists a "universal" polynomial ODE for computable functions. But there are major differences between Theorem 1.2 and the result of this paper (Theorem 1.3). Even if we have a strong link between the Turing machines's configuration and the evolution of the differential equation, this is not enough to guarantee what the trajectory of the system will be at all times. Indeed, Theorem 1.2 only guarantees that $y_1(t) \to f(x)$ asymptotically. On the other hand, Theorem 1.3 guarantees the value of $y_1(t)$ at all times. Notice that our universality result also applies to functions that are not computable (in which case the initial condition is computable from the function but still not computable).

We would like to mention some implications for experimental sciences that are related to the classical use of ODEs in such contexts. Of course, we know that this part is less formal from a mathematical point of view, but we believe this discussion has some importance: A key property in experimental sciences, in particular physics is analyticity. Recall that a function is analytic if it is equal to its Taylor expansion in any point. It has sometimes been observed that "natural" functions coming from Nature are analytic, even if this cannot be a formal statement, but more an observation (see e.g. [BC08, Moo90, KM99]). We obtain a fixed universal polynomial ODE, so in particular all its solution must be analytic[2], and it follows that universality holds even with analytic functions. All previous constructions mostly worked by gluing together $\mathcal{C}^\infty$ or $\mathcal{C}^n$ functions, and as it is well known "gluing" of analytic functions is impossible. We believe this is an important difference with previous works.

As we said, Rubel's proof can be seen as an indication that (fourth-order) polynomial implicit DAE is too loose model compared to classical ODEs, allowing in particular to glue solutions together to get new solutions. As observed in many articles citing Rubel's paper, this class appears so general that from an experimental point of view, it makes littles sense to try to fit a differential model because a single equation can model everything with arbitrary precision. Our result implies the same for polynomial ODEs since, for the same reason, a single equation of sufficient dimension can model everything.

Notice that our constructions have at the end some similarities with Voronin's theorem. This theorem states that Riemann's ζ function is such that for any analytic function $f(z)$ that is non-vanishing on a domain U homeomorphic to a closed disk, and any $\epsilon > 0$, one can find some real value t such that for all $z \in U$, $|\zeta(z + it) - f(z)| < \epsilon$. Notice that ζ function is a well-known function known not to be solution of any polynomial DAE (and consequently polynomial ODE), and hence there is no clear connection to our constructions based on ODEs. We invite to read the post [LR] in "Gödel's Lost Letter and P=NP" blog for discussions about potential implications of this surprising result to computability theory.

1.2. **Formal statements.** Our results are the following:

Theorem 1.3 (Universal PIVP). *There exists a **fixed** polynomial vector p in d variables with rational coefficients such that for any functions $f \in C^0(\mathbb{R})$ and $\varepsilon \in C^0(\mathbb{R}, \mathbb{R}_{>0})$, there exists $\alpha \in \mathbb{R}^d$ such that there exists a unique solution $y : \mathbb{R} \to \mathbb{R}^d$ to $y(0) = \alpha$, $y' = p(y)$. Furthermore, this solution satisfies that $|y_1(t) - f(t)| \leqslant \varepsilon(t)$ for all $t \in \mathbb{R}$, and it is analytic.*

Furthermore, α can be computed from f and ε in the sense of Computable Analysis, more precisely $(f, \varepsilon) \mapsto \alpha$ is $([\rho \to \rho]^2, \rho^d)$-computable (refer to Section 2.3 for formal definitions).

[2]Which is not the case for polynomial DAEs.

It is well-known that polynomial ODEs can be transformed into DAEs that have the same analytic solutions, see [CPSW05] for example. The following then follows for DAEs.

Theorem 1.4 (Universal DAE). *There exists a **fixed** polynomial p in $d+1$ variables with rational coefficients such that for any functions $f \in C^0(\mathbb{R})$ and $\varepsilon \in C^0(\mathbb{R}, \mathbb{R}_{>0})$, there exists $\alpha_0, \dots, \alpha_{d-1} \in \mathbb{R}$ such that there exists a unique **analytic** solution $y : \mathbb{R} \to \mathbb{R}$ to $y(0) = \alpha_0, y'(0) = \alpha_1, \dots, y^{(d-1)}(0) = \alpha_{d-1}$, $p(y, y', \dots, y^d) = 0$. Furthermore, this solution satisfies that $|y(t) - f(t)| \leqslant \varepsilon(t)$ for all $t \in \mathbb{R}$.*

Furthermore, α can be computed from f and ε in the sense of Computable Analysis, more precisely $(f, \varepsilon) \mapsto \alpha$ is $([\rho \to \rho]^2, \rho^d)$-computable (refer to Section 2.3 for formal definitions).

Remark 1.5. Notice that both theorems apply even when f is not computable. In this case, the initial condition(s) α exist but are not computable. We will prove that α is always *computable from* f and ε, that is the mapping $(f, \varepsilon) \mapsto \alpha$ is computable in the framework of Computable Analysis, with an adequate representation of f, ε and α.

Remark 1.6. Notice that we do not provide explicitly in this paper the considered polynomial ODE, nor its dimension d. But it can be derived by following the constructions. We currently estimate d to be more than three hundred following the precise constructions of this paper (but also to be very far from the optimal). We did not try to minimize d in the current paper, as we think our results are sufficiently hard to be followed in this paper for not being complicated by considerations about optimization of dimensions.

Remark 1.7. Both theorems are stated for *total functions* f and ε over $\mathbb{R}$. It trivially applies to any continuous partial function that can be extended to a continuous function over $\mathbb{R}$. In particular, it applies to any functions over $[a, b]$. It is not hard to see that it also applies to functions over (a, b) by rescaling $\mathbb{R}$ into (a, b) using the cotangent:

$$z(t) = y\left(-\cot\left(\tfrac{t-a}{b-a}\pi\right)\right) \quad \text{satisfies} \quad z'(t) = \phi'(t)p(z(t)), \quad \phi'(t) = \tfrac{\pi}{b-a}(1 + \phi(t)^2).$$

More complex domains such as $[a, b)$ and $(a, b]$ (with a possibly infinite) can also be obtained using a similar method.

Remark 1.8. Since the solution of a polynomial (or analytic) differential equation is analytic, our results can be compared with the problem of building uniform approximations of continuous function on the real line by analytic ones, and hence can be seen as a strengthening of such results (see e.g. [Kap55]).

Remark 1.9. Let $Y(\alpha)$ be the solution given by Theorem 1.3 satisfying $Y(\alpha)(0) = \alpha$. Note that the theorem does *not* specify the existence of $Y(\alpha)(t)$ for all t and α. In fact, because of function `fastgen` in what follows, $Y(\alpha)$ will explode in finite time for all α that have certain coordinates rational, and the length of the interval of life depends on α. Therefore, given $\alpha \in \mathbb{R}^d$, any ball around α contains a β such that $Y(\beta)$ explodes in finite time for the function Y corresponding to our constructions.

Remark 1.10. It may look at first like that Theorem 1.3 violates Brouwer's Invariance of domain but this is not the case. Indeed, continuing with the notation of above remark, Y is continuous[3] and Y is injective[4] with image in $S := \bigcup_{a<b} C^0((a, b), \mathbb{R}^d)$ (see Remark 1.9

[3] This is the local continuity of the solution to a smooth differential equation with respect to the initial condition.

[4] This is the fact that for an autonomous ODE, two trajectories are either disjoint or the same.

about domains). Clearly S is of much higher dimension than $\alpha \in \mathbb{R}^d$ but Y is not dense in S so there is no contradiction. On the other hand, if we only consider the first coordinate Y_1, then Y_1 is dense in $C^0(\mathbb{R}, \mathbb{R})$ but is not injective.

1.3. **Overview of the proof.** A first a priori difficulty is that if one considers a fixed polynomial ODE $y' = p(y)$, one could think that the growth of its solutions is constrained by p and thus cannot be arbitrary. This would then prevent us from building a universal ODE simply because it could not grow fast enough. This fact is related to Emil Borel's conjecture in [Bor99] (see also [Har12]) that a solution, *defined over* $\mathbb{R}$, to a system with n variables has growth bounded by roughly $e_n(x)$, the $n-$th iterate of exp. The conjecture is proved for $n = 1$ [Bor99], but has been proven to be false for $n = 2$ in [Vij32] and [BBV37]. Bank [Ban75] then adapted the previous counter-examples to provide a DAE whose non-unique increasing real-analytic solutions at infinity do not have any majorant. See the discussions (and Conjecture 6.1) in [Bos86] for discussions about the growth of solutions of DAEs, and their relations to functions $e_n(x)$.

Thus, the first important part of this paper is to refine Bank's counter-example to build `fastgen`, a fast-growing function that satisfies even stronger properties. The second major ingredient is to be able to approximate a function with arbitrary precision everywhere. Since this is a difficult task, we use `fastgen` to our advantage to show that it is enough to approximate functions that are bounded and change slowly (think 1-Lipschitz, although the exact condition is more involved). That is to say, to deal with the case where there is no problem about the growth and rate of change of functions in some way. This is the purpose of the function `pwcgen` which can build arbitrary almost piecewise constant functions as long as they are bounded and change slowly.

It should be noted that in the entire paper, we construct *generable functions* (in several variables) (see Section 2.1). For most of the constructions, we only use basic facts like the fact that generable functions are stable under arithmetic, composition and ODE solving. We know that generable functions satisfy polynomial partial equations and use this fact only at the very end to show that the generable approximation that we have built, in fact, translates to a polynomial ordinary differential equation.

The rest of the paper is organised as follows. In Section 2, we recall some concepts and results from other articles. The main purpose of this section is to present Theorem 2.13. This theorem is the analog equivalent of doing an assignment in a periodic manner. Section 3 is devoted to `fastgen`, the fast-growing function. In Section 4, we show how to generate a sequence of dyadic rationals. In Section 5, we show how to generate a sequence of bits. In Section 6, we show how to leverage the two previous sections to generate arbitrary almost piecewise constant functions. Section 7 is then devoted to the proof of our main theorem.

2. Concepts and results from previous work

2.1. **Generable functions.** The following concept can be attributed to [Sha41]: a function $f : \mathbb{R} \to \mathbb{R}$ is said to be a PIVP (Polynomial Initial Value Problem) function if there exists a system of the form $y' = p(y)$, where p is a (vector of) polynomial, with $f(t) = y_1(t)$ for all t, where y_1 denotes first component of the vector y defined in $\mathbb{R}^d$. We need in our proof to extend this concept to talk about multivariate functions. In [BGP17], we introduced the following class, which can be seen as extensions of [GBC09]. Let $\mathbb{K}$ be the smallest generable field (see [BGP17] for formal definitions and properties), the reader only needs to know that $\mathbb{Q} \subseteq \mathbb{K} \subseteq \mathbb{R}_P$ where $\mathbb{R}_P$ is the set of polynomial-time computable reals, and $\mathbb{K}$ is closed under images of generable functions.

Definition 2.1 (Generable function)**.** Let $d, e \in \mathbb{N}$, I be an open and connected subset of $\mathbb{R}^d$ and $f : I \to \mathbb{R}^e$. We say that f is generable if and only if there exists an integer $n \geqslant e$, a $n \times d$ matrix p consisting of polynomials with coefficients in $\mathbb{K}$, $x_0 \in \mathbb{K}^d$, $y_0 \in \mathbb{K}^n$ and $y : I \to \mathbb{R}^n$ satisfying for all $x \in I$:

- $y(x_0) = y_0$ and $J_y(x) = p(y(x))$ ▶ y satisfies a polynomial differential equation[5],
- $f(x) = (y_1(x), \ldots, y_e(x))$ ▶ the components of f are components of y.

This class strictly generalizes functions generated by polynomial ODEs. Indeed, in the special case of $d = 1$ (the domain of the function has dimension 1), the above definition is equivalent to saying that $y' = p(y)$ for some polynomial p. The interested reader can read more about this in [BGP17].

For the purpose of this paper, we will need to consider a slight generalisation of this notion where the initial condition is considered to be (depending of) a parameter, therefore defining not just a single function but a family of function, and most importantly, *all sharing the same differential equation*. Formally:

Definition 2.2 (Uniformly-generable function)**.** Let $d, m, e \in \mathbb{N}$, I be an open and connected subset of $\mathbb{R}^d$, $\Gamma \subseteq \mathbb{R}^m$, and $f : \Gamma \times I \to \mathbb{R}^e$. We say that f is uniformly-generable if and only if there exists an integer $n \geqslant e$, a $n \times d$ matrix p consisting of polynomials with coefficients in $\mathbb{K}$, $x_0 \in \mathbb{K}^d \cap I$ and a $(\rho^m, \rho^n)-$computable function $y_0 : \Gamma \to \mathbb{R}^n$ such that for all $\gamma \in \Gamma$, there exists $y : I \to \mathbb{R}^n$ satisfying for all $x \in I$:

- $y(x_0) = y_0(\gamma)$ and $J_y(x) = p(y(x))$ ▶ y satisfies a polynomial differential equation
- $f(\gamma; x) = (y_1(x), \ldots, y_e(x))$ ▶ the components of f are components of y.

For readability, we will distinguish parameters from variables using a semicolon, for example $f(\gamma; x)$ is parameterized by γ. This should make it clear from the context what is considered as parameter and what is considered as a variable.

Remark 2.3. Although we have chosen x_0 and the coefficients of p to be in $\mathbb{K}$ in the definition above, it is clear that we can change this set *at the cost of increasing the set* Γ *of parameters*. For example we could take all coefficients to be rational or in $\{0, 1\}$ by adding one extra parameter per coefficient and hence "hiding" them in y_0. The only real constraint is that since y_0 must remain computable, we still need all elements of $\mathbb{K}$ to be computable.

For the purpose of this paper, the reader only needs to know that the class of generable functions enjoys many stability properties that make it easy to create new functions from basic operations. Informally, one can add, subtract, multiply, divide and compose them at will, the only requirement is that the domain of definition must always be connected. In particular, the class of generable functions contains some common mathematical functions:

- (multivariate) polynomials;
- trigonometric functions: sin, cos, tan, etc;
- exponential and logarithm: exp, ln;
- hyperbolic trigonometric functions: sinh, cosh, tanh.

[5] J_y denotes the Jacobian matrix of y.

Two famous examples of functions that are *not* in this class are the ζ and Γ, we refer the reader to [BGP17] and [GBC09] for more information.

A nontrivial fact is that generable functions are always analytic. This property is well-known in the one-dimensional case but is less obvious in higher dimensions, see [BGP17] for more details. Moreover, generable functions satisfy the following crucial properties.

Lemma 2.4 (Closure properties of generable functions [BGP17]). *Let $f :\subseteq \mathbb{R}^d \to \mathbb{R}^n$ and $g :\subseteq \mathbb{R}^e \to \mathbb{R}^m$ be generable functions. Then $f+g$, $f-g$, fg, $\frac{f}{g}$ and $f \circ g$ are generable*[6]*.*

Lemma 2.5 (Generable functions are closed under ODE [BGP17]). *Let $d \in \mathbb{N}$, $J \subseteq \mathbb{R}$ an interval, $f :\subseteq \mathbb{R}^d \to \mathbb{R}^d$ generable, $t_0 \in J \cap \mathbb{K}$ and $y_0 \in \operatorname{dom} f \cap \mathbb{K}^d$. Assume there exists $y : J \to \operatorname{dom} f$ satisfying*

$$y(t_0) = y_0, \qquad y'(t) = f(y(t))$$

for all $t \in J$, then y is generable (and unique).

Those results can be generalised to uniformly-generable functions with the obvious restrictions on the domains and the roles of parameters. For example, if $f(\alpha; x)$ and $g(\beta; y)$ are uniformly-generable over $A \times X$ and $B \times Y$ respectively, then $h(\alpha, \beta; y) := f(\alpha; g(\beta; y))$ is uniformly-generable over $A \times B \times Y$. We will use those facts implicitly, and in particular the following result:

Theorem 2.6 (Uniformly-generable functions are closed under ODE). *Let $d, m \in \mathbb{N}$, $\Gamma \subseteq \mathbb{R}^m$, $t_0 \in \mathbb{K}$, J an open interval containing t_0, $f_0 : \Gamma \to \mathbb{R}^d$ a $(\rho^m, \rho^d)-$computable function and $F :\subseteq \Gamma \times \mathbb{R}^d \to \mathbb{R}^d$ uniformly-generable. Assume that there exists $f : \Gamma \times J \to \mathbb{R}^d$ satisfying*[7]

$$f(\gamma; t_0) = f_0(\gamma), \qquad \tfrac{\partial f}{\partial t}(\gamma; t) = F(\gamma; f(\gamma; t))$$

for all $\gamma \in \Gamma$ and $t \in J$. Then f is uniformly-generable (and unique).

Proof. Apply Definition 2.2 to F to get $n \in \mathbb{N}$, $x_0 \in \mathbb{K}^n$, $y_0 : \Gamma \to \mathbb{R}^n$ computable and p polynomial matrix with coefficients in $\mathbb{K}$. Then given $\gamma \in \Gamma$, there exists $y :\subseteq \mathbb{R}^d \to \mathbb{R}^n$ such that

$$y(x_0) = y_0(\gamma), \qquad J_y(x) = p(y(x))$$

and $F(\gamma; x) = (y_1(x), \ldots, y_d(x)) =: y_{1..d}(x)$ for all $(\gamma, x) \in \operatorname{dom} F$. Let $z(t) = y(f(\gamma; t))$ which is well-defined by assumption and check that

$$z'(t) = J_y(f(\gamma; t)) \tfrac{\partial f}{\partial t}(\gamma; t) = p(y(f(\gamma; t))) F(\gamma; f(\gamma; t)) = p(z(t)) z_{1..d}(t) = q(z(t))$$

for some polynomial q that does not depend on γ, and $z(t_0) = y(f(\gamma; t_0)) = y(f_0(\gamma))$ which is a computable function of γ since f_0 is computable and $(\gamma, x) \mapsto y(x)$ is also computable (note that y depends on γ) by Proposition 2.7. □

An important point, which we have in fact already used in the proof of the previous proposition, is that generable functions are always computable, in the sense of Computable Analysis.

Proposition 2.7 (Generable implies computable). *Assume $f : \Gamma \times I \to \mathbb{R}^e$ is uniformly generable according to Definition 2.2: Hence there is a $(\rho^m, \rho^n)-$computable function $y_0 : \Gamma \to \mathbb{R}^n$ and a $n \times d$ matrix p consisting of polynomials with coefficients in $\mathbb{K}$, $x_0 \in \mathbb{K}^d$ that*

[6]With the obvious dimensional condition associated with each operation.

[7]We are assuming that for all $\gamma \in \Gamma$, $(\gamma; f(\gamma; t)) \in \operatorname{dom} F$.

define $y(\gamma;x_0)=y_0(\gamma)$ *and* $J_y(x)=p(y(\gamma;x))$*. Then the function that maps* $(\gamma,x)\in\Gamma\times I$ *to* $y(\gamma;x)$ *is* $([\rho^m,\rho^d],\rho^n)-$*computable.*

Proof. We established in proposition [BGP17, Proposition 31] that $y(x)$ is necessarily real-analytic on some neighbourhood $V=V(x)$ of x for all x that corresponds to some point of the domain of f.

Some explicit upper bound on the radius of convergence is provided by [PG16, Theorem 5]: Assuming $t_0=0$, $k=\deg(p)\geq 2$, $\alpha=\max(1,\|y_0\|)$, the radius is at least $1/M$ with $M=M(y_0)=(k-1)\Sigma p\alpha^{k-1}$, where Σp is basically the sum of the absolute value of the coefficients of polynomials in matrix p.

Consequently, using classical techniques for evaluating a converging power series whose convergence radius is known up to a given precision (by restricting the sum up to suitable index) we get that y is computable over the ball $V(y_0)$ of radius $1/(2M)$.

Computability of y then follows from classical analytic continuation techniques: A Turing machine can then extend the computation starting from a new point y_1 in $V(y_0(\gamma))$, and then repeat the above process to compute y over some ball $V(y_1)$ of radius $1/(2M(y_1))$, and so on. Repeating the process, eventually, it will reach x and will be able to compute $y(x)$. Refer to [KTZ18, Thi18] for similar techniques and a finer complexity analysis. □

2.2. **Helper functions and constructions.** We mentioned earlier that a number of common mathematical functions are generable. However, for our purpose, we will need less common functions that one can consider to be programming gadgets.

Remark 2.8. In this subsection, some of the functions will be introduced as mapping arguments to value, i.e. as usual mathematical functions, but some others by the properties of their solutions (e.g. `reach`, `pereach`, `pil`). In the latter case, an explicit expression of a function satisfying those properties can be found in the proof.

One such operation is rounding (computing the nearest integer). Note that, by construction, generable functions are analytic and in particular must be continuous. It is thus clear that we cannot build a perfect rounding function and in particular we have to compromise on two aspects:

- we cannot round numbers arbitrarily close to $n+\frac{1}{2}$ for $n\in\mathbb{Z}$ because of continuity: thus the function takes a parameter λ to control the size of the "zone" around $n+\frac{1}{2}$ where the function does not round properly;
- we cannot round without error due to the uniqueness of analytic functions: thus the function takes a parameters μ that controls how good the approximation must be.

Lemma 2.9 (Round, [BGP17])**.** *There exists a generable function* `round` *such that for any* $n\in\mathbb{Z}$*,* $x\in\mathbb{R}$*,* $\lambda>2$ *and* $\mu\geqslant 0$*:*

- *if* $x\in\left[n-\frac{1}{2},n+\frac{1}{2}\right]$ *then* $|\mathtt{round}(x,\mu,\lambda)-n|\leqslant\frac{1}{2}$*;*
- *if* $x\in\left[n-\frac{1}{2}+\frac{1}{\lambda},n+\frac{1}{2}-\frac{1}{\lambda}\right]$ *then* $|\mathtt{round}(x,\mu,\lambda)-n|\leqslant e^{-\mu}$*.*

Another very useful operation is the analog equivalent of a discrete assignment, done in a periodic manner. More precisely, we consider a particular class of ODEs

$$y'(t)=\mathtt{pereach}(t,\phi(t),y(t),g(t))$$

adapted from the constructions of [BGP16a], where g and ϕ are sufficiently nice functions. Solutions to this equation alternate between two behaviours, for all $n\in\mathbb{N}$:

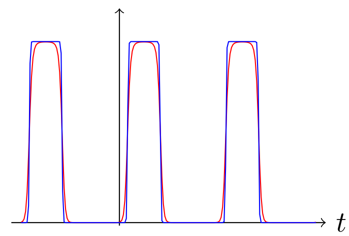

Figure 2: Illustration of $\mathtt{pil}(\mu, \cdot)$ from Lemma 2.12 for various values of μ: it has period 1, is very small ($\leqslant e^{-\mu}$) half of the time, and the integral of the remaining half is at least 1.

- During $J_n = [n, n + \frac{1}{2}]$, the system performs $y(t) \to \bar{g}$ for some $\bar{g}$ satisfying $\min_{t \in J_n} g(t) \leqslant \bar{g} \leqslant \max_{t \in J_n} g(t)$ (note that this is voluntarily underspecified). So in particular, if $g(t) \approx \bar{g}$ over this time interval, then $y(t) \to \bar{g}$ and the system performs an "assignment" in the sense that $y(n + \frac{1}{2}) := \bar{g}$. Then ϕ controls how good the convergence is: the error is of the order of $e^{-\phi}$.
- During $J'_n = [n + \frac{1}{2}, n + 1]$, the systems tries to keep y constant, ie $y' \approx 0$. More precisely, the system enforces that $|y'(t)| \leqslant e^{-\phi(t)}$.

As a result of this behavior, if $g(t) \approx \bar{g}$ for $t \in [n, n + \frac{1}{2}]$ then the system performs the "assignment" $y(n + 1) := \bar{g}$ with some error that is exponential small in ϕ.

We now go to the proof of the existence of such a function `pereach` (formally stated as Theorem 2.13): We will need the following bound on tanh, which essentially tells us that $\tanh(t)$ gets exponentially close (in $|t|$) to ± 1 as $t \to \pm\infty$.

Lemma 2.10. *For any $t \in \mathbb{R}$, $|\tanh(t) - \mathtt{sgn}(t)| \leqslant e^{-|t|}$.*

Lemma 2.11 (Reach, [BGP16b]). *There exists a generable function* `reach` *such that for any $\phi \in C^0(\mathbb{R}_{\geqslant 0})$, $g \in C^0(\mathbb{R})$ and $y_0 \in \mathbb{R}$, the unique solution to*

$$y(0) = y_0, \qquad y'(t) = \phi(t)\mathtt{reach}(g(t) - y(t))$$

exists over $\mathbb{R}_{\geqslant 0}$. Furthermore, for any $I = [a, b] \subseteq [0, +\infty)$, if there exists $\bar{g} \in \mathbb{R}$ and $\eta \in \mathbb{R}_{\geqslant 0}$ such that $|g(t) - \bar{g}| \leqslant \eta$ for all $t \in I$, then for all $t \in I$,

$$|y(t) - \bar{g}| \leqslant \eta + \exp\left(-\int_a^t \phi(u)du\right) \qquad \text{whenever } \int_a^t \phi(u)du \geqslant 1.$$

Furthermore, for all $t \in I$,

$$\min(\bar{g} - \eta, y(a)) \leqslant y(t) \leqslant \max(\bar{g} + \eta, y(a))$$

and in particular

$$|y(t) - \bar{g}| \leqslant \max(\eta, |y(a) - \bar{g}|).$$

Proof Remark. The statement of [BGP16b, Lemma 40] only contains the first and third inequalities but in fact the proof also contains the second inequality (which is strictly stronger than the third but less immediate to use). □

Lemma 2.12 (Periodic integral-low, see Figure 2). *There exists a generable function* $\mathtt{pil} : \mathbb{R}_{\geqslant 0} \times \mathbb{R} \to \mathbb{R}_{\geqslant 0}$ *such that:*

- $\mathtt{pil}(\mu, \cdot)$ *is 1-periodic, for any $\mu \in \mathbb{R}_{\geqslant 0}$;*

- $\int_0^{1/2} \texttt{pil}(\mu(t),t)dt \geqslant 1$ *for any* $\mu \in C^0(\mathbb{R}_{\geqslant 0})$*;*
- $|\texttt{pil}(\mu,t)| \leqslant e^{-\mu}$ *for any* $\mu \in \mathbb{R}_{\geqslant 0}$ *and* $t \in \left[\frac{1}{2},1\right]$*.*

Proof. For any $t \in \mathbb{R}$ and $\mu \in \mathbb{R}_{\geqslant 0}$, let

$$\texttt{pil}(\mu,t) = A\left(1+\tanh\left(2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu)\right)\right).$$

where $A=3$. Clearly $\texttt{pil}$ is generable and 1-periodic in t. Let $\mu \in \mathbb{R}_{\geqslant 0}$ and $t \in [\frac{1}{2},1]$, then

$$\begin{aligned}
\sin(2\pi t) &\leqslant 0 \\
2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu) &\leqslant -A-\mu \\
|\tanh(2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu))-(-1)| &\leqslant e^{-A-\mu} && \text{using Lemma 2.10} \\
A|\tanh(2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu))-(-1)| &\leqslant Ae^{-A-\mu} \\
|\texttt{pil}(\mu,t)| &\leqslant e^{-\mu}.
\end{aligned}$$

Let $\mu \in C^0(\mathbb{R}_{\geqslant 0})$ and $t \in [0,\frac{1}{2}]$. Observe that $\texttt{pil}(\mu(t),t) \geqslant 0$ and, furthermore, if $t \in [\frac{1}{8},\frac{3}{8}]$ then

$$\begin{aligned}
\sin(2\pi t) &\geqslant \frac{\sqrt{2}}{2} \\
2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu(t)) &\geqslant \sqrt{2}-1 && \text{since } A+\mu(t)\geqslant 1 \\
\tanh(2(\sin(2\pi t)-\tfrac{1}{2})(A+\mu(t))) &\geqslant \tanh(\sqrt{2}-1) \geqslant \frac{1}{3} \\
\texttt{pil}(\mu(t),t) &\geqslant \tfrac{4}{3}A.
\end{aligned}$$

It follows that

$$\int_0^{\frac{1}{2}} \texttt{pil}(\mu(t),t)dt \geqslant \int_{\frac{1}{8}}^{\frac{3}{8}} \texttt{pil}(\mu(t),t)dt \geqslant \left(\frac{3}{8}-\frac{1}{8}\right)\frac{4}{3}A \geqslant \frac{A}{3} \geqslant 1.$$ ☐

Theorem 2.13 (Periodic reach)**.** *There exists a generable function* $\texttt{pereach} : \mathbb{R}^2_{\geqslant 0}\times\mathbb{R}^2 \to \mathbb{R}$ *such that for any* $I=[n,n+1]$ *with* $n\in\mathbb{N}$, $y_0\in\mathbb{R}$, $\phi,\psi \in C^0(I,\mathbb{R}_{\geqslant 0})$ *and* $g \in C^0(I,\mathbb{R})$*, the unique solution to*

$$y(n)=y_0, \qquad y'(t)=\psi(t)\texttt{pereach}(t,\phi(t),y(t),g(t))$$

exists over I*. Furthermore,*

- **(i)** *For all* $\bar{g}\in\mathbb{R}$ *and* $\eta,\theta\in\mathbb{R}_{\geqslant 0}$ *such that* $|g(t)-\bar{g}|\leqslant\eta$ *and* $\psi(t)\phi(t)\geqslant\theta\geqslant 1$ *for all* $t\in[n,n+\frac{1}{2}]$*, we have that* $|y(n+\frac{1}{2})-\bar{g}|\leqslant \eta+\exp(-\theta)$*.*
- **(ii)** *For all* $t\in[n,n+1]$, $\bar{g}\in\mathbb{R}$ *and* $\eta\in\mathbb{R}_{\geqslant 0}$ *such that* $|g(u)-\bar{g}|\leqslant\eta$ *for all* $u\in[n,t]$*, we have that* $|y(t)-\bar{g}|\leqslant\max(\eta,|y(n)-\bar{g}|)$*.*
- **(iii)** *For all* $t\in[n+\frac{1}{2},n+1]$, $|y(t)-y(n+\frac{1}{2})|\leqslant\int_{n+\frac{1}{2}}^t\psi(u)\exp(-\phi(u))\,du$*.*
- **(iv)** *For all* $\theta\in\mathbb{R}_{\geqslant 0}$ *such that* $\psi(t)\phi(t)\geqslant\theta\geqslant 1$ *for all* $t\in[n,n+\frac{1}{2}]$*, we have that* $y(n+\frac{1}{2})\geqslant\min_{u\in[n,n+\frac{1}{2}]}g(u)-\exp(-\theta)$*.*
- **(v)** *For all* $t\in[n,n+1]$, $\min\left(y(n),\min_{u\in[n,t]}g(t)\right)\leqslant y(t)\leqslant\max\left(y(n),\max_{u\in[n,t]}g(t)\right)$*.*

Proof. Define $\texttt{pereach}(t,\phi,y,g)=\texttt{pil}(\phi+r^2,t)r$ where $r=\phi\texttt{reach}(g-y)$ where $\texttt{pil}$ is defined in Lemma 2.12 and $\texttt{reach}$ is defined in Lemma 2.11. Fix $n\in\mathbb{N}$ and $I=[n,n+1]$.

First notice that $\texttt{pil}$ is nonnegative by Lemma 2.12 thus by Lemma 2.11, the solution must exists over I. We now prove each point separately:

(i) We have that

$$\int_n^{n+\frac{1}{2}} \texttt{pil}(\phi(u)+r(u)^2,u)\psi(u)\phi(u)du \geqslant \theta \int_n^{n+\frac{1}{2}} \texttt{pil}(\phi(u)+r(u)^2,u)du \geqslant \theta \geqslant 1$$

by Lemma 2.12. Thus $|y(n+\frac{1}{2})-\bar{g}| \leqslant \eta + e^{-\theta}$ by Lemma 2.11.

(ii) Apply Lemma 2.11 to the interval $[n,t]$.

(iii) We have that

$$|y'(t)| = |\psi(t)\texttt{pil}(\phi(t)+r(t)^2,u)r(t)| \leqslant \psi(t)e^{-\phi(t)-r(t)^2}|r(t)| \leqslant \psi(t)e^{-\phi(t)}$$

by Lemma 2.12, for all $t \in [n+\frac{1}{2}, n+1]$. The inequality follows by integration.

(iv) Let $m = \min_{u\in[n,n+\frac{1}{2}]} g(u)$ and $M = \max_{u\in[n,n+\frac{1}{2}]} g(u)$, define $\bar{g} = \frac{m+M}{2}$ and $\eta = \frac{M-m}{2}$. Then the assumptions of item **(i)** are satisfied and we get that $|y(n+\frac{1}{2})-\bar{g}| \leqslant \eta + e^{-\theta}$ so in particular $y(t) \geqslant \bar{g} - \eta - e^{-\theta}$ but $\bar{g} - \eta = m$ so this concludes.

(v) The last item is more subtle because we want to use item **(ii)** but we do not know if $y(n) - \bar{g}$ and $y(t) - \bar{g}$ have the same sign. Let $m = \min_{u\in[n,t]} g(u)$ and $M = \max_{u\in[n,t]} g(u)$, define $\bar{g} = \frac{m+M}{2}$ and $\eta = \frac{M-m}{2}$. Then the assumptions of Lemma 2.11 are satisfied over $[n,t]$ and we get that $\min(\bar{g}-\eta, y(n)) \leqslant y(t)$ but $\bar{g} - \eta = m$ so this concludes. □

2.3. **Computable Analysis and Representations.** In order to prove the computability of the map $(f,\varepsilon) \mapsto \alpha$ in Theorems 1.3 and 1.4, we need to express the related notion of computability for real numbers, functions and operators. We recall here the related concepts: Computable Analysis, specifically Type-2 Theory of Effectivity (TTE) [Wei00], is a theory to study algorithmic aspects of real numbers, functions and higher-order operators over real numbers. Subsets of real numbers are also of great interest to this theory but will not need them in this paper. This theory is based on classical notions of computability (and complexity) of Turing machines which are applied to problems involving real numbers, usually by means of (effective) approximation schemes. We refer the reader to [Wei00, BHW08, Bra05] for tutorials on Computable Analysis. In order to avoid a lengthy introduction on the subject, we simply introduce the elements required for the paper at a very high level. In what follows, Σ is a finite alphabet.

The core concept of TTE is that of *representation*: a representation of a space X is simply a surjective function $\delta :\subseteq \Sigma^\omega \to X$. If $x \in X$ and $p \in \Sigma^\omega$ is such that $\delta(p) = x$ then p is called a δ-*name* of x: p is one way of describing x with a (potentially infinite) string. In TTE, all computations are done on infinite string (names) using Type 2 machines, which are Turing machines operating on infinite strings but where each bit of the output only depends on a finite prefix of the input. Type 2 machines give rise to the notion of computable functions from Σ^ω to Σ^ω. Given two representations δ_X, δ_Y of some spaces X and Y, one can define two interesting notions:

- δ_X-computable elements of X: those are the elements x such that $\delta_X(p) = x$ for some *computable name* p ($p : \mathbb{N} \to \Sigma$ is computable by a usual Turing machine);

- (δ_X, δ_Y)-computable functions from X to Y: those are the functions $f :\subseteq X \to Y$ for which we can find a *computable realiser* F ($F :\subseteq \sigma^\omega \to \omega$ is computable by a Type 2 machine) such that $f \circ \delta_X = \delta_Y \circ F$.

$$\begin{tikzcd} X \arrow[r, "f"] & Y \\ \Sigma^\omega \arrow[u, "\delta_X"] \arrow[r, "F"] & \Sigma^\omega \arrow[u, "\delta_Y"'] \end{tikzcd}$$

In this paper, we will only need a few representations to manipule real numbers, sequences and continuous real functions:

- $\nu_{\mathbb{N}} :\subseteq \Sigma^\omega \to \mathbb{N}$ is a representation of the integers. The details of the encoding at not very important, since natural representations such as unary and binary representations are equivalent.
- $\nu_{\mathbb{Q}} :\subseteq \Sigma^\omega \to \mathbb{N}$ is a representation of the rational numbers, again the details of the encoding at not very important for natural representations.
- $\rho :\subseteq \Sigma^\omega \to \mathbb{R}$ is the *Cauchy representation* of real numbers which intuitively encodes a real number x by a converging sequence of intervals $[l_n, r_n] \ni x$ of rationals numbers. Alternatively, one can also use Cauchy sequences with a known rate of convergence.
- $[\delta_X, \delta_Y] :\subseteq \Sigma^\omega \to X \times Y$ is the representation of pairs of elements of (X, Y) where the first (resp. second) component uses δ_X (resp. δ_Y). In particular, δ^k is a shorthand notation of the representation $[\delta, [\delta, [\ldots]]]$ of X^k. In this paper we will often use ρ^k to represent $\mathbb{R}^k$.
- $\delta^\omega :\subseteq \Sigma^\omega \to X^{\mathbb{N}}$ is the representation of sequences of elements of X, represented by δ. For example ρ^ω can be used to represent sequences of real numbers.
- $[\delta_X \to \delta_Y]_Z :\subseteq \Sigma^\omega \to C^0(Z, Y)$ is the representation of *continuous*[8] *functions* from $Z \subseteq X$ to Y, we omit Z if $Z = X$. We will mostly need $[\rho^k \to \rho]$ which represents[9] $C^0(\mathbb{R}^k, \mathbb{R})$ as a list of boxes which enclose the graph of the function with arbitrary precision. Informally, it means we can "zoom" on the graph of the function and plot it with arbitrary precision.

It will be enough for the reader to know that those representations are well-behaved. In particular, the following functions are computable (we always use ρ to represent $\mathbb{R}$):

- the arithmetical operations $+, -, \cdot, / :\subseteq \mathbb{R} \times \mathbb{R} \to \mathbb{R}$,
- polynomials $p : \mathbb{R}^n \to \mathbb{R}$ with computable coefficients,
- elementary functions $\cos, \sin, \exp$ over $\mathbb{R}$.

Furthermore, the following operators on continuous functions are computable:

- the arithmetical operators $+, -, \cdot, / :\subseteq C^0(\mathbb{R}) \times C^0(\mathbb{R}) \to C^0(\mathbb{R})$,
- composition $\circ : C^0(X, Y) \times C^0(Y, Z) \to C^0(X, Z)$,
- inverse $\cdot^{-1} : C^0(X, Y) \to C^0(Y, X)$ for increasing (or decreasing) functions,
- evaluation $C^0(X, Y) \times X \to Y$, $(f, x) \mapsto f(x)$.

We will also use the fact that the map $X^{\mathbb{N}} \times \mathbb{N} \to X, (x, i) \mapsto x_i$ is $([\delta^\omega, \nu_N], \delta)$-computable for any space X represented by δ.

[8] Without giving too much details, this requires X and Y to be T_0 spaces with countable basis and δ_X, δ_Y to be admissible. It will be enough to know that ρ is admissible for the usual topology on $\mathbb{R}$.

[9] Technically, is equivalent to a representation of.

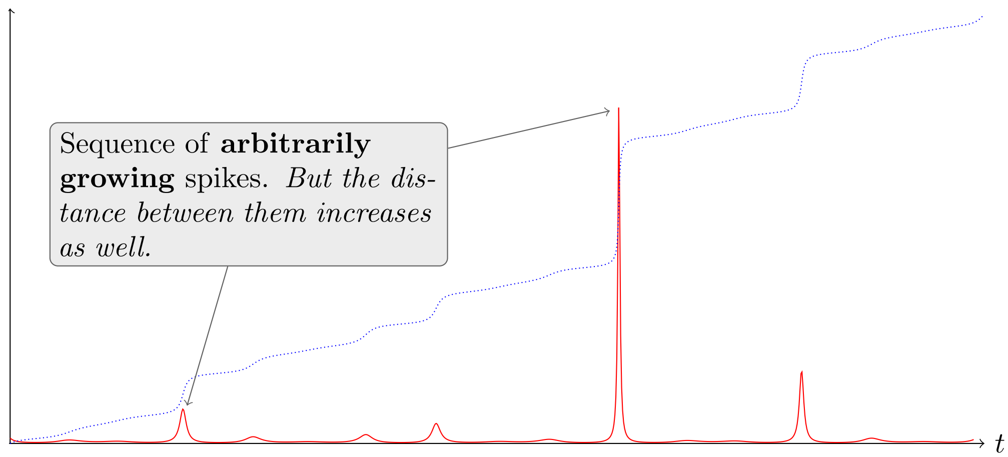

Figure 3: Illustration of g (in dotted blue) from Lemma 3.1: we start from a function f (in red) that spikes and then integrate it to make it increasing.

Refer to [Wei00, BHW08, Bra05] for more complete discussions, and in particular to [KTZ18, Thi18] for computability and complexity issues related to ordinary differential equations solving.

3. Generating fast growing functions

Our construction crucially relies on our ability to build functions of arbitrary growth. At the end of this section, we obtain a function `fastgen` with a straightforward specification: for any infinite sequence $a_0, a_1, \dots$ of positive numbers, we can find a suitable $\alpha \in \mathbb{R}$ such that $\texttt{fastgen}(\alpha; n) \geqslant a_n$ for all $n \in \mathbb{N}$. Furthermore, we can ensure that $\texttt{fastgen}(\alpha; \cdot)$ is increasing. Notice, and this is the key point, that the definition of `fastgen` is independent of the sequence a: a single generable function (and thus differential system) can have arbitrary growth by simply tweaking its initial value.

Our construction builds on the following lemma proved by [Ban75], based on an example of [BBV37]. The proof essentially relies on the function $\frac{1}{2-\cos(x)-\cos(\alpha x)}$ which is generable and well-defined for all positive x if α is irrational. By carefully choosing α, we can make $\cos(x)$ and $\cos(\alpha x)$ simultaneously arbitrary close to 1. This function is illustrated on Figure 3.

Lemma 3.1 [Ban75]**.** *There exists a positive generable function g and an absolute constant $c > 0$ such that for any increasing sequence $a \in \mathbb{N}^{\mathbb{N}}$ with $a_n \geqslant 2$ for all n, there exists $\alpha \in \mathbb{R}$ such that $g(\alpha, \cdot)$ is defined over $[1, \infty)$, nondecreasing and for any $n \in \mathbb{N}$ and $t \geqslant 2\pi b_n$, $g(\alpha, t) \geqslant c a_n$ where $b_n = \prod_{k=0}^{n-1} a_k$. Furthermore, the map $a \mapsto \alpha$ is $(\nu_{\mathbb{N}}^{\omega}, \rho)$-computable.*

Proof. We give a sketch of the proof, following the presentation from [Ban75]. For any $\alpha \in \mathbb{R}$ and $t > 0$, let

$$f(\alpha, t) = \frac{1}{2 - \cos(t) - \cos(\alpha t)}.$$

Since sin and cos are generable, it follows that f is generable because it has a connected domain of definition. Indeed, $f(\alpha,t)$ is well-defined except on

$$X = \{(\alpha, 2k\pi) : \alpha \in \mathbb{Q}, k \in \mathbb{N}, k\alpha \in \mathbb{N}\}$$

which is a totally disconnected set in $\mathbb{R}^2$. Let

$$\alpha_a = \sum_{n=1}^{\infty} b_n^{-1} \qquad \text{where } b_n = \prod_{k=0}^{n-1} a_k \tag{3.1}$$

which is well-defined if a_n is a strictly increasing sequence. Indeed, it implies that $b_n \geqslant (n-1)!$ and $\alpha_a \leqslant \sum_{n=0}^{\infty} \frac{1}{n!} = e$. One can easily show (by contradiction for example) that α_a must be irrational. Also define

$$g(\alpha, t) = \int_1^t f(\alpha, u)du$$

which is generable. Let $n \in \mathbb{N}$, define $\delta_n = \sum_{k=n+1}^{\infty} \frac{b_n}{b_k}$. Let $t \in [2\pi(b_n - \delta_n), 2\pi b_n]$, write $\varepsilon = 2\pi b_n - t \in [0, 2\pi\delta_n]$ and observe that

$$1 - \cos(t) = 1 - \cos(t - 2\pi b_n) = 1 - \cos(\varepsilon) \leqslant \varepsilon^2 \leqslant 4\pi^2\delta_n^2.$$

Furthermore, and note that

$$\begin{aligned}
1 - \cos(\alpha t) &= 1 - \cos(2\pi\alpha b_n - \varepsilon) \\
&= 1 - \cos\left(2\pi \sum_{k=0}^{n} \tfrac{b_n}{b_k} + 2\pi \sum_{k=n+1}^{\infty} \tfrac{b_n}{b_k} - \varepsilon\right) \\
&= 1 - \cos\left(2\pi \sum_{k=n+1}^{\infty} \tfrac{b_n}{b_k} - \varepsilon\right) && \text{since } \tfrac{b_n}{b_k} \in \mathbb{N} \text{ for } k \leqslant n \\
&= 1 - \cos\left(2\pi\delta_n - \varepsilon\right) \\
&\leqslant (2\pi\delta_n - \varepsilon)^2 && \text{since } 1 - \cos(x) \leqslant x^2 \\
&\leqslant (2\pi\delta_n)^2 && \text{since } \varepsilon \leqslant 2\pi\delta_n.
\end{aligned}$$

It follows that

$$f(\alpha_a, t) = (1 - \cos(t) + 1 - \cos(\alpha t))^{-1} \geqslant (4\pi^2\delta_n^2 + 4\pi^2\delta_n^2)^{-1} \geqslant \tfrac{1}{8\pi^2\delta_n^2}.$$

Thus

$$\begin{aligned}
g(\alpha_a, 2\pi b_n) &= \int_1^{2\pi b_n} f(\alpha_a, t)dt \\
&\geqslant \int_{2\pi(b_n-\delta_n)}^{2\pi b_n} f(\alpha_a, t)dt && \text{since } f \text{ is positive} \\
&\geqslant \int_{2\pi(b_n-\delta_n)}^{2\pi b_n} \tfrac{1}{8\pi^2\delta_n^2} dt \\
&= \tfrac{\delta_n}{8\pi^2\delta_n^2} = \tfrac{1}{8\pi^2\delta_n}.
\end{aligned}$$

But note that

$$\delta_n = \sum_{k=n+1}^{\infty} \tfrac{b_n}{b_k} \leqslant \sum_{k=n+1}^{\infty} a_n^{n-k} \qquad \text{since } \tfrac{b_n}{b_k} = (a_n \cdots a_{k-1})^{-1}$$

$$= \frac{a_n^{-1}}{1-a_n^{-1}} \leqslant 2a_n^{-1} \qquad\qquad \text{since } a_n \geqslant 2.$$

It then easily follows that

$$g(\alpha_a, 2\pi b_n) \geqslant \tfrac{a_n}{16\pi^2}$$

and the result follows from the fact that g is nondecreasing.

The computability of the map $a \mapsto \alpha$ is the only missing result. It is immediate from (3.1) that the map $(a, n) \mapsto b_n$ is $([\nu_{\mathbb{N}}^{\omega}, \nu_{\mathbb{N}}], \nu_{\mathbb{N}})$-computable since each b_n is a product of finitely many a_i. Furthermore, $b_n \geqslant (n-1)!$ thus for any $n \geqslant 1$,

$$\left|\alpha_a - \sum_{i=1}^{n} b_i^{-1}\right| \leqslant \sum_{i \geqslant n-1} \frac{1}{(i-1)!} \leqslant \sum_{i \geqslant n} \frac{1}{i!} \leqslant \sum_{i \geqslant 0} \frac{1}{n!2^i} \leqslant \tfrac{2}{n!} \leqslant 2^{2-n}.$$

It follows that $\left(\sum_{i=1}^{n} b_i^{-1}\right)_{i \in \mathbb{N}}$ is a Cauchy sequence of α_a of known convergence rate. It suffice to note that $(b, n) \mapsto \sum_{i=1}^{n} b_i^{-1}$ is $([\nu_{\mathbb{N}}^{\omega}, \nu_{\mathbb{N}}], \rho)$-computable, since it only involves a finite number of sum and inverses of real numbers. □

Remark 3.2. As noted earlier, Lemma 3.1 and the Theorem 3.3 build *partial functions*. Indeed we only show that the solution exists at all times $t \in \mathbb{R}_{\geqslant 0}$ for certain well-chosen α. In particular, it can be easily checked that the ODE in Lemma 3.1 explodes in finite time for all rational α.

Essentially, Lemma 3.1 proves that there exists a function g such that for any $n \in \mathbb{N}$, $g(\alpha, a_0 a_1 \cdots a_{n-1}) \geqslant a_n$. Note that this is not quite what we are aiming for: the function g is indeed $\geqslant a_n$ but at times $a_0 a_1 \cdots a_{n-1}$ instead of n. Since $a_0 a_1 \cdots a_{n-1}$ is a very big number compared to n, g does not grow fast enough for our needs. The idea is to “accelerate” g by composing it with a fast growing function h, ideally such that $h(n) \geqslant a_0 \cdots a_{n-1}$. This would ensure that $g(h(n)) \geqslant n$. This is a chicken-and-egg problem because to build such a function h, we need to build a fast growing function! We now try to explain how to solve this problem.

Fix a sequence $(a_n)_n$ and let g be the function from Lemma 3.1 and α_a be the parameter that corresponds to a (we omit the α_a for readability so $g(x) = g(\alpha_a, x)$). Consider the following sequence:

$$x_0 = a_0, \qquad x_{n+1} = x_n g(x_n).$$

Then observe that

$$x_1 = x_0 g(x_0) = a_0 g(a_0) \geqslant a_0 a_1, \qquad x_2 = x_1 g(x_1) \geqslant a_0 a_1 g(a_0 a_1) \geqslant a_0 a_1 a_2, \quad \ldots$$

It is not hard to see that $x_n \geqslant a_0 a_1 \cdots a_n \geqslant a_n$. We then use our generable gadget of Section 2.2 to simulate this discrete sequence with a differential equation. Intuitively, we build a differential equation such that the solution y satisfies $y(n) \approx x_n$. More precisely, we use two variables y and z such that over $[n, n+1/2]$, $z' \approx 0$ and $y(t) \to zg(z)$ and over $[n+1/2, n+1]$, $y' \approx 0$ and $z(t) \to y$. Then if $y(n) \approx z(n) \approx x_n$ then $y(n+1) \approx z(n+1) \approx x_{n+1}$.

Theorem 3.3. *There exists $\Gamma \subseteq \mathbb{R}$ and a positive uniformly-generable function* $\texttt{fastgen}$: $\Gamma \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}$ *such that for any $x \in \mathbb{R}_{\geqslant 0}^{\mathbb{N}}$, there exists $\alpha \in \Gamma$ such that for any $n \in \mathbb{N}$ and $t \in \mathbb{R}_{\geqslant 0}$,*

$$\texttt{fastgen}(\alpha; t) \geqslant x_n \qquad \text{if } t \geqslant n.$$

Furthermore, $\texttt{fastgen}(\alpha; \cdot)$ *is nondecreasing. In addition, the map $x \mapsto \alpha$ is (ρ^{ω}, ρ)-computable.*

Proof. Let $\delta = 4$. Apply Lemma 3.1 to get g and c. Let $a \in \mathbb{N}^N$ be an increasing sequence such that $a_n \geqslant x_n$, then there exists $\alpha_x \in \mathbb{R}$ such that for all $n \in \mathbb{N}$,

$$g(\alpha_x, t) \geqslant c a_n \tag{3.2}$$

for all $t \geqslant 2\pi b_n$ where $b_n = \prod_{k=0}^{n-1} a_k$. Consider the following system of differential equations, for $\phi = 2$,

$$\begin{cases} y(0) = \delta + 2\pi \\ z(0) = \delta + 2\pi \end{cases}, \qquad \begin{cases} y'(t) = \texttt{pereach}\left(t, \phi, y(t), \delta + \frac{1}{c} z(t) g(\alpha_x, z(t))\right) \\ z'(t) = \texttt{pereach}\left(t + \frac{1}{2}, \phi, z(t), 1 + y(t)\right) \end{cases}.$$

Apply Theorem 2.13 to show that y and z exist over $\mathbb{R}_{\geqslant 0}$. We will show the following result by induction on $n \in \mathbb{N}$:

$$\min(y(n), z(n)) \geqslant \delta + 2\pi b_n. \tag{3.3}$$

The result is trivial for $n = 0$ since $y(0) = z(0) = \delta + 2\pi = \delta + 2\pi b_0 \geqslant 1 + 2\pi b_0$. Let $n \in \mathbb{N}$ and assume that (3.3) holds for n. Apply Theorem 2.13 (item **(iii)**) to z to get that for any $t \in [n, n + \frac{1}{2}]$,

$$|z(t) - z(n)| \leqslant \int_n^t \exp(-\phi)\, du \leqslant \int_n^{n+\frac{1}{2}} \exp(-\phi)\, du \leqslant \tfrac{1}{2} e^{-\phi} \leqslant 1.$$

In particular, it follows that for any $t \in [n, n + \frac{1}{2}]$,

$$\begin{aligned} z(t) &\geqslant z(n) - 1 \geqslant 2\pi b_n && \text{since } z(n) \geqslant 1 + 2\pi b_n \\ g(\alpha_x, z(t)) &\geqslant c a_n && \text{using (3.2)} \\ z(t) g(\alpha_x, z(t)) &\geqslant 2\pi b_n c a_n && \text{since } z(t) \geqslant 2\pi b_n \\ &= 2\pi c b_{n+1} && \text{since } b_{n+1} = b_n a_n \\ \delta + \tfrac{1}{c} z(t) g(\alpha_x, z(t)) &\geqslant \delta + 2\pi b_{n+1}. \end{aligned}$$

Note that $\phi \geqslant 1$ then apply Theorem 2.13 (item **(iv)**) to y using the above inequality to get that

$$y(n + \tfrac{1}{2}) \geqslant \delta + 2\pi b_{n+1} - e^{-1} \geqslant \delta - 1 + 2\pi b_{n+1}$$

and (item **(v)**) for any $t \in [n, n + \frac{1}{2}]$,

$$y(t) \geqslant \min(y(n), \delta + 2\pi b_{n+1}) \geqslant \min(1 + 2\pi b_n, \delta + 2\pi b_{n+1}) \geqslant 1 + 2\pi b_n, \tag{3.4}$$

and (item **(iii)**) for any $t \in [n + \frac{1}{2}, n + 1]$,

$$|y(t) - y(n + \tfrac{1}{2})| \leqslant \int_{n+\frac{1}{2}}^t e^{-\phi} du \leqslant \tfrac{e^{-\phi}}{2} \leqslant 1. \tag{3.5}$$

Thus $y(t) \geqslant y(n + \frac{1}{2}) - e^{-1} \geqslant \delta - 2 - 2\pi b_{n+1}$ for any $t \in [n + \frac{1}{2}, n + 1]$. Note that $\phi \geqslant 1$ and apply Theorem 2.13 (item **(iv)**) to z using the above inequality to get that

$$z(n+1) \geqslant \min_{u \in [n+\frac{1}{2}, n+1]} y(u) - e^{-1} \geqslant \delta - 2 + 2\pi b_{n+1} - e^{-1} \geqslant \delta - 3 + 2\pi b_{n+1}.$$

And since $\delta - 3 \geqslant 1$, we have shown that $y(n+1)$ and $z(n+1)$ are greater than $1 + 2\pi b_{n+1}$. Furthermore, (3.4) and (3.5) prove that for any $t \in [n, n+1]$,

$$y(t) \geqslant \min(1 + 2\pi b_n, \delta - 2 + 2\pi b_{n+1}) \geqslant 1 + 2\pi b_n.$$

We can thus let $\texttt{fastgen}(\alpha; t) = y(1+t)$ and get the result since $1 + 2\pi b_{n+1} \geqslant a_n$. Finally, $(\alpha_x; t) \mapsto y(t)$ is uniformly-generable by Theorem 2.6 because $\texttt{pereach}$ and g are generable and the initial condition is computable.

The computability of the map $x \mapsto \alpha$ follows from the computability of the map $a \mapsto \alpha$ (Lemma 3.1) and the map $x \mapsto a$. Note that the only condition which $a \in \mathbb{N}^{\mathbb{N}}$ has to satisfy is $a_n \geqslant x_n$. Given a real number represented by its Cauchy sequence, *with a known rate of convergence*, it is trivial to compute an integer upper bound on this number. □

4. Generating a sequence of dyadic rationals

A major part of the proof requires to build a function to approximate arbitrary numbers over intervals $[n, n+1]$. Ideally we would like to build a function that gives x_0 over $[0,1]$, x_1 over $[1,2]$, etc. Before we get there, we solve a somewhat simpler problem by making a few assumptions:

- we only try to approximate dyadic numbers, *i.e.* numbers of the form $m2^{-p}$, and furthermore we only approximate with error 2^{-p-3};
- if a dyadic number has size p, meaning that it can be written as $m2^{-p}$ but not $m'2^{-p+1}$ then it will take a time interval of p units to approximate: $[k, k+p]$ instead of $[k, k+1]$;
- the function will only approximate the dyadics over intervals $[k, k+\frac{1}{2}]$ and not $[k, k+1]$.

This processus is illustrated in Figure 4: given a sequence $d_0, d_1, \dots$ of dyadics, there is a corresponding sequence $a_0, a_1, \dots$ of times such that the function approximates d_k over $[a_k, a_k + \frac{1}{2}]$ within error 2^{-p_k} where p_k is the size of d_k. The theorem contains an explicit formula for a_k that depends on some absolute constant δ.

Figure 4 highlights a feature of $\texttt{dygen}$: it is an almost piecewise constant function. However we only control the values it takes over small intervals $[a_i, a_i + \frac{1}{2}]$, and we have no idea what is the value the rest of the time (even if we know that it is almost piecewise constant).

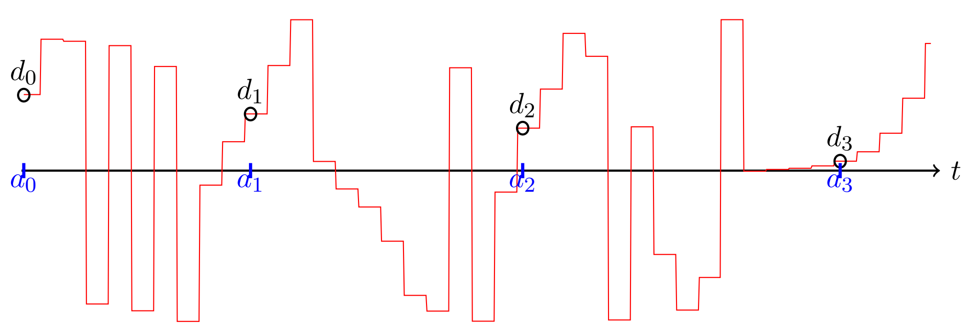

Figure 4: Graph of $\texttt{dygen}$ for $d_0 = 2^{-1}$, $d_1 = 2^{-3} + 2^{-1}$, $d_2 = 2^{-5} + 2^{-2}$ and $d_3 = 2^{-4}$ (other values ignored) assuming that $\delta = 9$. We get that $a_0 = 0$, $a_1 = 10$, $a_2 = 22$, $a_3 = 36$.

Let $\mathbb{D}_p = \{m2^{-p} : m \in \{0, 1, \dots, 2^p - 1\}\}$ and $\mathbb{D} = \bigcup_{n \in \mathbb{N}} \mathbb{D}_p$ denote the set of dyadic rationals in $[0,1)$. For any $q \in \mathbb{D}$, we define its *size* by $\mathfrak{L}(q) = \min\{p \in \mathbb{N} : q \in \mathbb{D}_p\}$.

Theorem 4.1. *There exists $\delta \in \mathbb{N}_{>0}$, $\Gamma \subseteq \mathbb{R}^2$ and a uniformly-generable function* $\mathtt{dygen} : \Gamma \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}$ *such that for any dyadic sequence $q \in \mathbb{D}^{\mathbb{N}}$, there exists $(\alpha, \beta) \in \Gamma$ such that for any $n \in \mathbb{N}$,*

$$|\mathtt{dygen}(\alpha, \beta; t) - q_n| \leqslant 2^{-\mathfrak{L}(q_n)-3} \qquad \text{for any } t \in [a_n, a_n + \tfrac{1}{2}]$$

where $a_n = \sum_{k=0}^{n-1}(\mathfrak{L}(q_k) + \delta)$. Furthermore, $|\mathtt{dygen}(\alpha, \beta; t)| \leqslant 1$ for all α, β and t. In addition, the map $q \mapsto (\alpha, \beta)$ is $(\nu_{\mathbb{Q}}^{\omega}, [\rho, \rho])$-computable.

Lemma 4.2. *For any $q \in \mathbb{D}_p$, there exists $q' \in \mathbb{D}_{p+3}$ such that $|\sin(2q'\pi) - q| \leqslant 2^{-p}$ and $|q'| \leqslant 1$. Furthermore, $x \mapsto \sin(2\pi x)$ is 8-Lipschitz. In addition, the map $q \mapsto q'$ is $(\nu_{\mathbb{Q}}, \nu_{\mathbb{Q}})$-computable.*

Proof. Let $f(x) = \sin(2\pi x)$ for $x \in [0, \frac{1}{4}]$. Clearly f is surjective from $[0, \frac{1}{4}]$ to $[0, 1]$ thus there exists $x' \in [0, \frac{1}{4}]$ such that $f(x') = q$. Furthermore, since $f'(x) = 2\pi\cos(2\pi x)$ and $2\pi \leqslant 8$, f is 8-Lipschitz. Let $q' = \lfloor 2^{p+3}x' \rfloor 2^{-p-3}$, then $q' \in \mathbb{D}_{p+3}$ and $|q' - x'| \leqslant 2^{-p-3}$ by construction. Clearly $|q'| \leqslant 1$, and furthermore,

$$|f(q') - q| \leqslant |f(q') - f(x')| + |f(x') - q| \leqslant 8|q' - x'| + 0 \leqslant 8 \cdot 2^{-p-3} \leqslant 2^{-p}.$$

Note that f is not only surjective from $[0, \frac{1}{4}]$ to $[0, 1]$ but also increasing and 8-Lipschitz. Furthermore, f is (ρ, ρ)-computable thus a simple dichotomy is enough to find a suitable **rational** x'. To conclude, use the fact that the map $x' \mapsto q'$ is $(\nu_{\mathbb{Q}}, \nu_{\mathbb{Q}})$-computable. Note that it is crucial that x' is rational because the floor function is not (ρ, ρ)-computable. □

Proof of Theorem 4.1. Let $\delta = 9$. Consider the function

$$f(\alpha, t) = \sin\left(2\alpha\pi 2^t\right)$$

defined for any $\alpha, t \in \mathbb{R}$. Then f is generable because sin and exp are generable. For all $n \in \mathbb{N}$, note that $q_n \in \mathbb{D}_{\mathfrak{L}(q_n)} \subseteq \mathbb{D}_{\mathfrak{L}(q_n)+\delta-3}$ and apply Lemma 4.2 to q_n to get $q_n' \in \mathbb{D}_{\mathfrak{L}(q_n)+\delta}$ such that

$$|\sin(2q_n'\pi) - q_n| \leqslant 2^{-\mathfrak{L}(q_n)-\delta+3}. \tag{4.1}$$

Now define

$$\alpha_q = \sum_{k=0}^{\infty} q_k' 2^{-a_k} \qquad \text{where } a_k = \sum_{\ell=0}^{k-1}(\mathfrak{L}(q_\ell) + \delta). \tag{4.2}$$

It is not hard to see that α_q is well-defined (*i.e.* the sum converges). Let $n \in \mathbb{N}$, then

$$\begin{aligned} f(\alpha_q, a_n) &= \sin(2\pi\alpha_q 2^{a_n}) \\ &= \sin\left(2\pi\sum_{k=0}^{\infty} q_k' 2^{-a_k+a_n}\right) \\ &= \sin\left(2\pi\sum_{k=0}^{n-1} q_k' 2^{-a_k+a_n} + 2\pi q_n' + 2\pi\sum_{k=n+1}^{\infty} q_k' 2^{-a_k+a_n}\right). \end{aligned}$$

But for any $k \leqslant n - 1$,

$$a_n - a_k = \sum_{\ell=k}^{n-1}(\mathfrak{L}(q_\ell) + \delta) \geqslant \mathfrak{L}(q_k) + \delta$$

and since $q'_k \in \mathbb{D}_{\mathfrak{L}(q_k)+\delta}$ and $a_n - a_k \in \mathbb{N}$, it follows that $q'_k 2^{-a_k+a_n} \in \mathbb{N}$. Consequently,

$$f(\alpha_q, a_n) = \sin\left(2\pi q'_n + 2\pi \sum_{k=n+1}^{\infty} q'_k 2^{-a_k+a_n}\right) = \sin\big(2\pi(q'_n + u)\big)$$

where $u = \sum_{k=n+1}^{\infty} q'_k 2^{-a_k+a_n}$. But for any $k \geqslant n+1$,

$$a_k - a_n = \sum_{\ell=n}^{k-1} (\mathfrak{L}(q_\ell) + \delta) \geqslant \mathfrak{L}(q_n) + \sum_{\ell=n}^{k-1} \delta = \mathfrak{L}(q_n) + \delta(k-n).$$

Consequently,

$$|u| \leqslant \sum_{k=n+1}^{\infty} |q'_k| 2^{-a_k+a_n} \leqslant \sum_{k=n+1}^{\infty} 2^{-\mathfrak{L}(q_n)-\delta(k-n)} \leqslant 2^{-\mathfrak{L}(q_n)} \sum_{k=1}^{\infty} 2^{-\delta k} \leqslant 2^{-\mathfrak{L}(q_n)-\delta+1}.$$

Since $x \mapsto \sin(2\pi x)$ is 8-Lipschitz, it follows that

$$\begin{aligned}
|f(\alpha_q, a_n) - q_n| &= |\sin\big(2\pi(q'_n + u)\big) - q_n| \\
&\leqslant |\sin\big(2\pi(q'_n + u)\big) - \sin(2\pi q'_n)| + |\sin(2q'_n\pi) - q_n| \\
&\leqslant 8|u| + 2^{-\mathfrak{L}(q_n)-\delta+3} && \text{using (4.1)} \\
&\leqslant 8 \cdot 2^{-\mathfrak{L}(q_n)-\delta+1} + 2^{-\mathfrak{L}(q_n)-\delta+3} \\
&\leqslant 2^{-\mathfrak{L}(q_n)-\delta+5}. && (4.3)
\end{aligned}$$

Recall that `round` is the generable rounding function from Lemma 2.9, and `fastgen` the fast growing function from Theorem 3.3. Let $\alpha, \beta, t \in \mathbb{R}$, if $\texttt{fastgen}(\beta; t)$ exists then let

$$\texttt{dygen}(\alpha, \beta; t) = f(\alpha, r(\beta; t))$$

where

$$r(\beta; t) = \texttt{round}\left(t - \tfrac{1}{4}, \texttt{fastgen}(\beta; t) \ln 2, 4\right).$$

Note that `dygen` is uniformly-generable because f, `round` and `fastgen` are uniformly-generable. Apply[10] Theorem 3.3 to get $\beta_q \in \mathbb{R}$ such that for any $n \in \mathbb{N}$ and $t \in \mathbb{R}_{\geqslant 0}$,

$$\texttt{fastgen}(\beta_q; t) \geqslant a_n + \mathfrak{L}(q_n) + \delta \qquad \text{if } t \in [a_n, a_n + 1).$$

Let $n \in \mathbb{N}$ and $t \in [a_n, a_n + \frac{1}{2}]$, then $t - \frac{1}{4} \in \left[a_n - \frac{1}{2} + \frac{1}{\lambda}, a_n + \frac{1}{2} - \frac{1}{\lambda}\right]$ for $\lambda = 4$. Thus we can apply Lemma 2.9 and get that

$$|r(\beta_q, t) - a_n| \leqslant e^{-\texttt{fastgen}(\beta_q;t)\ln 2} = 2^{-\texttt{fastgen}(\beta_q;t)} \leqslant 2^{-a_n - \mathfrak{L}(q_n) - \delta} \leqslant 1. \qquad (4.4)$$

Observe that

$$\left|\tfrac{\partial f}{\partial t}(\alpha_q, t)\right| = 2\pi|\alpha_q| 2^t |\cos(2\alpha_q \pi 2^t)| \leqslant 2\pi 2^t \leqslant 2^{t+3}.$$

Thus for any $t, t' \in \mathbb{R}$,

$$|f(\alpha_q, t) - f(\alpha_q, t')| \leqslant 2^{3+\max(t,t')}|t - t'|.$$

It follows that for any $n \in \mathbb{N}$ and $t \in [a_n, a_n + \frac{1}{2}]$,

$$\begin{aligned}
|\texttt{dygen}(\alpha_q, \beta_q; t) - q_n| &= |f(\alpha_q, r(\beta_q; t)) - q_n| \\
&\leqslant |f(\alpha_q, r(\beta_q; t)) - f(\alpha_q, a_n)| + |f(\alpha_q, a_n) - q_n| \\
&\leqslant 2^{3+\max(a_n, r(\beta_q;t))}|r(\beta_q; t) - a_n| + 2^{-\mathfrak{L}(q_n)-3} && \text{using (4.3)}
\end{aligned}$$

[10]Technically speaking, we apply it to the sequence $x_n = a_k$ if $n = a_k + \mathfrak{L}(q_n) + \delta$, and $x_n = 0$ otherwise.

$$\begin{aligned}
&\leqslant 2^{3+a_n+1}2^{-\mathfrak{L}(q_n)-a_n-\delta}+2^{-\mathfrak{L}(q_n)-\delta+5} && \text{using (4.4)}\\
&\leqslant 2^{-\mathfrak{L}(q_n)-\delta+4}+2^{-\mathfrak{L}(q_n)-\delta+5}\\
&\leqslant 2^{-\mathfrak{L}(q_n)-\delta+6}\\
&\leqslant 2^{-\mathfrak{L}(q_n)-3} && \text{since } \delta-6\geqslant 3.
\end{aligned}$$

To see that the map $q\mapsto(\alpha_q,\beta_q)$ is computable, first note that the map $q_n\mapsto q'_n$ is computable (Lemma 4.2), thus the map $q\mapsto q'$ is $(\nu_{\mathbb{Q}}^{\omega},\nu_{\mathbb{Q}}^{\omega})$-computable. It is clear from (4.2) that $q'\mapsto a$ is also computable. Using a similar argument as above, one can easily see that the partial sums (of the infinite sum) defining α_q in (4.2) form a Cauchy sequence with convergence rate $k\mapsto 2^{-k}$ because $a_k\geqslant k\delta\geqslant k$. Finally, $q\mapsto\beta_q$ is computable by Theorem 3.3. ☐

5. Generating a sequences of bits

We saw in the previous section how to generate a dyadic generator. Unfortunately, we saw that it generates dyadic d_n at times a_n, whereas we would like to get d_n at time n for our approximation. Our approach is to build a signal generator that will be high exactly at times a_n. Each time the signal will be high, the system will copy the value of the dyadic generator to a variable and wait until the next signal. Since the signal is binary, we only need to generate a sequence of bits. Note that this theorem has a different flavour from the dyadic generator: it generates a more restrictive set of values (bits) but does so much better because we have control over the timing and we can approximate the bits with arbitrary precision.

Figure 5 shows what bitgen looks like: it is an almost piecewise constant function such that the value in the interval $[n,n+\frac{1}{2}]$ is almost the n^{th} digit of α.

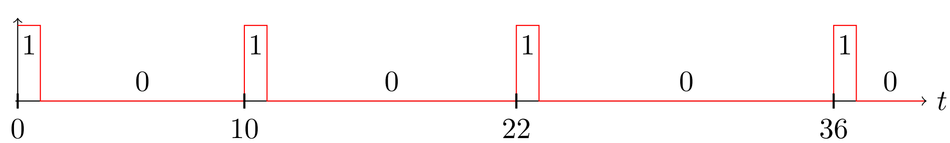

Figure 5: (Ideal) graph of bitgen for $b\in\{0,1\}^{\mathbb{N}}$ where $b_0=b_{10}=b_{22}=b_{36}=1$ and all the other bits are 0.

Remark 5.1. Although it is possible to define bitgen using dygen, it does not, in fact, give a shorter proof but definitely gives a more complicated function.

Theorem 5.2. *There exists* $\Gamma\subseteq\mathbb{R}$ *and a generable function* $\mathtt{bitgen}:\Gamma\times\mathbb{R}^2_{\geqslant 0}\to\mathbb{R}$ *such that for any bit sequence* $b\in\{0,1\}^{\mathbb{N}}$*, there exists* $\alpha_b\in\Gamma$ *such that for any* $\mu\in\mathbb{R}_{\geqslant 0}$*,* $n\in\mathbb{N}$ *and* $t\in[n,n+\frac{1}{2}]$*,*

$$|\mathtt{bitgen}(\alpha_b,\mu,t)-b_n|\leqslant e^{-\mu}.$$

Furthermore, $|\mathtt{bitgen}(\alpha,\mu,t)|\leqslant 1$ *for all* α,μ *and* t*. Finally, the map* $b\mapsto\alpha_b$ *is* $(\nu_{\mathbb{N}}^{\omega},\rho)$*-computable.*

Proof. Consider the function

$$f(\alpha,t)=\sin\left(2\pi\alpha 4^t+\tfrac{4\pi}{3}\right)$$

defined for any $\alpha,t\in\mathbb{R}$. Then f is generable because sin and exp are generable. For any $b\in\{0,1\}^{\mathbb{N}}$, let

$$\alpha_b=\sum_{k=0}^{\infty}2b_k4^{-k-1}. \tag{5.1}$$

Let $b\in\{0,1\}^{\mathbb{N}}$ and $n\in\mathbb{N}$, observe that

$$\begin{aligned}
f(\alpha_b,n)&=\sin\left(2\pi\sum_{k=0}^{\infty}2b_k4^{-k-1}4^n+\tfrac{4\pi}{3}\right)\\
&=\sin\left(2\pi\underbrace{\sum_{k=0}^{n-1}2b_k4^{n-k-1}}_{\in\mathbb{N}}+2\pi 2b_n4^{-1}+2\pi\sum_{k=n+1}^{\infty}2b_k4^{n-k-1}+\tfrac{4\pi}{3}\right)\\
&=\sin\left(\pi b_n+\tfrac{4\pi}{3}+\delta\right)
\end{aligned}$$

where

$$\delta=2\pi\sum_{k=n+1}^{\infty}2b_k4^{n-k-1}=4\pi\sum_{k=0}^{\infty}b_k4^{-k-2}\leqslant 4\pi\sum_{k=0}^{\infty}4^{-k-2}=4\pi\tfrac{4^{-2}}{3}=\frac{\pi}{3}.$$

It follows that if $b_n=0$ then $\pi b_n+\frac{4\pi}{3}+\varepsilon\in\left[\frac{4\pi}{3},\frac{5\pi}{3}\right]$ and if $b_n=1$ then $\pi b_n+\frac{4\pi}{3}+\varepsilon\in\left[\frac{7\pi}{3},\frac{8\pi}{3}\right]$. Thus

$$f(\alpha_b,n)\in\left[-1,-\tfrac{\sqrt{3}}{2}\right]\ \text{ if } b_n=0,\qquad\qquad f(\alpha_b,n)\in\left[\tfrac{\sqrt{3}}{2},1\right]\ \text{ if } b_n=1. \tag{5.2}$$

Furthermore,

$$|\alpha_b|=\sum_{k=0}^{\infty}2b_k4^{-k-1}\leqslant\tfrac{2}{4}\sum_{k=0}^{\infty}4^{-k}\leqslant\tfrac{2}{3}. \tag{5.3}$$

Let $\varepsilon\in\left[-\frac{1}{2},\frac{1}{2}\right]$, then for any $t\in\mathbb{R}$,

$$\begin{aligned}
|f(\alpha_b,t+\varepsilon)-f(\alpha_b,t)|&=\left|\sin\left(2\pi\alpha_b4^{t+\varepsilon}+\tfrac{4\pi}{3}\right)-\sin\left(2\pi\alpha_b4^t+\tfrac{4\pi}{3}\right)\right|\\
&=\left|2\cos(2\pi\alpha_b(4^t+4^{t+\varepsilon})+\tfrac{8\pi}{3})\sin\left(2\pi\alpha_b(4^{t+\varepsilon}-4^t)\right)\right|\\
&\leqslant 2|\sin(2\pi\alpha_b(4^{t+\varepsilon}-4^t))|\\
&\leqslant 4\pi|\alpha_b||4^{t+\varepsilon}-4^t|\\
&\leqslant 4\pi\frac{2}{3}4^t|4^\varepsilon-1| && \text{using (5.3)}\\
&\leqslant\frac{8\pi}{3}4^t2\varepsilon && \text{using that } |\varepsilon|\leqslant\tfrac{1}{2}\\
&\leqslant 4^tB\varepsilon && (5.4)
\end{aligned}$$

for some constant $B>0$. Recall that `round` is the generable rounding function from Lemma 2.9. Let $\alpha,t\in\mathbb{R}$, $\mu\in\mathbb{R}_{\geqslant 0}$ and define

$$g(\alpha,t)=f(\alpha,r(t))\quad\text{where}\quad r(t)=\texttt{round}(t-\tfrac{1}{4},t\ln 4+\ln B,4).$$

Note again that g is generable because f and $\texttt{round}$ are generable. Let $n \in \mathbb{N}$ and $t \in [n, n+\frac{1}{2}]$, then $t - \frac{1}{4} \in \left[n - \frac{1}{4}, n + \frac{3}{4}\right] = \left[n - \frac{1}{4} + \frac{1}{\lambda}, n + \frac{1}{2} - \frac{1}{\lambda}\right]$ for $\lambda = 4$. Thus we can apply Lemma 2.9 and get that

$$|r(t) - n| \leqslant e^{-(n+1)\ln 4 + \ln B} = \tfrac{4^{-n-1}}{B}.$$

It follows using (5.4) that

$$|g(\alpha_b, t) - f(\alpha_b, n)| = |f(\alpha_b, r(t)) - f(\alpha_b, n)| \leqslant 4^t B \tfrac{4^{-n-1}}{B} = \tfrac{1}{4}.$$

And since $\frac{\sqrt{3}}{2} - \frac{1}{4} \geqslant \frac{1}{2}$, we conclude using (5.2) that for any $t \in \left[n, n + \frac{1}{2}\right]$,

$$g(\alpha_b, t) \in \left[-1, -\tfrac{1}{2}\right] \text{ if } b_n = 0, \qquad g(\alpha_b, t) \in \left[\tfrac{1}{2}, 1\right] \text{ if } b_n = 1. \tag{5.5}$$

Finally, let $\alpha, t \in \mathbb{R}$, $\mu \in \mathbb{R}_{\geqslant 0}$ and define

$$\texttt{bitgen}(\alpha, \mu\, t) = \frac{1 + \tanh(2\mu g(\alpha, t))}{2}.$$

Note that $\texttt{bitgen}$ is generable because tanh and g are generable. Let $\mu \in \mathbb{R}_{\geqslant 0}$, $n \in \mathbb{N}$ and $t \in \left[n, n + \frac{1}{2}\right]$. If $b_n = 0$, then it follows from (5.5) that

$$\begin{aligned} g(\alpha_b, n) &\leqslant -\tfrac{1}{2} \\ |\tanh(2\mu g(\alpha_b, n)) - \texttt{sgn}(2\mu g(\alpha_b, n))| &\leqslant e^{-2\mu |g(\alpha_b, n)|} && \text{using Lemma 2.10} \\ |\tanh(2\mu g(\alpha_b, n)) + 1| &\leqslant e^{-\mu} \\ |\texttt{bitgen}(\alpha_b, \mu, t) - b_n| &\leqslant \tfrac{1}{2} e^{-\mu} \leqslant e^{-\mu} && \text{since } b_n = 0. \end{aligned}$$

Similarly, if $b_n = 1$, then

$$\begin{aligned} g(\alpha_b, n) &\geqslant \tfrac{1}{2} \\ |\tanh(2\mu g(\alpha_b, n)) - \texttt{sgn}(2\mu g(\alpha_b, n))| &\leqslant e^{-2\mu |g(\alpha_b, n)|} \\ |\tanh(2\mu g(\alpha_b, n)) - 1| &\leqslant e^{-\mu} \\ |\texttt{bitgen}(\alpha_b, \mu, t) - b_n| &\leqslant \tfrac{1}{2} e^{-\mu} \leqslant e^{-\mu} && \text{since } b_n = 1. \end{aligned}$$

Finally, it is clear from (5.1) that the partial sums are easily computable and form a Cauchy sequence that converges at rate 4^{-k}, thus α_b is computable from b. □

6. Generating an almost piecewise constant function

We have already explained the main intuition of this section in previous sections. Using the dyadic generator and the bit generator as a signal, we can construct a system that "samples" the dyadic at the right time and then holds this value still until the next dyadic. In essence, we just described an almost piecewise constant function. This function still has a limitation: its rate of change is small so it can only approximate slowly changing functions. Figure 6 illustrates how this process works at the high-level.

Theorem 6.1. *There exists an absolute constant $\delta \in \mathbb{N}$, $p \in \mathbb{N}$, $\Gamma \subseteq \mathbb{R}^p$ and a uniformly-generable function $\texttt{pwcgen} : \Gamma \times \mathbb{R}_{\geqslant 0} \to \mathbb{R}$ such that for any dyadic sequence $q \in \mathbb{D}^{\mathbb{N}}$, there exists $\alpha_q \in \Gamma$ such that for any $n \in \mathbb{N}$, putting $a_n = \sum_{k=0}^{n-1}(\delta + \mathfrak{L}(q_k))$, we have that*

$$|\texttt{pwcgen}(\alpha_q; t) - q_n| \leqslant 2^{-\mathfrak{L}(q_n)} \qquad \text{for any } t \in [a_n + \tfrac{1}{2}, a_{n+1}]$$

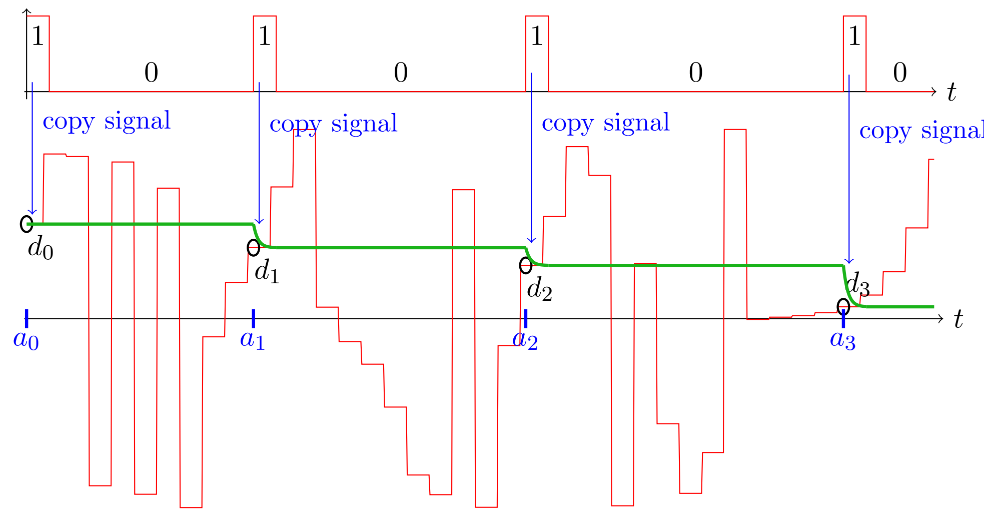

Figure 6: Illustration of the process to generate an almost piecewise-constant function: `bitgen` is used to generate a copy signal, which we synchronise with `dygen` to copy exactly the sequence of dyadic numbers we specified. In-between copies, we ensure that copied value does not change (sample and hold).

and $\texttt{pwcgen}(\alpha_q;t)\in I_n$ *for any* $t\in[a_n,a_n+\frac{1}{2}]$ *where*[11]

$$I_n:=\left[\texttt{pwcgen}(\alpha_q;a_n),\texttt{pwcgen}(\alpha_q;a_n+\tfrac{1}{2})\right]+2^{-\mathfrak{L}(q_n)}[-1,1].$$

Finally, the map $q\mapsto\alpha_q$ *is* $(\nu_{\mathbb{Q}}^{\omega},\rho^p)$*-computable.*

Proof. Apply Theorem 4.1 to get $\delta\in\mathbb{N}$, `dygen` uniformly-generable, and α_q,β_q such that for any $n\in\mathbb{N}$,

$$|\texttt{dygen}(\alpha_q,\beta_q;t)-q_n|\leqslant 2^{-\mathfrak{L}(q_n)-3}\qquad\text{for any } t\in[a_n,a_n+\tfrac{1}{2}]$$

where $a_n=\sum_{k=0}^{n-1}(\mathfrak{L}(q_k)+\delta)$. Let $b\in\{0,1\}^{\mathbb{N}}$ be the bit sequence defined by

$$b_n=\begin{cases}1 & \text{if } n=a_k \text{ for some } k\\ 0 & \text{otherwise.}\end{cases}$$

Apply Theorem 5.2 to get `bitgen` and γ_b such that for any $\mu\in\mathbb{R}_{\geqslant 0}$, $n\in\mathbb{N}$ and $t\in[n,n+\frac{1}{2}]$,

$$|\texttt{bitgen}(\gamma_b,\mu,t)-b_n|\leqslant e^{-\mu}.$$

Let $x_n=\mathfrak{L}(q_n)+3$ and apply Theorem 3.3 to get λ_a and `fastgen` such that for any $n\in\mathbb{N}$,

$$\texttt{fastgen}(\lambda_a;t)\geqslant x_n\qquad\text{for all } t\in[n,n+1].$$

Consider the following system for all $t\in\mathbb{R}_{\geqslant 0}$:

$$y(0)=q_0,\qquad y'(t)=\psi(t)r(t),\qquad r(t)=\texttt{pereach}(t,\phi(t),y(t),g(t))$$

where

$$\phi(t)=t+\texttt{fastgen}(\lambda_a;t),\qquad \psi(t)=2\texttt{bitgen}(\gamma_b,\phi(t)+R(t),t),$$
$$g(t)=\texttt{dygen}(\alpha_q,\beta_q;t),\qquad R(t)=1+r(t)^2.$$

[11] With the convention that $[a,b]=[\min(a,b),\max(a,b)]$ and $I+\alpha J=\{x+\alpha y:x\in I,y\in J\}$.

We omitted the parameters $\alpha, \beta, \gamma, \lambda$ in the functions g and ϕ to make it more readable. It is clear that g and ϕ are uniformly-generable since $\mathtt{bitgen}, \mathtt{fastgen}, \mathtt{dygen}$ are uniformly-generable. It follows that[12] $(q_0, \alpha, \beta, \gamma, \lambda, t) \mapsto y(t)$ is also uniformly-generable by Theorem 2.6. Indeed, the only extra thing we need to check is that the initial condition is a computable function of the parameters, which it is since we just need to extract q_0 from the list of parameters.

We will show the result by induction. Let $n \in \mathbb{N}$ and assume that $|y(a_n) - q_n| \leqslant 2^{-x_n}$. Note that this is trivially satisfied for $n = 0$ since $a_0 = 0$ and thus $y(a_0) = y(0) = q_0$. We will now do the analysis of the behavior of y over $[a_n, a_{n+1}]$ by making a case distinction between $[a_n, a_n + 1]$ and $[a_n + 1, a_{n+1}]$. Note that for all t, $R(t) \geqslant |r(t)| \geqslant 0$.

When $\mathbf{t} \in [\mathbf{a_n}, \mathbf{a_n} + \frac{\mathbf{1}}{\mathbf{2}}]$, we have that

$$|\mathtt{bitgen}(\gamma_b, \mu, t) - b_{a_n}| \leqslant e^{-\mu}$$

but $b_{a_n} = 1$ by definition thus

$$\mathtt{bitgen}(\gamma_b, \phi(t) + R(t), t) \geqslant 1 - e^{-\phi(t)} \geqslant \tfrac{1}{2}$$

since $\phi(t) \geqslant \mathtt{fastgen}(\lambda_a; t) \geqslant 1$. Furthermore,

$$\phi(t) \geqslant \mathtt{fastgen}(\lambda_a; t) \geqslant \mathtt{fastgen}(\lambda; n) \geqslant x_n \geqslant 2.$$

Thus $\psi(t)\phi(t) = 2\mathtt{bitgen}(\gamma_b, \phi(t) + R(t), t)\phi(t) \geqslant x_n \geqslant 2$. Furthermore,

$$|g(t) - q_n| = |\mathtt{dygen}(\alpha_q, \beta_q; t) - q_n| \leqslant 2^{-x_n} \tag{6.1}$$

thus we can apply Theorem 2.13 to get the existence of y and item **(i)** to get that

$$|y(a_n + \tfrac{1}{2}) - q_n| \leqslant 2^{-x_n} + e^{-x_n} \leqslant 2^{-x_n+1}. \tag{6.2}$$

Note that (6.1) implies that

$$q_n - 2^{-x_n} \leqslant g(t) \leqslant q_n + 2^{-x_n}$$

and thus (6.2) proves that

$$y(a_n + \tfrac{1}{2}) - 2^{-x_n+1} \leqslant q_n \leqslant y(a_n + \tfrac{1}{2}) + 2^{-x_n+1}. \tag{6.3}$$

Furthermore, Theorem 2.13 item **(v)** also gives us that

$$\begin{aligned} \min(y(a_n), q_n - 2^{-x}) &\leqslant y(t) \leqslant \max(y(a_n), q_n + 2^{-x_n}) \\ \min(y(a_n), y(a_n + \tfrac{1}{2}) - 3 \cdot 2^{-x_n}) &\leqslant y(t) \leqslant \max(y(a_n), y(a_n + \tfrac{1}{2}) + 3 \cdot 2^{-x_n}) \qquad \text{using (6.3)} \\ y(t) &\in [y(a_n), y(a_n + \tfrac{1}{2})] + [-3 \cdot 2^{-x_n}, 3 \cdot 2^{-x_n}] \\ y(t) &\in [y(a_n), y(a_n + \tfrac{1}{2})] + [-2^{-\mathfrak{L}(q_n)}, 2^{-\mathfrak{L}(q_n)}]. \end{aligned} \tag{6.4}$$

When $\mathbf{t} \in [\mathbf{a_n} + \mathbf{k} + \frac{\mathbf{1}}{\mathbf{2}}, \mathbf{a_n} + \mathbf{k} + \mathbf{1}]$ for $\mathbf{a_n} \leqslant \mathbf{a_n} + \mathbf{k} < \mathbf{a_{n+1}}$, we have that

$$y'(t) = \psi(t)\mathtt{pereach}(t, \phi(t), y(t), g(t))$$

where $|\psi(t)| \leqslant 2$ since $|\mathtt{bitgen}| \leqslant 1$ by Theorem 5.2 and

$$\phi(t) \geqslant t + \mathtt{fastgen}(\lambda_a; t) \geqslant t + \mathtt{fastgen}(\lambda; n) \geqslant t + x_n.$$

[12] Technically, we have to include q_0 in the parameters, even though the sequence q is implicitly encoded in other parameters. This is because the initial condition must be q_0 for the proof to work.

Thus by Theorem 2.13 item **(iii)** we have that

$$|y(t)-y(a_n+k+\tfrac{1}{2})|\leqslant\int_{a_n+k+\frac{1}{2}}^{t}\psi(u)\exp(-\phi(u))du\leqslant\tfrac{1}{2}2e^{-x_n}\int_{a_n+k+\frac{1}{2}}^{t}e^{-u}du.$$

When $\mathbf{t}\in[\mathbf{a_n}+\mathbf{k},\mathbf{a_n}+\mathbf{k}+\frac{\mathbf{1}}{\mathbf{2}}]$ for $\mathbf{a_n}<\mathbf{a_n}+\mathbf{k}<\mathbf{a_{n+1}}$, we have that

$$|\texttt{bitgen}(\gamma_b,\mu,t)-b_{a_n+k}|\leqslant e^{-\mu}$$

but $b_{a_n+k}=0$ by definition thus

$$|\texttt{bitgen}(\gamma_b,\phi(t)+R(t),t)|\leqslant e^{-\phi(t)-R(t)}.$$

Furthermore,

$$\phi(t)\geqslant t+\texttt{fastgen}(\lambda_a;t)\geqslant t+\texttt{fastgen}(\lambda;n)\geqslant t+x_n.$$

Thus

$$|\psi(t)|=2|\texttt{bitgen}(\gamma_b,\phi(t)+R(t),t)|\leqslant 2e^{-t-x_n}e^{-R(t)}\leqslant 2e^{-t-x_n}e^{-|r(t)|}.$$

It follows that

$$y'(t)=\psi(t)r(t)$$

where

$$|\psi(t)r(t)|\leqslant 2e^{-\phi(t)-R(t)}|r(t)|\leqslant 2e^{-t-x_n}.$$

Consequently,

$$|y(t)-y(a_n+k)|\leqslant\int_{a_n+k}^{a_n+k+\frac{1}{2}}|\psi(u)r(u)|du\leqslant e^{-x_n}\int_{a_n+k}^{a_n+k+\frac{1}{2}}e^{-u}du.$$

Putting everything together we get that for all $t\in[a_n+\frac{1}{2},a_{n+1}]$,

$$|y(t)-y(a_n+\tfrac{1}{2})|\leqslant e^{-x_n}\int_{a_n+1}^{t}e^{-u}du\leqslant e^{-x_n}$$

and thus using (6.2), for all $t\in[a_n+\frac{1}{2},a_{n+1}]$,

$$|y(t)-q_n|\leqslant e^{-x_n}+|y(a_n+\tfrac{1}{2})-q_n|\leqslant e^{-x_n}+2^{-x_n+1}\leqslant 2^{-x_n+2}\leqslant 2^{-\mathfrak{L}(q_n)}.$$

Also recall (6.4) that for all $t\in[a_n,a_n+\frac{1}{2}]$,

$$y(t)\in[y(a_n),y(a_n+\tfrac{1}{2})]+[-2^{-\mathfrak{L}(q_n)},2^{-\mathfrak{L}(q_n)}].$$

We have already shown that the map $Y(q_0,\alpha_q,\beta_q,\gamma_q,\lambda_a,t)=y(t)$ is uniformly-generable. Finally, we need to show computability of the map $q\mapsto(\alpha_q,\beta_q,\gamma_b,\lambda_a)$. Computability of α_q and β_q follows from Theorem 4.1. The sequence $(a_n)_n$, and thus $(b_n)_n$, is easily computed from q. It follows that from Theorem 5.2 that γ_b is computable from b, and from Theorem 3.3 that λ_a is computable from a. □

7. Proof of the main theorem

The proof works in several steps. First we show that using an almost constant function, we can approximate functions that are bounded and change very slowly. We then relax all these constraints until we get to the general case. In the following, we only consider total functions over $\mathbb{R}$. See Remark on page 6 for more details.

Definition 7.1 (Universality)**.** Let $I \subseteq \mathbb{R}$ and $\mathcal{C} \subseteq C^0(I) \times C^0(I, \mathbb{R}_{>0})$. We say that the universality property holds for $\mathcal{C}$ if there exists $d \in \mathbb{N}$, $\Gamma \subseteq \mathbb{R}^d$ and a uniformly-generable function $\mathfrak{u} : \Gamma \times I \to \mathbb{R}$ such that for every $(f, \varepsilon) \in \mathcal{C}$, there exists $\alpha \in \Gamma$ such that

$$|\mathfrak{u}(\alpha; t) - f(t)| \leqslant \varepsilon(t) \qquad \text{for all } t \in I.$$

The universality property is said to be *effective* if furthermore the map $(f, \varepsilon) \mapsto \alpha$ is $([\rho \to \rho]^2, \rho^d)$-computable.

Lemma 7.2. *There exists a constant $c > 0$ such that the universality property holds for all (f, ε) on $\mathbb{R}_{\geqslant 0}$ such that for all $t \in \mathbb{R}_{\geqslant 0}$:*

- *ε is decreasing and $-\log_2 \varepsilon(t) \leqslant c' + t$ for some constant c',*
- *$f(t) \in [0, 1]$,*
- *$|f(t) - f(t')| \leqslant c\varepsilon(t + 1)$ for all $t' \in [t, t + 1]$.*

Furthermore, the universality property is effective for this class.

Proof Sketch. This is essentially a application of `pwcgen` with a small twist. Indeed the bound on f guarantees that dyadic rationals are enough. The bound on the rate of change of f guarantees that a single dyadic can provide an approximation for a long enough time. And the bound on ε guarantees that we do not need too many digits for the approximations. □

Proof. Let $c = \frac{1}{8}$. Apply Theorem 6.1 to get $p, \delta \in \mathbb{N}$ and `pwcgen`. Let f and ε be as described in the statement. For any $n \in \mathbb{N}$, let $q_n \in \mathbb{D}$ be such that

$$|f(n) - q_n| \leqslant 2^{-\mathfrak{L}(q_n)} \leqslant c\varepsilon(n + 1). \tag{7.1}$$

Since by assumption, $-\log_2 \varepsilon(n + 1) \leqslant c' + n + 1$, we can always choose q_n so that

$$\mathfrak{L}(q_n) = \lceil c' \rceil + n + 1.$$

Then by Theorem 6.1, there exists $\alpha_q \in \mathbb{R}^p$ such that

$$|\mathtt{pwcgen}(\alpha_q; t) - q_n| \leqslant 2^{-\mathfrak{L}(q_n)} \qquad \text{for any } t \in [a_n + \tfrac{1}{2}, a_{n+1}]$$

and

$$\mathtt{pwcgen}(\alpha_q; t) \in \left[\mathtt{pwcgen}(\alpha_q; a_n), \mathtt{pwcgen}(\alpha_q; a_n + \tfrac{1}{2})\right] \qquad \text{for any } t \in [a_n, a_n + \tfrac{1}{2}] \tag{7.2}$$

where

$$a_n = \sum_{k=0}^{n-1} (\delta + \mathfrak{L}(q_k)) = \sum_{k=0}^{n-1} (\delta + \lceil c' \rceil + n + 1) = n(\delta + \lceil c' \rceil + 1) + \tfrac{1}{2} n(n - 1).$$

Introduce the function

$$\xi(t) = t(\delta + \lceil c' \rceil + 1) + \tfrac{1}{2} t(t - 1)$$

so that $a_n = \xi(n)$ and note that ξ is increasing and generable. Let $t \in [\xi^{-1}(a_n + \frac{1}{2}), \xi^{-1}(a_{n+1})]$, since ξ is increasing, so is ξ^{-1} and

$$n = \xi^{-1}(a_n) \leqslant \xi^{-1}(a_n + \tfrac{1}{2}) \leqslant t \leqslant \xi^{-1}(a_{n+1}) = n + 1.$$

So in particular,

$$|f(n)-f(t)|\leqslant c\varepsilon(n+1)$$

by the assumption on f, since $t\in[n,n+1]$. It follows that

$$\begin{aligned}|\mathtt{pwcgen}(\alpha_q;\xi(t))-f(t)| &\leqslant |\mathtt{pwcgen}(\alpha_q;\xi(t))-q_n|\\ &\quad+|q_n-f(n)|+|f(n)-f(t)|\\ &\leqslant 2^{-\mathfrak{L}(q_n)}+2^{-\mathfrak{L}(q_n)}+c\varepsilon(n+1) &&\text{since } \xi(t)\in[a_n+\tfrac{1}{2},a_{n+1}]\\ &\leqslant 3c\varepsilon(n+1)\\ &\leqslant 3c\varepsilon(t) &&\text{since } \varepsilon \text{ is decreasing}\\ &\leqslant \varepsilon(t) &&\text{since } 3c\leqslant 1.\end{aligned}$$

So in particular, in implies that for all $n\in\mathbb{N}$,

$$|\mathtt{pwcgen}(\alpha_q;a_n+\tfrac{1}{2})-f(\xi^{-1}(a_n+\tfrac{1}{2}))|\leqslant 3c\varepsilon(n+1) \tag{7.3}$$

and

$$|\mathtt{pwcgen}(\alpha_q;a_{n+1})-f(n+1)|\leqslant 3c\varepsilon(n+2). \tag{7.4}$$

Let $t\in[\xi^{-1}(a_{n+1}),\xi^{-1}(a_{n+1}+\frac{1}{2})]$, it follows using (7.2) that there exists $\lambda\in[0,1]$ such that

$$\mathtt{pwcgen}(\alpha_q;\xi(t))=\lambda\mathtt{pwcgen}(\alpha_q;a_{n+1})+(1-\lambda)\mathtt{pwcgen}(\alpha_q;a_{n+1}+\tfrac{1}{2}).$$

We also have that

$$n+1=\xi^{-1}(a_{n+1})\leqslant t\leqslant\xi^{-1}(a_{n+1}+\tfrac{1}{2})\leqslant\xi^{-1}(a_{n+2})=n+2.$$

Thus

$$\begin{aligned}|\mathtt{pwcgen}(\alpha_q;\xi(t))-f(t)| &\leqslant \lambda|\mathtt{pwcgen}(\alpha_q;a_{n+1})-f(t)|\\ &\quad+(1-\lambda)|\mathtt{pwcgen}(\alpha_q;a_{n+1}+\tfrac{1}{2})-f(t)|\\ &\leqslant |\mathtt{pwcgen}(\alpha_q;a_{n+1})-f(n+1)|+|f(n+1)-f(t)|\\ &\quad+|\mathtt{pwcgen}(\alpha_q;a_{n+1}+\tfrac{1}{2})-f(\xi^{-1}(a_{n+1}+\tfrac{1}{2}))|\\ &\quad+|f(\xi^{-1}(a_{n+1}+\tfrac{1}{2}))-f(t)|\\ &\leqslant 3c\varepsilon(n+2)+|f(n+1)-f(t)| &&\text{using (7.3)}\\ &\quad+3c\varepsilon(n+2) &&\text{using (7.4)}\\ &\quad+|f(\xi^{-1}(a_{n+1}+\tfrac{1}{2}))-f(t)|\\ &\leqslant 3c\varepsilon(n+2)+c\varepsilon(n+2) &&\text{since } t\in[n+1,n+2]\\ &\quad+3c\varepsilon(n+2)\\ &\quad+c\varepsilon(t) &&\text{since } \xi^{-1}(a_{n+1}+\tfrac{1}{2})\in[t,t+1]\\ &\leqslant 7c\varepsilon(n+2)+c\varepsilon(t)\\ &\leqslant 8c\varepsilon(t) &&\text{since } \varepsilon \text{ decreasing}\\ &\leqslant \varepsilon(t) &&\text{since } 8c\leqslant 1.\end{aligned}$$

Putting everything together, we can get that

$$|\mathtt{pwcgen}(\alpha_q;\xi(t))-f(t)|\leqslant\varepsilon(t)\qquad\text{for all } t\geqslant\xi^{-1}(a_0+\tfrac{1}{2}).$$

But note that $\xi^{-1}(a_0+\frac{1}{2})\leqslant\xi^{-1}(a_1)=1$ so we have that

$$|\mathtt{pwcgen}(\alpha_q;\xi(t))-f(t)|\leqslant\varepsilon(t)\qquad\text{for all } t\geqslant 1$$

and note that $(\alpha;t)\mapsto \texttt{pwcgen}(\alpha;\xi(t))$ is uniformly-generable.

Note that this is not exactly the claimed result since it is only true for $t\geqslant 1$ instead of $t\geqslant 0$ but this can remedied for with proper shifting. Indeed, consider the operator

$$(Sf)(t)=f(\max(t-1,0)).$$

We claim that if (f,ε) satisfies the assumption of the Lemma, then so does $(Sf,S\varepsilon)$ and

$$\begin{aligned}
&|\texttt{pwcgen}(\alpha;\xi(t))-(Sf)(t)|\leqslant(S\varepsilon)(t) && \text{for all } t\geqslant 1\\
&|\texttt{pwcgen}(\alpha;\xi(t))-f(t-1)|\leqslant\varepsilon(t-1) && \text{for all } t\geqslant 1\\
&|\texttt{pwcgen}(\alpha;\xi(t+1))-f(t)|\leqslant\varepsilon(t) && \text{for all } t\geqslant 0.
\end{aligned}$$

We need to show computability of the map $(f,\varepsilon)\mapsto\alpha_q$. By Theorem 6.1, it is enough to show computability of $(f,\varepsilon)\mapsto q$. Since the continuous function evaluation map is computable (for the representation we use), the maps $(f,n)\mapsto f(n)$ and $(\varepsilon,n)\mapsto c\varepsilon(n+1)$ are $([[\rho\to\rho],\nu_{\mathbb{N}}],\rho)$-computable. It follows that for every n, we can compute an integer p_n such that $2^{-p_n}\leqslant c\varepsilon(n+1)$. Indeed, $c\varepsilon(n+1)>0$ thus such a p_n exists and any Cauchy sequence for $c\varepsilon(n+1)$ is eventually positive; therefore it suffices to compute rational approximations of $c\varepsilon(n+1)$ with increasing precision until we get a positive one, from which we can compute p_n. Given such a p_n, one can compute a dyadic approximation q_n of $f(n)$ with precision p_n. This sequence $(q_n)_n$ then satisfies (7.1). □

Lemma 7.3. *The universality property holds for all (f,ε) on $\mathbb{R}_{\geqslant 0}$ such that f and ε are differentiable, ε is decreasing and $f(t)\in[0,1]$ for all $t\in\mathbb{R}_{\geqslant 0}$. Furthermore, the universality property is effective for this class if we are given a representation of f' and ε' as well*[13]

Proof Sketch. Consider $F=f\circ h^{-1}$ and $E=\varepsilon\circ h^{-1}$ where h is a fast-growing function like $\texttt{fastgen}$. Then the faster h grows, the slower E and F change and thus we can apply Lemma 7.2 to (F,E). We recover an approximation of f from the approximation of F. □

Proof. Apply Lemma 7.2 to get $c>0$ and $\mathfrak{u}$ uniformly-generable. For every $n\in\mathbb{N}$, let

$$a_n=\max\left(\frac{\max_{u\in[0,n+2]}|f'(u)|}{c\varepsilon(n)},\frac{-\min_{u\in[0,n+1]}\varepsilon'(u)}{\ln(2)\varepsilon(n)}\right)-1.$$

Check that a_n is increasing because ε is decreasing. Then apply Theorem 3.3 to get α_a. Recall that $\texttt{fastgen}(\alpha_a;\cdot)$ is positive, thus we can let

$$g(\alpha;t)=\int_0^t 1+\texttt{fastgen}(\alpha;u)du,\qquad h_\alpha(t)=g(\alpha,t).$$

Clearly g is uniformly-generable since $\texttt{fastgen}$ is uniformly-generable. Since $\texttt{fastgen}$ is increasing, h_α is increasing. Furthermore,

$$h_{\alpha_a}(n+1)\geqslant\int_n^{1+n}\texttt{fastgen}(\alpha_a;u)du\geqslant\int_n^{1+n}\texttt{fastgen}(\alpha_a;n)du=\texttt{fastgen}(\alpha_a;n)\geqslant a_n.$$

Thus $h_{\alpha_a}(n)\to+\infty$ as $m\to+\infty$. This implies that h_{α_a} is bijective from $\mathbb{R}_{\geqslant 0}$ to $\mathbb{R}_{\geqslant 0}$. Note that since h_{α_a} is increasing then $h_{\alpha_a}^{-1}$ is also increasing. Also since $h_{\alpha_a}(t)\geqslant t$ then $h_{\alpha_a}^{-1}(t)\leqslant t$ for all $t\in\mathbb{R}_{\geqslant 0}$. Let f,ε be as described in the statement. For any $\xi\in\mathbb{R}_{\geqslant 0}$, let

$$F(\xi)=f(h_{\alpha_a}^{-1}(\xi)),\qquad E(\xi)=\varepsilon(h_{\alpha_a}^{-1}(\xi)).$$

[13]In other words, the map $(f,f',\varepsilon)\mapsto\alpha$ is $([\rho\to\rho]^4,\rho^d])-$computable. This is necessary because one can build some computable f such that f' is not computable [Wei00].

Then for any $t \in \mathbb{R}_{\geqslant 0}$,

$$\begin{aligned}
F'(h_{\alpha_a}(t)) &= (h_{\alpha_a}^{-1})'(h_{\alpha_a}(t)) f'(h_{\alpha_a}^{-1}(h_{\alpha_a}(t))) \\
&= \tfrac{1}{h'_{\alpha_a}(t)} f'(t) && \text{since } h'(h^{-1})' \circ h = 1 \\
&= \tfrac{1}{1+\mathtt{fastgen}(\alpha_a;t)} f'(t).
\end{aligned}$$

Also note that since $h'_{\alpha_a}(t) = 1 + \mathtt{fastgen}(\alpha_a; t) \geqslant 1$, then $(h_{\alpha_a}^{-1})'(t) \leqslant 1$ and thus $h_{\alpha_a}^{-1}$ is 1-Lipschitz. Let $\xi \in \mathbb{R}_{\geqslant 0}$ and $\xi' \in [\xi, \xi+1]$. Write $\xi = h_{\alpha_a}(t)$ and $\xi' = h_{\alpha_a}(t')$, then

$$\begin{aligned}
\tfrac{|F(\xi)-F(\xi')|}{E(\xi+1)} &\leqslant \tfrac{|\xi-\xi'|\max_{u\in[\xi,\xi']}|F'(\xi)|}{E(\xi+1)} \\
&\leqslant \tfrac{\max_{u\in[t,t']}|F'(h_{\alpha_a}(u))|}{E(\xi+1)} && \text{since } |\xi-\xi'| \leqslant 1 \\
&= \max_{u\in[t,t']} \tfrac{|f'(u)|}{E(\xi+1)(1+\mathtt{fastgen}(\alpha_a;u))} \\
&\leqslant \tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\varepsilon(h_{\alpha_a}^{-1}(h_{\alpha_a}(t)+1))(1+\mathtt{fastgen}(\alpha_a;t))} && \text{since } \mathtt{fastgen} \text{ is increasing.}
\end{aligned}$$

but since $h_{\alpha_a}^{-1}$ is 1-Lipschitz and increasing, $h_{\alpha_a}^{-1}(h_{\alpha_a}(t)+1) \leqslant h_{\alpha_a}^{-1}(h_{\alpha_a}(t)) + 1 = t+1$,

$$\begin{aligned}
&\leqslant \tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\varepsilon(t+1)(1+\mathtt{fastgen}(\alpha_a;t))} && \text{since } \varepsilon \text{ is decreasing} \\
&\leqslant \tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\varepsilon(t+1)(1+\mathtt{fastgen}(\alpha_a;\lfloor t\rfloor))} && \text{since } \mathtt{fastgen} \text{ is decreasing} \\
&\leqslant \tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\varepsilon(t+1)(1+a_{\lfloor t\rfloor}))} \\
&\leqslant \tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\varepsilon(t+1)\frac{\max_{u\in[0,\lfloor t\rfloor+2]}|f'(u)|}{c\varepsilon(\lfloor t\rfloor)}} \\
&= c\tfrac{\max_{u\in[t,t+1]}|f'(u)|}{\max_{u\in[0,\lfloor t\rfloor+2]}|f'(u)|}\tfrac{\varepsilon(\lfloor t\rfloor)}{\varepsilon(t+1)} \\
&\leqslant c && \text{since } \varepsilon \text{ is decreasing.}
\end{aligned}$$

Similarly,

$$\begin{aligned}
E'(h_{\alpha_a}(t)) &= (h_{\alpha_a}^{-1})'(h_{\alpha_a}(t))\varepsilon'(h_{\alpha_a}^{-1}(h_{\alpha_a}(t))) \\
&= \tfrac{1}{h'_{\alpha_a}(t)}\varepsilon'(t) \\
&= \tfrac{1}{1+\mathtt{fastgen}(\alpha_a;t)}\varepsilon'(t).
\end{aligned}$$

Thus

$$\begin{aligned}
-\log_2 E(\xi) &= -\tfrac{1}{\ln 2}\int_0^\xi \frac{E'(e)}{E(e)}de - \tfrac{1}{2}\log_2 E(0) \\
&= -\tfrac{1}{2}\log_2\varepsilon(0) + \tfrac{1}{\ln 2}\int_0^\xi \frac{-E'(e)}{E(e)}de \\
&\leqslant -\tfrac{1}{2}\log_2\varepsilon(0) + \tfrac{\xi}{\ln 2}\sup_{e\in[0,\xi]}\frac{-E'(e)}{E(e)} \\
&\leqslant -\tfrac{1}{2}\log_2\varepsilon(0) + \tfrac{\xi}{\ln 2}\sup_{t\in[0,h_{\alpha_a}^{-1}(\xi)]}\frac{-E'(h_{\alpha_a}(t))}{E(h_{\alpha_a}(t))}
\end{aligned}$$

$$\begin{aligned}
&\leqslant -\tfrac{1}{2}\log_2 \varepsilon(0) + \tfrac{\xi}{\ln 2} \sup_{t\in[0,h_{\alpha_a}^{-1}(\xi)]} \frac{-\varepsilon'(t)}{\varepsilon(t)(1+\mathtt{fastgen}(\alpha_a;t))} \\
&\leqslant -\tfrac{1}{2}\log_2 \varepsilon(0) + \tfrac{\xi}{\ln 2} \sup_{t\in[0,\xi]} \frac{-\varepsilon'(t)}{\varepsilon(t)(1+\mathtt{fastgen}(\alpha_a;t))} \qquad \text{using } h_{\alpha_a}^{-1}(\xi)\leqslant\xi \\
&\leqslant -\tfrac{1}{2}\log_2 \varepsilon(0) + \tfrac{\xi}{\ln 2} \sup_{t\in[0,\xi]} \frac{-\varepsilon'(t)}{\varepsilon(t)(1+a_{\lfloor t\rfloor})} \\
&\leqslant -\tfrac{1}{2}\log_2 \varepsilon(0) + \tfrac{\xi}{\ln 2} \sup_{t\in[0,\xi]} \frac{-\varepsilon'(t)}{\varepsilon(t)\frac{-\min_{u\in[0,\lfloor t\rfloor+1]}|\varepsilon'(u)|}{\ln(2)\varepsilon(\lfloor t\rfloor)}} \\
&= -\tfrac{1}{2}\log_2 \varepsilon(0) + \xi \sup_{t\in[0,\xi]} \frac{\varepsilon(\lfloor t\rfloor)}{\varepsilon(t)}\frac{\varepsilon'(t)}{\min_{u\in[0,\lfloor t\rfloor+1]}\varepsilon'(u)} \\
&\leqslant -\tfrac{1}{2}\log_2 \varepsilon(0) + \xi \qquad \text{since } \varepsilon \text{ is decreasing and } \varepsilon' \text{ is negative}
\end{aligned}$$

and thus

$$-\log_2 E(\xi) \leqslant c' + \xi$$

for some constant c'. Therefore we can apply Lemma 7.2 to (F, E) and get $\beta_{E,F} \in \mathbb{R}^p$ such that

$$|\mathfrak{u}(\beta_{E,F};\xi) - F(\xi)| \leqslant E(\xi) \qquad \text{for all } \xi \in \mathbb{R}_{\geqslant 0}.$$

For any α, β, t, let

$$\bar{\mathfrak{u}}(\alpha,\beta;t) = \mathfrak{u}(\beta; g(\alpha,t)).$$

Clearly $\bar{\mathfrak{u}}$ is uniformly-generable because $\mathfrak{u}$ and g are uniformly-generable. Then for any $t \in \mathbb{R}_{\geqslant 0}$, recall that $g(\alpha_a;t) = h_{\alpha_a}(t)$ and thus

$$|\bar{\mathfrak{u}}(\alpha_a,\beta_{E,F};t) - f(t)| = |\mathfrak{u}(\beta;h_{\alpha_a}(t)) - F(h_{\alpha_a}(t))| \leqslant E(h_{\alpha_a}(t)) = \varepsilon(t).$$

To show the effectiveness of the property, it suffices to show that $(f, f', \varepsilon) \mapsto (a, E, F)$ is computable. Indeed, α_a and $\beta_{E,F}$ are computable from a, E, F by Lemma 7.2 and Theorem 3.3. Given a, the maps E and F are computable from f and ε because h_{α_a} is computable and increasing, thus its inverse is computable. Finally, to show computability of a, notice that to define a suitable value for each a_n, it is enough to compute an upper bound on the maximum of continuous functions – defined from $f, f', \varepsilon, \varepsilon'$ – over compact intervals, which is a computable operation. □

Lemma 7.4. *The universality property holds for all (f, ε) on $\mathbb{R}_{\geqslant 0}$ such that f and ε are differentiable and ε is decreasing. Furthermore, the universality property is effective for this class if we are given a representation of f' and ε' as well.*

Proof. Apply Lemma 7.3 to get $p \in \mathbb{N}$ and $\mathfrak{u}$ uniformly-generable. Let $a \in \mathbb{N}^{\mathbb{N}}$ be an increasing sequence, and apply Theorem 3.3 to get α_a. Recall that $\mathtt{fastgen}(\alpha_a;\cdot)$ is positive and increasing. Let f, ε be as described in the statement. For any $t \in \mathbb{R}_{\geqslant 0}$, let

$$F(t) = \tfrac{1}{2} + \tfrac{f(t)}{1+\mathtt{fastgen}(\alpha_a;t)}, \qquad E(t) = \tfrac{\varepsilon(t)}{1+\mathtt{fastgen}(\alpha_a;t)}.$$

Then for any $n \in \mathbb{N}$ and $t \in [n, n+1]$, we have that

$$|F(t) - \tfrac{1}{2}| = \tfrac{f(t)}{1+\mathtt{fastgen}(\alpha_a;t)} \leqslant \tfrac{|f(t)|}{1+a_n}.$$

Thus we can choose $a_n = 2\max_{u\in[t,t+1]}|f(u)|$ and get that $|F(t)-\frac{1}{2}|\leqslant\frac{1}{2}$ for all $t\in\mathbb{R}_{\geqslant 0}$, and thus $F(t)\in[0,1]$. Furthermore, F is differentiable and E is decreasing because ε is decreasing and `fastgen` increasing. Apply Lemma 7.3 to (F,ε) to get $\beta_F\in\mathbb{R}^p$ such that

$$|\mathfrak{u}(\beta_F;t)-F(t)|\leqslant E(t)\qquad\text{for all } t\in\mathbb{R}_{\geqslant 0}.$$

For any α,β,t, let

$$\bar{\mathfrak{u}}(\alpha,\beta;t)=(1+\mathtt{fastgen}(\alpha;t))(\mathfrak{u}(\beta;t)-\tfrac{1}{2}).$$

Clearly $\bar{\mathfrak{u}}$ is uniformly-generable because $\mathfrak{u}$ and `fastgen` are uniformly-generable. Then for any $t\in\mathbb{R}_{\geqslant 0}$,

$$\begin{aligned}|\bar{\mathfrak{u}}(\alpha_a,\beta_F;t)-f(t)| &= \left|(1+\mathtt{fastgen}(\alpha_a;t))\left(\mathfrak{u}(\beta;h_{\alpha_a}(t))-\tfrac{1}{2}-\tfrac{f(t)}{1+\mathtt{fastgen}(\alpha_a;t)}\right)\right|\\ &= (1+\mathtt{fastgen}(\alpha_a;t))|(\mathfrak{u}(\beta;h_{\alpha_a}(t))-F(t))|\\ &\leqslant (1+\mathtt{fastgen}(\alpha_a;t))E(t)\\ &\leqslant \varepsilon(t).\end{aligned}$$

The effectiveness of α_a and β_F comes from previous lemmas and boils down again to compute an upper bound on the maximum of a continuous function. □

Lemma 7.5. *The universality property holds for all continuous (f,ε) on $\mathbb{R}_{\geqslant 0}$. Furthermore, the universality property is effective for this class.*

Proof. Apply Lemma 7.4 to get $p\in\mathbb{N}$ and $\mathfrak{u}$ uniformly-generable. Let $f\in C^0(\mathbb{R}_{\geqslant 0},\mathbb{R})$ and $\varepsilon\in C^0(\mathbb{R}_{\geqslant 0},\mathbb{R}_{>0})$. Then there exists $\tilde{f}\in C^1(\mathbb{R}_{\geqslant 0},\mathbb{R})$ and a decreasing $\tilde{\varepsilon}\in C^1(\mathbb{R}_{\geqslant 0},\mathbb{R}_{>0})$ such that

$$|\tilde{f}(t)-f(t)|\leqslant\tilde{\varepsilon}(t)\leqslant\tfrac{\varepsilon(t)}{2}.\tag{7.5}$$

We can then apply Lemma 7.4 to $(\tilde{f},\tilde{\varepsilon})$ to get $\alpha_{\tilde{f}}\in\mathbb{R}^p$ such that

$$|\mathfrak{u}(\alpha_{\tilde{f}};t)-\tilde{f}(t)|\leqslant\tilde{\varepsilon}(t)\qquad\text{for all } t\in\mathbb{R}_{\geqslant 0}.$$

But then for any $t\in\mathbb{R}_{\geqslant 0}$,

$$\begin{aligned}|\mathfrak{u}(\alpha_{\tilde{f}};t)-f(t)| &\leqslant |\mathfrak{u}(\alpha_{\tilde{f}};t)-\tilde{f}(t)|+|\tilde{f}(t)-f(t)|\\ &\leqslant \tilde{\varepsilon}(t)+\tilde{\varepsilon}(t)\\ &\leqslant \varepsilon(t).\end{aligned}$$

To show the computability of $\alpha_{\tilde{f}}$, it suffices to show computability of $\tilde{f}$ and ε, and their derivatives, from f and ε, and apply Lemma 7.4. Is it not hard to find C^1 functions satisfying (7.5). For example, one can proceed over all intervals $[n,n+1]$ and then use C^1 pasting. Over a compact interval, one can use an effective variant of Stone-Weierstrass theorem. □

Lemma 7.6. *There exists a uniformly-generable function $\mathfrak{u}$ such that for all $f\in C^0(\mathbb{R}_{\geqslant 0},\mathbb{R})$, $\varepsilon\in C^0(\mathbb{R},\mathbb{R}_{>0})$ and $\delta>0$, there exists α such that*

- *$|\mathfrak{u}(\alpha;t)-f(t)|\leqslant\varepsilon(t)$ for all $t\geqslant 0$,*
- *$|\mathfrak{u}(\alpha;t)|\leqslant|f(-t)|+\varepsilon(t)$ for all $t\in[-\delta,0]$,*
- *$|\mathfrak{u}(\alpha;t)|\leqslant\varepsilon(t)$ for all $t\leqslant-\delta$.*

Furthermore, the map $(f,\varepsilon,\delta)\mapsto\alpha$ is $([[\rho\to\rho]^2,\rho],\rho)-$computable.

Proof. Apply Lemma 7.5 to get $\mathfrak{u}$. Note that $t \mapsto f(\sqrt{t})$, $t \mapsto \varepsilon(\sqrt{(t)})$ and max are continuous, so there exists α such that

$$|\mathfrak{u}(\alpha;t) - f(\sqrt{t})| \leqslant \tfrac{1}{2}\min(\varepsilon(\sqrt{t}), \varepsilon(-\sqrt{t})) \qquad \text{for all } t \geqslant 0.$$

For any α, t define

$$\mathfrak{U}(\alpha,\beta,\delta;t) = s(t)\mathfrak{u}(\alpha;t^2), \qquad s(t) = \tfrac{1}{2} + \tfrac{1}{2}\tanh(A(t)(\tfrac{\delta}{2}+t)), \qquad A(t) = \mathtt{fastgen}(\beta;t).$$

Let a_n be a sequence such that for all $n \in \mathbb{N}$,

$$a_n \geqslant \tfrac{2}{\delta}\left(\sup_{t\in[n,n+1]}|f(t)| + \sup_{t\in[-n-1,n+1]} -\log\varepsilon(t)\right).$$

Then there exists β_a such that for all $n \in \mathbb{N}$,

$$\mathtt{fastgen}(\beta;t) \geqslant a_n \qquad \text{for all } t \geqslant n.$$

Let $t \geqslant 0$, then

$$\begin{aligned}
A(t) &= \mathtt{fastgen}(\beta;t)\\
&\geqslant a_{\lfloor t\rfloor}\\
&\geqslant \tfrac{2}{\delta}\left(\sup_{u\in[\lfloor t\rfloor,\lceil t\rceil]}|f(u)| + \sup_{u\in[-\lceil t\rceil,\lceil t\rceil]} -\log\varepsilon(u)\right)\\
&\geqslant \tfrac{2}{\delta}\left(|f(t)| - \log\varepsilon(t)\right)\\
(\tfrac{\delta}{2}+t)A(t) &\geqslant |f(t)| - \log\varepsilon(t)\\
|1-\tanh((\tfrac{\delta}{2}+t)A(t))| &\leqslant e^{-|f(t)|+\log\varepsilon(t)}\\
|1-s(t)| &\leqslant \tfrac{\varepsilon(t)}{2}e^{-|f(t)|}.
\end{aligned}$$

It follows that

$$\begin{aligned}
|f(t) - \mathfrak{U}(\alpha,\beta,\delta;t)| &\leqslant |f(t) - \mathfrak{u}(\alpha;t^2)| + |\mathfrak{u}(\alpha;t^2)(1-s(t))|\\
&\leqslant \tfrac{1}{2}\varepsilon(t) + |\mathfrak{u}(\alpha;t^2)||1-s(t)|\\
&\leqslant \tfrac{1}{2}\varepsilon(t) + |\mathfrak{u}(\alpha;t^2)||1-s(t)|\\
&\leqslant \tfrac{1}{2}\varepsilon(t) + (|f(t)| + \tfrac{1}{2}\varepsilon(t))\tfrac{\varepsilon(t)}{2}e^{-|f(t)|}\\
&\leqslant \varepsilon(t) && \text{using } e^{-x}x \leqslant 1.
\end{aligned}$$

Let $t \in [-\delta, 0]$, then $|s(t)| \leqslant 1$ thus

$$|\mathfrak{U}(\alpha,\beta,\delta;t)| \leqslant |\mathfrak{u}(\alpha;t^2)| \leqslant |f(-t)| + \varepsilon(t).$$

Let $t \leqslant -\delta$, then with a similar argument as above

$$|s(t)| \leqslant \tfrac{\varepsilon(t)}{2}e^{-|f(-t)|}.$$

It follows that

$$|\mathfrak{U}(\alpha,\beta,\delta;t)| \leqslant |\mathfrak{u}(\alpha;t^2)||s(t)| \leqslant (|f(-t)| + \varepsilon(t))\tfrac{\varepsilon(t)}{2}e^{-|f(-t)|} \leqslant \varepsilon(t).$$

The effectiveness of α comes from Lemma 7.5 and the fact the map $(f,t) \mapsto f(\sqrt{t})$ is $([[\rho\to\rho],\rho],\rho)$−computable since $\sqrt{\cdot}$ is (ρ,ρ)−computable. To show the computability of β, it suffices to show the computability of a_n, which boils down to computing an upper bound for f and ε over compact intervals. The effectiveness for δ is trivial since it is the identity mapping (δ is given unmodified to $\mathfrak{U}$). $\square$

Lemma 7.7. *The universality property holds for all continuous (f, ε) on $\mathbb{R}$. Furthermore, the universality property is effective for this class.*

Proof. Let $c = \frac{1}{4}$. Apply Lemma 7.6 to get $\mathfrak{u}$. Then there exists α such that

- $|\mathfrak{u}(\alpha; t) - f(t)| \leqslant c\varepsilon(t)$ for all $t \geqslant 0$,
- $|\mathfrak{u}(\alpha; t)| \leqslant |f(-t)| + c\varepsilon(t)$ for all $t \in [-1, 0]$,
- $|\mathfrak{u}(\alpha; t)| \leqslant c\varepsilon(t)$ for all $t \leqslant -1$.

For all $t \in \mathbb{R}$, let

$$g(t) = f(-t) - \mathfrak{u}(\alpha; -t).$$

Since this is a continuous function, we can apply the lemma again to get α' such that

- $|\mathfrak{u}(\alpha'; t) - g(t)| \leqslant c\varepsilon(t)$ for all $t \geqslant 0$,
- $|\mathfrak{u}(\alpha'; t)| \leqslant |g(-t)| + c\varepsilon(t)$ for all $t \in [-1, 0]$,
- $|\mathfrak{u}(\alpha'; t)| \leqslant c\varepsilon(t)$ for all $t \leqslant -1$.

For all α, α', t let

$$\mathfrak{U}(\alpha, \alpha'; t) = \mathfrak{u}(\alpha; u) + \mathfrak{u}(\alpha'; -t).$$

We claim that $\mathfrak{U}$ satisfies the theorem:

- If $t \geqslant 1$ then $-t \leqslant -1$ thus
$$|\mathfrak{U}(\alpha, \alpha'; t) - f(t)| \leqslant |\mathfrak{u}(\alpha; t) - f(t)| + |\mathfrak{u}(\alpha'; -t)| \leqslant c\varepsilon(t) + c\varepsilon(t) \leqslant \varepsilon(t).$$
- If $0 \leqslant t \leqslant 1$ then $-1 \leqslant -t \leqslant 0$ thus
$$\begin{aligned} |\mathfrak{U}(\alpha, \alpha'; t) - f(t)| &\leqslant |\mathfrak{u}(\alpha; t) - f(t)| + |\mathfrak{u}(\alpha'; -t)| \\ &\leqslant c\varepsilon(t) + |g(-t)| + c\varepsilon(t) \\ &\leqslant c\varepsilon(t) + |g(-t)| + c\varepsilon(t) \\ &\leqslant c\varepsilon(t) + |f(t) - \mathfrak{u}(\alpha; t)| + c\varepsilon(t) \\ &\leqslant 3c\varepsilon(t) \leqslant \varepsilon(t). \end{aligned}$$
- If $t \leqslant 0$ then $-t \geqslant 0$ and thus
$$\begin{aligned} |\mathfrak{U}(\alpha, \alpha'; t) - f(t)| &= |\mathfrak{u}(\alpha'; t) + \mathfrak{u}(\alpha; t) - f(t)| \\ &= |\mathfrak{u}(\alpha'; -t) - g(-t)| \\ &\leqslant c\varepsilon(t). \end{aligned}$$

The computability of α follows directly from the previous lemma. $\square$

We can now show the main theorem.

Proof of Theorem 1.3. This is mostly rewriting but we do it full for completeness.

Apply Lemma 7.7 to get a uniformly-generable function $\mathfrak{u} : \Gamma \times \mathbb{R} \to \mathbb{R}$. By definition of $\mathfrak{u}$, there exists an integer d, a polynomial matrix q with coefficients in $\mathbb{K}$, $t_0 \in \mathbb{K}$ and a computable function $y_0 : \Gamma \to \mathbb{R}^d$ such that for all $\alpha \in \Gamma$ there exists $y_\alpha : \mathbb{R} \to \mathbb{R}^d$ such that

- $y_\alpha(t_0) = y_0(\alpha)$ and $y_\alpha'(t) = q(y_\alpha(t))$ for all $t \in \mathbb{R}$,
- $(y_\alpha(t))_1 = \mathfrak{u}(\alpha; t)$ for all $t \in \mathbb{R}$.

Note that q is a polynomial that does not depend on α but potentially has coefficients in $\mathbb{K}$ that are not rational. We can eliminate those as explained in Remark 2.3. Now we get that for all continuous functions f, ε, there exists $\alpha \in \Gamma$ such that $|f(t) - \mathfrak{u}(\alpha, t)| \leqslant \varepsilon(t)$. Therefore if we consider the unique solution to $z(0) = y(\alpha; 0)$ and $z' = q(z)$ then $z(t) = y_\alpha(t)$ and we have the result. Note that we have not used the initial condition $z(t_0) = y_0(\alpha)$ because we want $t_0 = 0$ in the statement of the theorem. Furthermore, the initial condition $y(\alpha; 0)$ is computable from f and ε because $y(\alpha; 0)$ is computable from t_0 and $y_0(\alpha)$ by Proposition 2.7, $t_0 \in \mathbb{K}$ is computable, y_0 is computable and α is computable from f, ε by Lemma 7.7. □

REFERENCES

[Ban75] Steven B Bank. Some results on analytic and meromorphic solutions of algebraic differential equations. *Advances in Mathematics*, 15(1):41 – 62, 1975.

[BBV37] N. M. Basu, Satyendra N. Bose, and Tirukkannapuram. Vijayaraghavan. A simple example for a theorem of vijayaraghavan. *Journal of the London Mathematical Society*, s1-12(4):250–252, 1937.

[BC08] Olivier Bournez and Manuel L. Campagnolo. *New Computational Paradigms. Changing Conceptions of What is Computable*, chapter A Survey on Continuous Time Computations, pages 383–423. Springer-Verlag, New York, 2008.

[BCGH07] Olivier Bournez, Manuel L. Campagnolo, Daniel S. Graça, and Emmanuel Hainry. Polynomial differential equations compute all real computable functions on computable compact intervals. *Journal of Complexity*, 23(3):317–335, June 2007.

[BGP16a] Olivier Bournez, Daniel S. Graça, and Amaury Pouly. Polynomial Time corresponds to Solutions of Polynomial Ordinary Differential Equations of Polynomial Length. The General Purpose Analog Computer and Computable Analysis are two efficiently equivalent models of computations. In *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPIcs*, pages 109:1–109:15. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

[BGP16b] Olivier Bournez, Daniel Graça, and Amaury Pouly. Computing with polynomial ordinary differential equations. *Journal of Complexity*, 36:106 – 140, 2016.

[BGP17] Olivier Bournez, Daniel Graça, and Amaury Pouly. On the functions generated by the general purpose analog computer. *Information and Computation*, 257:34 – 57, 2017.

[BHW08] Vasco Brattka, Peter Hertling, and Klaus Weihrauch. *New Computational Paradigms. Changing Conceptions of What is Computable*, chapter A tutorial on computable analysis. Springer-Verlag, New York, 2008.

[Bor99] Emile Borel. Mémoire sur les séries divergentes. *Annales Scientifiques de l'Ecole Normale Supérieure*, 16:9–136, 1899.

[Bos86] Michael Boshernitzan. Universal formulae and universal differential equations. *Annals of mathematics*, 124(2):273–291, 1986.

[BP17] Olivier Bournez and Amaury Pouly. A universal ordinary differential equation. In *International Colloquium on Automata Language Programming, ICALP'2017*, 2017.

[Bra05] Vasco Brattka. Computability & complexity in analysis: Tutorial. `http://www.cca-net.de/vasco/cca/tutorial.pdf`, 2005.

[Bri02] Keith Briggs. Another universal differential equation. *arXiv preprint math/0211142*, 2002.

[Bus31] Vannevar Bush. The differential analyzer. A new machine for solving differential equations. *J. Franklin Inst.*, 212:447–488, 1931.

[CJ16] Etienne Couturier and Nicolas Jacquet. Construction of a universal ordinary differential equation $\mathcal{C}^\infty$ of order 3. *arXiv preprint arXiv:1610.09148*, 2016.

[CPSW05] David C. Carothers, G. Edgar Parker, James S. Sochacki, and Paul G. Warne. Some properties of solutions to polynomial systems of differential equations. *Electron. J. Diff. Eqns.*, 2005(40), April 2005.

[Duf81] Richard J Duffin. Rubel's universal differential equation. *Proceedings of the National Academy of Sciences*, 78(8):4661–4662, 1981.

[GBC09] Daniel S. Graça, Jorge Buescu, and Manuel L. Campagnolo. Computational bounds on polynomial differential equations. *Appl. Math. Comput.*, 215(4):1375–1385, 2009.

[GC03] Daniel S. Graça and José Félix Costa. Analog computers and recursive functions over the reals. *Journal of Complexity*, 19(5):644–664, 2003.

[GCB08] Daniel S. Graça, Manuel L. Campagnolo, and Jorge Buescu. Computability with polynomial differential equations. *Adv. Appl. Math.*, 40(3):330–349, 2008.

[Har12] Godfrey H. Hardy. Some results concerning the behaviour at infinity of a real and continuous solution of an algebraic differential equation of the first order. *Proceedings of the London Mathematical Society*, 2(1):451–468, 1912.

[Kap55] Wilfred Kaplan. Approximation by entire functions. *Michigan Math. J.*, 3(1):43–52, 1955.

[KM99] P. Koiran and Cristopher Moore. Closed-form analytic maps in one and two dimensions can simulate universal Turing machines. *Theoret. Comput. Sci.*, 210(1):217–223, 1999.

[KTZ18] Akitoshi Kawamura, Holger Thies, and Martin Ziegler. Average-Case Polynomial-Time Computability of Hamiltonian Dynamics. In Igor Potapov, Paul Spirakis, and James Worrell, editors, *43rd International Symposium on Mathematical Foundations of Computer Science (MFCS 2018)*, volume 117 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 30:1–30:17, Dagstuhl, Germany, 2018. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.

[LR] Richard Jay Lipton and Ken W. Regan. The amazing zeta code. Post on Blog "Gödel's Lost Letter and P=NP", `https://rjlipton.wordpress.com/2012/12/04/the-amazing-zeta-code/`, December 4, 2012.

[Moo90] Cristopher Moore. Unpredictability and undecidability in dynamical systems. *Physical Review Letters*, 64(20):2354–2357, May 1990.

[PE74] Marian Boykan Pour-El. Abstract computability and its relations to the general purpose analog computer. *Trans. Amer. Math. Soc.*, 199:1–28, 1974.

[PG16] Amaury Pouly and Daniel S Graça. Computational complexity of solving polynomial differential equations over unbounded domains. *Theoretical Computer Science*, 626:67–82, 2016.

[Rub81] Lee A. Rubel. A universal differential equation. *Bulletin of the American Mathematical Society*, 4(3):345–349, May 1981.

[Sha41] C. E. Shannon. Mathematical theory of the differential analyser. *Journal of Mathematics and Physics MIT*, 20:337–354, 1941.

[Thi18] Holger Thies. *Uniform computational complexity of ordinary differential equations with applications to dynamical systems and exact real arithmetic*. PhD thesis, University of Tokyo, Graduate School of Arts and Sciences, 2018.

[Vij32] Tirukkannapuram Vijayaraghavan. Sur la croissance des fonctions définies par les équations différentielles. *CR Acad. Sci. Paris*, 194:827–829, 1932.

[Wei00] Klaus Weihrauch. *Computable Analysis: an Introduction.* Springer, 2000.