A Domain Theoretic Account of Picard's Theorem

A. Edalat[1] and D. Pattinson[2]

[1] Department of Computing, Imperial College London, UK
[2] Institut für Informatik, LMU München, Germany

Abstract. We present a domain-theoretic version of Picard's theorem for solving classical initial value problems in $\mathbb{R}^n$. For the case of vector fields that satisfy a Lipschitz condition, we construct an iterative algorithm that gives two sequences of piecewise linear maps with rational coefficients, which converge, respectively from below and above, exponentially fast to the unique solution of the initial value problem. We provide a detailed analysis of the speed of convergence and the complexity of computing the iterates. The algorithm uses proper data types based on rational arithmetic, where no rounding of real numbers is required. Thus, we obtain an implementation framework to solve initial value problems, which is sound and, in contrast to techniques based on interval analysis, also complete: the unique solution can be actually computed within any degree of required accuracy.

1 Introduction

We consider the initial value problem (IVP) given by the system of differential equations

$$\dot{y}_i(x) = v_i(y_1, \ldots, y_n), \quad y_i(0) = 0 \qquad (i = 1, \ldots, n) \tag{1}$$

where the vector field $v : O \to \mathbb{R}^n$ is continuous in a neighbourhood $O \subseteq \mathbb{R}^n$ of the origin, and we look for a differentiable function $y = (y_1, \ldots, y_n) : [-a, a] \to \mathbb{R}^n$, defined in a neighbourhood of $0 \in \mathbb{R}$, which satisfies (1). By a theorem of Peano there is always a solution [9, page 19]. Uniqueness of the solution is guaranteed, by Picard's theorem, if v satisfies a Lipschitz condition. The question of computability and the complexity of the initial value problem has been studied in different contexts in computable analysis [12,3,8,14,19,17,6].

On the algorithmic and more practical side, standard numerical packages for solving IVP's try to compute an approximation to a solution with a specified degree of accuracy. Although these packages are usually robust, their methods are not guaranteed to be correct and it is easy to find examples where they output inaccurate results [13].

Interval analysis [16] provides a method to give upper and lower bounds for the unique solution in the Lipschitz case with a prescribed tolerance, and has been developed and implemented for analytic vector fields [18,1]. In this approach, arithmetic operations are performed on intervals, and outward rounding is applied if the resulting interval endpoints are not machine representable. While this strategy guarantees soundness, i.e. containment of the exact result in the computed interval, one has in general no control over the rounding, which can produce unduly large intervals. As a consequence, for an implementation of the framework for solving IVP based on interval analysis, one cannot in general guarantee completeness, that is, actual convergence to the solution. For the same reason, one has no control over the speed of convergence.

Domain theory [4] presents an alternative technique, based on proper data types, to produce a provably correct solution with any given degree of accuracy. Using the domain of Scott continuous interval valued functions on a compact interval, we define here a domain theoretic Picard operator, whose least fixed point contains any solution of the IVP. When the vector field is Lipschitz, the solution is unique and we construct an iterative algorithm that gives two sequences of piecewise linear maps with rational coefficients, which converge, respectively from below and above, exponentially fast to the unique solution of the initial value problem. Since the data types for representing the piecewise linear maps with rational coefficients are directly representable on a digital computer, no rounding of real numbers is required. As a consequence, the implementation of the domain theoretic approach is also complete, that is, we can guarantee the convergence of the approximating iterates to the solution of the IVP also for the implementation. To our knowledge, this property is not present in any other approach to validated solutions of differential equations. Furthermore, as a result of the data types we use, we can give estimates for the speed of convergence of the approximating iterates, which are still valid for an actual implementation of our algorithm.

This simplifies the earlier treatment [10], which used a domain for C^1 functions [11] and, at each stage of iteration, required a new approximation of the derivative of the solution. The new treatment is much more similar to the classical theorem in that it gives rise, in the Lipschitz case, to fast convergence of the approximations to the solution.

We discuss two different bases to represent approximations to the solutions of the IVP, namely the piecewise linear and the piecewise constant functions with rational coefficients. Using piecewise linear functions, there is no need to compute rectangular enclosures of the solution, and we therefore avoid the wrapping effect, a well known phenomenon in interval analysis. This comes at the expense of an increase in the size of the representation of the approximations to the solution. Using the base consisting of piecewise constant functions, we show that the order of the speed of convergence to the solution remains unchanged, while the space complexity for the representation of the iterates is much reduced.

A prototypical implementation using the GNU multi precision library [2] shows that the resulting algorithms are actually feasible in practice, and we plan to refine the implementation and compare it in scope and performance with existing interval analysis packages like AWA [1]. Of course we have to bear in mind that floating point arithmetic used by interval software is executed on highly optimised processors, whereas the rational arithmetic needed for our implementation is performed by software.

2 Preliminaries and Notation

For the remainder of the paper, we fix a continuous vector field

$$v = (v_1, \dots, v_n) : [-K, K]^n \to [-M, M]^n$$

which is defined in a compact rectangle containing the origin and consider the IVP given by Equation (1). Note that any continuous function on a compact rectangle is bounded, hence we can assume, without loss of generality, that v takes values in $[-M, M]^n$.

We construct solutions $y : [-a, a] \to \mathbb{R}^n$ of Equation (1) where $a > 0$ satisfies $aM \leq K$. This will guarantee that the expression $v(y)$ is well defined, since M is a bound for the derivative of y. We consider the n-dimensional Euclidean space $\mathbb{R}^n$ equipped with the maximum norm $\|x\| = \max\{|x_1|, \ldots, |x_n|\}$, as this simplifies dealing with the Lipschitz conditions, which we introduce later. Approximations of real numbers live in the interval domain

$$\mathbb{IR} = \{[a, b] \mid a, b \in \mathbb{R}, a \leq b\} \cup \{\mathbb{R}\} \text{ with } [a, b] \sqsubseteq [c, d] \Leftrightarrow [c, d] \subseteq [a, b]$$

ordered by reverse inclusion; the way below relation is given by $[a, b] \ll [c, d]$ iff $[c, d] \subseteq (a, b)$. For $n \geq 1$, the domain $\mathbb{IR}^n$ is isomorphic to the domain of n-dimensional rectangles $\{\alpha_1 \times \cdots \times \alpha_n \mid \alpha_i \in \mathbb{IR} \text{ for all } 1 \leq i \leq n\}$, and we do not distinguish between these two presentations. For a rectangle $R \subseteq \mathbb{R}^n$, the subset $\{S \in \mathbb{IR}^n \mid S \subseteq R \text{ is a rectangle}\}$ of rectangles contained in R is a sub-domain of $\mathbb{IR}^n$, which is denoted by $\mathrm{I}R$. The powers $\mathbb{IR}^n$ of the interval domain and the sub-domain $\mathrm{I}R$, for a rectangle R, are continuous Scott domains. If $\alpha^-, \alpha^+ \in \mathbb{R}^n$ with $\alpha_i^- \leq \alpha_i^+$ for all $1 \leq i \leq n$, we write $[\alpha^-, \alpha^+]$ for the rectangle $[\alpha_1^-, \alpha_1^+] \times \cdots \times [\alpha_n^-, \alpha_n^+]$. Similarly, if $f : X \to \mathbb{IR}^n$ is a function, we write $f = [f^-, f^+]$ if $f(x) = [f^-(x), f^+(x)]$ for all $x \in X$.

The link between ordinary and interval valued function is provided by the notion of *extension*. If $R \subseteq \mathbb{R}^n$ is a rectangle, we say that $F : \mathrm{I}R \to \mathbb{IR}^n$ is an extension of $f : R \to \mathbb{R}^n$ if

$$F(\{x_1\}, \ldots, \{x_n\}) = \{f(x_1, \ldots, x_n)\}$$

for all $x \in R$. Note that every continuous function $f : R \to \mathbb{R}^n$ has a *canonical* extension F defined by

$$F = (F_1, \ldots, F_n) : \mathrm{I}R \to \mathbb{IR}^n \text{ with } F_i(S) = [\inf_{x \in S} f_i(x), \sup_{x \in S} f_i(x)],$$

where $S \in \mathrm{I}R$ is a rectangle, which is maximal in the set of interval valued functions extending f. It is easy to see that F is continuous wrt. the Scott topology on $\mathrm{I}R$ and $\mathbb{IR}^n$ if f is continuous wrt. the Euclidean topology.

We consider the following spaces for approximating the vector field and the solutions to the IVP:

- $\mathcal{S} = [-a, a] \to \mathrm{I}[-K, K]^n$, the set of continuous functions wrt. the Euclidean topology on $[-a, a]$ and the Scott topology on $\mathrm{I}[-K, K]^n$
- $\mathcal{V} = \mathrm{I}[-K, K]^n \to \mathrm{I}[-M, M]^n$, the set of continuous functions wrt. the Scott topology on $\mathrm{I}[-K, K]^n$ and $\mathrm{I}[-M, M]^n$.

In order to measure the speed of convergence, as well as for technical convenience in the formulation of some of our results, we introduce the following notation, where X is an arbitrary set:

- For a rectangle $\alpha = [\alpha^-, \alpha^+]$, $w(\alpha) = \|\alpha^+ - \alpha^-\|$ denotes the *width* of α. Similarly, if $f : X \to \mathbb{IR}^n$ is a function, $w(f) = \sup_{x \in X} w(f(x))$ is the *width* of f.
- For $u, u' : X \to \mathbb{IR}$ with $u'(x) \sqsubseteq u(x)$ for all $x \in X$, the *width of u' relative to u* is defined as $w_u(u') = \sup_{x \in X} \|u'^+(x) - u^+(x) + u^-(x) - u'^-(x)\|$.

Considering u' as approximation to u, the relative width $w_u(u')$ can be understood as measuring the quality of the approximation.

3 The Picard Operator in Domain Theory

In the classical proof of Picard's theorem on the existence and uniqueness of the solution of the initial value problem (1) one defines an integral operator on $C^0[-a,a]$ by

$$y \mapsto \lambda x. \int_0^x v(y(t))dt$$

(with the integral understood componentwise), which can be shown to be a contraction for sufficiently small a provided v satisfies a Lipschitz condition [15]. An application of Banach's theorem then yields a solution of the initial value problem. We now define the domain-theoretic Picard operator for arbitrary continuous vector fields $u : \mathrm{I}[-K,K]^n \to \mathrm{I}[-M,M]^n$ and focus on the special case where u is an extension of a classical function later. As in the classical proof, the Picard operator is an integral operator, and we therefore introduce the integral of interval-valued functions.

Definition 1. *Suppose $f = [f^-, f^+] : [-a,a] \to \mathbb{IR}$ is Scott continuous. For $x \in [-a,a]$ we let*

$$\int_0^x f(t)dt = [\int_0^x f^{-\sigma}(t)dt, \int_0^x f^{\sigma}(t)dt]$$

where $\sigma = \mathrm{sgn}(x)$ is the sign of x. If $f = (f_1, \ldots, f_n) : [-a,a] \to \mathbb{IR}^n$, we let $\int_0^x f(t)dt = (\int_0^x f_1(t)dt, \ldots, \int_0^x f_n(t)dt)$.

Note that, if we integrate in the positive x-direction, then f^- contributes to the lower function associated with the integral of f and f^+ contributes to the upper function. If we integrate in the negative x-direction, the roles of f^- and f^+ are swapped. The following shows that our definition is meaningful:

Lemma 1. *Suppose $f : [-a,a] \to \mathbb{IR}$ is Scott continuous.*

(i) f^- and f^+ are Lebesgue integrable
(ii) $\int_0^x f(t)dt \in \mathbb{IR}$ for all $x \in [-a,a]$.

Proof. For Scott continuous f, the functions f^-, f^+ are lower (resp. upper) semi continuous, hence Lebesgue integrable. If $\sigma = \mathrm{sgn}(x)$, then $\sigma f^{-\sigma} \le \sigma f^{\sigma}$ and $\int_0^x f^{-\sigma}(t)dt \le \int_0^x f^{\sigma}(t)dt$ follows from the definition of the ordinary integral. Finally $\int_0^x f(t)dt$ is either compact or the whole of $\mathbb{R}$, since $f^+(t) = \infty$ iff $f^-(t) = -\infty$, for all $t \in [-a,a]$.

The following lemma shows that integration is compatible with taking suprema.

Lemma 2. *Let $f : [-a,a] \to \mathbb{IR}^n$.*

(i) The function $\lambda x. \int_0^x f(t)dt$ is Scott continuous.
(ii) The function $\int : f \mapsto \lambda x. \int_0^x f(t)dt$ is Scott continuous.

Proof. We assume $n = 1$ from which the general case follows. If $g(x) = \int_0^x f(t)dt$, then g^-, g^+ are continuous, hence g is Scott continuous. The second statement follows from the monotone convergence theorem.

The domain theoretic Picard operator can now be defined as follows:

Definition 2. *Suppose* $u \in \mathcal{V}$. *The* domain theoretic Picard operator $P_u : \mathcal{S} \to \mathcal{S}$ *is defined by* $P_u(y) = \lambda x. \int_0^x u(y(t))dt$.

Lemma 3. P_u *is well defined and continuous.*

Proof. That $P_u(y) \in \mathcal{S}$ follows from our assumption $aM \leq K$. Lemma 2 shows that $P_u(y)$, for $y \in \mathcal{S}$, and P_u itself are continuous.

In the classical proof of Picard's theorem, one constructs solutions of IVP's as fixpoint of the (classical) Picard operator. The domain theoretic proof replaces Banach's theorem with Klenee's theorem in the construction of a fixed point of the (domain theoretic) Picard operator. Unlike the classical case, where one chooses an arbitrary initial approximation, we choose the function $y_0 = \lambda t.[-K, K]^n$ with the least possible information as initial approximation.

Theorem 4. *Let* $u \in \mathcal{V}$ *and* $y_{k+1} = P_u(y_k)$. *Then* $y = \bigsqcup_{k \in \mathbb{N}} y_k$ *satisfies* $P_u(y) = y$.

Proof. Follows immediately from the Kleene's Theorem, see e.g. [4, Theorem 2.1.19].

The bridge between the solution of the domain-theoretic fixpoint equation and the classical initial value problem is established in the following proposition, where $\mathrm{I}f : [-a, a] \to \mathrm{I}[-K, K]^n$ denotes the function $\lambda x.\{f(x)\}$, for $f : [-a, a] \to [-K, K]^n$.

Proposition 5. *Suppose* u *is an extension of* v *and* y *is the least fixpoint of* P_u.

(i) If $f : [-a, a] \to \mathrm{I}[-K, K]^n$ *solves (1) then* $\mathrm{I}f \sqsubseteq y$.
(ii) If y *has width* 0, *then* $y^- = y^+$ *solves (1).*

Proof. For the first statement, note that $\mathrm{I}f$ is a fixed point of P_u and y is the least such. The second statement follows from the fundamental theorem of calculus; note that $y^- = y^+$ implies the continuity of both.

The previous proposition can be read as a soundness result: Every solution of the IVP is contained in the least fixpoint of the domain theoretic Picard operator.

4 The Lipschitz Case

We can ensure the uniqueness of the solution of the IVP by requiring that the vector field satisfies an interval version of the Lipschitz property. Recall that for metric spaces (M, d) and (M', d'), a function $f : M \to M'$ is Lipschitz, if there is $L \geq 0$ such that $d'(f(x), f(z)) \leq L \cdot d(x, z)$ for all $x, z \in M$. The following definition translates this property into an interval setting.

Definition 3 (Lipschitz Condition). *Suppose* $u : \mathrm{I}[-K, K]^n \to \mathrm{I}[-M, M]^n$. *Then* u *is interval Lipschitz if there is some* $L \geq 0$ *such that* $w(u(\alpha)) \leq L \cdot w(\alpha)$ *for all* $\alpha \in \mathrm{I}[-K, K]^n$. *In this case,* L *is called a* Lipschitz constant *for* u.

The following Proposition describes the relationship between the classical notion and its interval version.

Proposition 6. *For $v : [-K, K]^n \to [-M, M]^n$, the following are equivalent:*

(i) v is Lipschitz
(ii) The canonical extension of v satisfies an interval Lipschitz condition
(iii) v has an interval Lipschitz extension.

Note that every interval Lipschitz function induces a total and continuous classical function.

Corollary 7. *Suppose u is interval Lipschitz. Then $w(u(\alpha)) = 0$ whenever $w(\alpha) = 0$, and the induced real valued function $\bar{u}$, given by $\bar{u}(x) = z$ iff $u(\{x\}) = \{z\}$ is continuous.*

In order to guarantee that the sequence of approximations to the solution of the IVP does converge to a width-zero function, we make the following assumption.

For the remainder of the paper, u denotes an extension of v, which satisfies an interval Lipschitz condition with Lipschitz constant L such that $aL < 1$. For later use, we fix $c \in \mathbb{R}$ with $aL < c < 1$.

In case this assumption is not valid, i.e. $aL > 1$, we pick $a' < a$ such that $a'L < 1$ and divide the interval $[-a, a]$ into subintervals of length $< a'$. Replacing a by a' allows us to compute solutions on each subinterval. As we will show in the full version, we can use a glueing process to obtain a solution defined on the whole of $[-a, a]$; this is as in the classical theory [9, page 13].

Assuming the Lipschitz condition, we have the following estimate, which guarantees that the least fixed point of P_u is of width 0:

Lemma 8. $w(P_u(y)) \leq aL \cdot w(y)$ *for all* $y \in \mathcal{S}$.

The above estimate allows us to show that – in the Lipschitz case – the least fixed point of the domain-theoretic Picard operator has width 0, i.e. solves the initial value problem, as shown in Proposition 5.

Proposition 9. *Let $y_{k+1} = P_u(y_k)$ for $k \in \mathbb{N}$. Then $w(y_k) \leq c^k w(y_0)$. In particular, $y = \bigsqcup_{k \in \mathbb{N}} y_k$ satisfies $P_u(y) = y$ and $w(y) = 0$.*

Proof. Follows immediately from $aL < c < 1$ by induction.

In order to be able to compute the integrals, we now consider approximations to u; the basic idea is that every continuous vector field can be approximated by a sequence of step functions (i.e. functions taking only finitely many values), which allows us to compute the integrals involved in calculating the approximations to the solution effectively. The key property which enables us to use approximations also to the vector field is the continuity of the mapping $u \mapsto P_u$.

Lemma 10. *The map $P : \mathcal{V} \to \mathcal{S} \to \mathcal{S}$, $u \mapsto P_u$, is continuous.*

Proof. Follows from continuity of u and the monotone convergence theorem.

This continuity property allows us to compute solutions to the classical initial value problem by means of a converging sequence of approximations of u.

Theorem 11. *Suppose* $u = \bigsqcup_{k\in\mathbb{N}} u_k$ *and* $y_{k+1} = P_{u_k}(y_k)$ *for* $k \in \mathbb{N}$. *Then* $y = \bigsqcup_{k\in\mathbb{N}} y_k$ *satisfies* $y = P_u(y)$ *and* $w(y) = 0$.

Proof. Follows from Theorem 4 and continuity of $u \mapsto P_u$ by the interchange-of-suprema law (see e.g. [4, Proposition 2.1.12]).

We have seen that the Lipschitz condition on the vector field ensures that the approximations of the solution converge exponentially fast (Proposition 9). In presence of approximations of the vector field, the speed of convergence will also depend on how fast the vector field is approximated. The following estimate allows to describe the speed of convergence of the iterates if the vector field is approximated by an increasing chain of vector fields.

Lemma 12. *Suppose* $u' \sqsubseteq u$ *and* $y \in \mathcal{S}$. *Then* $w(P_{u'}(y)) \leq aL \cdot w(y) + a \cdot w_u(u')$.

As a corollary we deduce that the approximations converge exponentially fast, if the approximations of the vector field do so too.

Proposition 13. *Suppose* $u = \bigsqcup_{k\in\mathbb{N}} u_k$ *and* $y_{k+1} = P_{u_k}(y_k)$. *Then* $w(y_k) \leq c^k \cdot w(y_0)$ *provided* $w_u(u_k) \leq c^k \cdot 2M(c - aL)$.

Given a representation of u in terms of step functions, Theorem 11 gives rise to an algorithm for computing the solution of the initial value problem. Our next goal is to show that this algorithm can be restricted to bases of the respective domains, showing that it can be implemented without loss of accuracy. We then give an estimate of the algebraic complexity of the algorithm.

5 An Implementation Framework for Solving IVP's

We now show that the algorithm contained in Theorem 11 is indeed implementable by showing that the computations can be carried out in the bases of the domains. In fact, we demonstrate that every increasing chain of (interval valued) vector fields $(u_k)_{k\in\mathbb{N}}$, where each u_k is a base element of $\mathcal{V}$, gives rise to a sequence of base elements of $\mathcal{S}$, which approximate the solution and converge to it.

In view of the algorithm contained in Theorem 11, we consider simple step functions as base of $\mathcal{V}$ and piecewise linear function as base of $\mathcal{S}$. Note that in this setup, the domain-theoretic Picard operator computes integrals of piecewise constant functions, hence produces piecewise linear functions.

We begin by introducing the bases which we are going to work with.

Definition 4. *Let* $D \subseteq \mathbb{R}$ *and assume that* $-a = a_0 < \cdots < a_k = a$ *with* $a_0, \ldots, a_k \in D$, $\beta_0, \ldots, \beta_k \in \mathrm{I}[-K, K]^n_D$ *and* $\gamma_1, \ldots, \gamma_k \in \mathrm{I}[-M, M]^n_D$, *where* R_D *denotes the set of rectangles, which are contained in* R *and whose endpoints lie in* D. *We consider the following classes of functions:*

(i) The class $\mathcal{S}_D^L$ of piecewise D-linear functions $[-a,a] \to \mathrm{I}[-K,K]^n$,

$$f = (a_0, \ldots, a_k) \searrow^L (\beta_0, \ldots, \beta_k)$$

where $f(x)^{\pm} = \beta_{j-1}^{\pm} + \frac{x-a_{j-1}}{a_j-a_{j-1}}(\beta_j^{\pm} - \beta_{j-1}^{\pm})$ for $x \in [a_{j-1}, a_j]$. Every component of a D-linear function is piecewise linear and takes values in D at $a_0, a_1 \ldots, a_k$.

(ii) The set $\mathcal{S}_D^C$ of piecewise D-constant functions $[-a,a] \to \mathrm{I}[-K,K]^n$,

$$f = (a_0, \ldots, a_k) \searrow^C (\beta_1, \ldots, \beta_n), x \mapsto \begin{cases} \beta_i & x \in [a_{i-1}, a_i]^o \\ \beta_{i-1} \sqcap \beta_i & x = a_i \text{ and } 1 < i < k \end{cases}$$

where $\sqcap$ denotes the greatest lower bound. The components of a D-constant function assume constant values in D, which only change at $a_0, a_1, \ldots, a_k$.

(iii) The set $\mathcal{V}_D$ of finite sups of step functions $\mathrm{I}[-K,K]^n \to \mathrm{I}[-M,M]^n$,

$$f = \bigsqcup_{1 \leq j \leq k} \beta_j \searrow \gamma_j : x \mapsto \bigsqcup \{\gamma_j \mid 1 \leq j \leq k, \beta_j \ll x\}.$$

(iv) For any f as above, we put $\mathcal{N}(f) = k$ and call it the complexity of representation *of f.*

Since we will not consider different representations for the same functions, we allow ourselves to blur the distinction between a function and its representation as step function. Note that any computable vector field u can be approximated by a sequence of basis elements $(u_k)_{k \in \mathbb{N}}$ in $\mathcal{V}_{\mathbb{Q}}$, and such approximating sequences can be constructed from a library of elementary functions.

If D is dense in $\mathbb{R}$, it is well known that the sets defined above are bases of their respective superspaces:

Proposition 14. *Suppose $D \subseteq \mathbb{R}$ is dense and $-a, a \in D$.*

(i) $\mathcal{V}_D$ is a base of $\mathcal{V}$.
(ii) $\mathcal{S}_D^C$ and $\mathcal{S}_D^L$ are bases of $\mathcal{S}$.

We can now show that the Picard operator P_u associated with a simple step function u restricts to an endofunction on the set of basis elements of the space of linear step functions $\mathcal{S}_D^L$.

Lemma 15. *Suppose $D \subseteq \mathbb{R}$ is a subfield, $u \in \mathcal{V}_D$ and $y \in \mathcal{S}_D^L$. Then, there is $f \in \mathcal{S}_D^C$ with $\mathcal{N}(f) \leq 3\mathcal{N}(y)\mathcal{N}(u)$ and $u \circ y(x) = f(x)$ for all but finitely many $x \in [-a,a]$. Moreover, f can be computed in time $\mathcal{O}(\mathcal{N}(u)\mathcal{N}(y))$.*

Now that we have a basis representation of $u \circ y$, it's easy to obtain a basis representation of $P_u(y)$ by integration. Note that computing integrals can be performed over a base defined over a subring of $\mathbb{R}$; we will make use of this fact later.

Lemma 16. *Suppose $D \subseteq \mathbb{R}$ is a subring and let $g(x) = \int_0^x f(x)dx$ for $f \in \mathcal{S}_D^C$. Then $g \in \mathcal{S}_D^L$ and $\mathcal{N}(g) = \mathcal{N}(f)$. Furthermore, g can be computed in $\mathcal{O}(\mathcal{N}(f))$ steps.*

Summing up, we have the following estimate on the algorithm induced by Theorem 11 if we compute over the base of piecewise linear functions.

Proposition 17. *Suppose $D \subseteq \mathbb{R}$ is a subfield, $u \in \mathcal{V}_D$ and $y \in \mathcal{S}_D^L$.*

(i) $P_u(y) \in \mathcal{S}_D^L$
(ii) $P_u(y)$ *can be computed in time* $\mathcal{O}(\mathcal{N}(u)\mathcal{N}(y))$.
(iii) $\mathcal{N}(P_u(y)) \in \mathcal{O}(\mathcal{N}(u)\mathcal{N}(y))$.

We can now summarise our results as follows:

Theorem 18. *Suppose $D \subseteq \mathbb{R}$ is a subfield and $u = \bigsqcup_{k\in\mathbb{N}} u_k$ with $u_k \in \mathcal{V}_D$. If $y_{k+1} = P_{u_k}(y_k)$, then*

(i) $y_k \in \mathcal{S}_D^L$ *for all* $k \in \mathbb{N}$
(ii) $y = \bigsqcup_{k\in\mathbb{N}} y_k$ *has width* 0 *and* $y^- = y^+$ *solves the IVP (1).*
(iii) $w(y_k) \in \mathcal{O}(c^k)$ *if* $w_u(u_k) \in \mathcal{O}(c^k)$.

Since the elements of $\mathcal{S}_D^L$ for $D = \mathbb{Q}$, the set of rational numbers, can be represented faithfully on a digital computer, the theorem – together with Proposition 5 – guarantees soundness and completeness also for implementations of the domain theoretic method. We also provide a guarantee on the speed of convergence, since the condition $w_u(u_k) \in \mathcal{O}(c^k)$ can always be ensured by the library used to construct the sequence (u_k) of approximations to the vector field.

Also, computing over the base of piecewise linear functions eliminates the need of computing rectangular enclosures at every step of the computation. This avoids the well-known wrapping effect of interval analysis, but it comes at the cost of a high complexity of the representation of the iterates. The next section presents an alternative, which uses piecewise constant functions only.

6 Computing with Piecewise Constant Functions

We have seen that the time needed to compute $P_u(y)$ is quadratic in terms of the complexity of the representation of u and y. However, the complexity of the representation of $P_u(y)$ is also quadratic in general. This implies that

$$\mathcal{N}(y_{k+1}) \in \mathcal{O}(\mathcal{N}(u_0)\dots\mathcal{N}(u_k)),$$

if $u = \bigsqcup_{k\in\mathbb{N}} u_k$ and $y_{k+1} = P_{u_k}(y_k)$.

The blow up of the complexity of the representation of the iterates is due to the fact that each interval on which y is linear is subdivided when computing $u \circ y$, since we have to intersect linear functions associated with y with constant functions induced by u, as illustrated by the left diagram in Figure 1.

This can be avoided if we work with piecewise constant functions only. The key idea is to transform the linear step function $P_u(y)$ into a simple step function before computing the next iterate: on every interval, replace the upper (linear) function by its maximum and the lower function by its minimum. We now develop the technical apparatus which is needed to show that the approximations so obtained still converge to the solution. Technically, this is achieved by making the partitions of the interval $[-a, a]$ explicit.

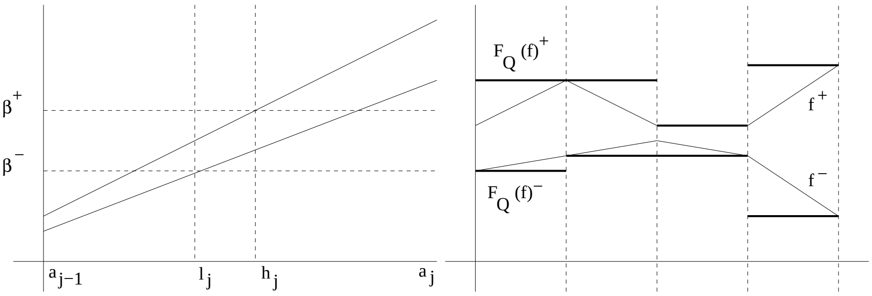

Fig. 1. Subdivision of Intervals (left) and Flattening (right)

Definition 5 (Partitions).

(i) A partition *of $[-a, a]$ is a finite sequence $(q_0, \dots, q_k)$ of real numbers such that $-a = q_0 < \dots < q_k = a$; the set of partitions of $[-a, a]$ is denoted by $\mathcal{P}$. If $D \subseteq \mathbb{R}$ then $\mathcal{P}_D \subset \mathcal{P}$ is the subset of partitions of $[-a, a]$ whose points lie in D.*

(ii) The norm *$|Q|$ of a partition $Q = (q_0, \dots, q_k)$ is given by $|Q| = \max_{1 \le i \le k} q_i - q_{i-1}$.*

(iii) A partition $Q = (q_0, \dots, q_k)$ refines *a partition $R = (r_0, \dots, r_l)$ if $\{r_0, \dots, r_l\} \subseteq \{q_0, \dots, q_k\}$; this is denoted by $R \le Q$.*

We are now ready for the definition of the flattening functional, which transforms piecewise linear functions to piecewise constant functions.

Definition 6. *Suppose $Q \in \mathcal{P}$. The* flattening functional *$F_Q : \mathcal{S} \to \mathcal{S}$ associated with Q is defined by*

$$F_Q(f) = (q_0, \dots, q_k) \searrow^{C} (\gamma_1, \dots, \gamma_k)$$

where $\gamma_i = \bigsqcap\{f(x) \mid x \in [q_{i-1}, q_i]\}$ for $1 \le i \le k$.

Note that, geometrically speaking, F_Q computes an enclosure of semi continuous functions into rectangles, as illustrated by the right diagram in Figure 1.

Lemma 19. *F_Q is well defined, that is, $F_Q(f)$ is continuous, if f is continuous.*

In order to reduce the complexity of the representations of the iterates, we want to apply the flattening functional at every step of the computation. The following lemma is the stepping stone in proving that this does not affect convergence to the solution.

Lemma 20. *Suppose $(Q_k)_{k \in \mathbb{N}}$ is an increasing sequence of partitions with $\lim_{k \to \infty} |Q_k| = 0$. Then $\bigsqcup_{k \in \mathbb{N}} F_{Q_k} = \mathrm{id}$.*

Proof. This follows from the fact that for every upper semi continuous function $f : [-a, a] \to \mathbb{R}$ and every decreasing chain $\alpha_0 \subseteq \alpha_1 \subseteq \dots$ of compact intervals containing x with $w(\alpha_k) \to 0$ as $k \to \infty$ one has $f(x) = \inf_{k \in \mathbb{N}} \sup\{f(x) \mid x \in \alpha_k\}$, and the dual statement for lower semi continuous functions.

The last lemma puts us in the position to show that the application of the flattening functional at every stage of the construction does not affect the convergence of the iterates to the solution.

Theorem 21. *Suppose $u = \bigsqcup_{k\in\mathbb{N}} u_k$, $(Q_k)_{k\in\mathbb{N}}$ is an increasing sequence of partitions with $\lim_{k\to\infty} |Q_k| = 0$ and $y_{k+1} = F_{Q_k}(P_{u_k}(y_k))$. Then $y = \bigsqcup_{k\in\mathbb{N}} y_k$ satisfies $y = P_u(y)$ and $w(y) = 0$.*

Proof. Follows from the interchange-of-suprema law (see e.g. [4, Proposition 2.1.12]), the previous lemma and Theorem 11.

We now show that the speed of convergence is essentially unaffected if we apply the flattening functional at every stage of the computation. This result hinges on the following estimate:

Lemma 22. *Suppose $u' \in \mathcal{V}$ with $u' \sqsubseteq u$ and $Q \in \mathcal{P}$, $y \in D$. Then $w(F_Q(P_{u'}(y))) \leq aL \cdot w(y) + a \cdot w_u(u') + 2\frac{K}{a}|Q|$.*

This lemma implies that flattening does not affect the speed of convergence.

Proposition 23. *Suppose $u = \bigsqcup_{k\in\mathbb{N}} u_k$ with $w_u(u_k) \leq c^k \cdot M(c - aL)$ and $(Q_k)_{k\in\mathbb{N}}$ is an increasing sequence of partitions with $|Q_k| \leq c^k \cdot \frac{a}{2}(c - aL)$. Then $w(y_k) \leq c^k w(y_0)$ if $y_{k+1} = F_{Q_k}(P_{u_k}(y_k))$ for all $k \geq 0$.*

We now show that the application of the flattening functional at every step avoids the blow up of the size of the iterates. As a consequence, the algorithm with flattening can be implemented using a base of functions defined over a dense subring of $\mathbb{R}$, such as the dyadic numbers.

Lemma 24. *Suppose $D \subseteq \mathbb{R}$ is a subring and $Q \in \mathcal{P}_D$. Then F_Q restricts to a mapping $\mathcal{S}_D^L \to \mathcal{S}_D^C$.*

The complexity of the algorithm underlying Theorem 21 over the bases $\mathcal{V}_D$ and $\mathcal{S}_D^C$ can now be summarised as follows, where $\mathcal{N}(Q) = k$ for a partition $Q = (q_0, \ldots, q_k)$.

Lemma 25. *Suppose $D \subseteq \mathbb{R}$ is a subring, $y \in \mathcal{S}_D^C$ and $u \in \mathcal{V}_D$.*

(i) $F_Q(P_u(y)) \in \mathcal{S}_D^C$ *and* $\mathcal{N}(F_Q(P_u(y)) = \mathcal{N}(Q)$
(ii) $F_Q(P_u(y))$ *can be computed in time* $\mathcal{O}(\max(\mathcal{N}(u) \cdot \mathcal{N}(y), \mathcal{N}(Q))$.

We can now summarise our results concerning soundness and completeness of the algorithm with flattening as follows:

Theorem 26. *Suppose $D \subseteq \mathbb{R}$ is a subring and $u = \bigsqcup_{k\in\mathbb{N}} u_k$ with $u_k \in \mathcal{V}_D$. Furthermore, assume $(Q_k)_{k\in\mathbb{N}}$ is an increasing sequence of partitions with $\lim_{k\to\infty} |Q_k| = 0$ and $y_{k+1} = F_{Q_k}(P_{u_k})(y_k)$.*

(i) $y_k \in \mathcal{S}_D^C$ *for all* $k \in \mathbb{N}$ *and* $\mathcal{N}(y_k) = \mathcal{N}(Q_k)$.
(ii) $y = \bigsqcup_{k\in\mathbb{N}} y_k$ *has width* 0 *and* $y^- = y^+$ *solves the IVP (1)*
(iii) $w(y_k) \in \mathcal{O}(c^n)$, *if both* $w_u(u_k)$ *and* $|Q_k| \in \mathcal{O}(c^k)$.

Note that, for a subring $R \subseteq \mathbb{Q}$ of the rational numbers, the elements of $\mathcal{V}_D$ and $\mathcal{S}_D^C$ can be faithfully represented on a digital computer. Hence we can guarantee both soundness and completeness also for an implementation of the domain theoretic approach where furthermore the size of the iterates are bounded above by the size of the partitions.

Acknowledgements

This work has been supported by EPSRC in the UK and the EU project "APPSEM-II".

References

1. AWA. A software package for validated solution of ordinary differential equations. `www.cs.utep.edu/interval-comp/intsoft.html`
2. *The GNU multi precision library*. `www.swox.com/gmp/`.
3. O. Aberth. Computable analysis and differential equations. In *Intuitionism and Proof Theory, Studies in Logic and the Foundations of Mathematics*, pages 47–52. North-Holland, 1970. Proc. of the Summer Conf. at Buffalo N.Y. 1968.
4. S. Abramsky and A. Jung. Domain theory. In S. Abramsky, D. M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3. Clarendon Press, 1994.
5. J. P. Aubin and A. Cellina. *Differential Inclusions*. Springer, 1984.
6. V. Brattka. Computability of Banach space principles. Informatik Berichte 286, FernUniversität Hagen, Fachbereich Informatik, June 2001.
7. F. H. Clarke, Yu. S. Ledyaev, R. J. Stern, and P. R. Wolenski. *Nonsmooth Analysis and Control Theory*. Springer, 1998.
8. J. P. Cleave. The primitive recursive analysis of ordinary differential equations and the complexity of their solutions. *Journal of Computer and Systems Sciences*, 3:447–455, 1969.
9. E. A. Coddington and N. Levinson. *Theory of Ordinary Differential Equations*. McGraw-Hill, 1955.
10. A. Edalat, M. Krznarić, and A. Lieutier. Domain-theoretic solution of differential equations (scalar fields). In *Proceedings of MFPS XIX*, volume 83 of *Elect. Notes in Theoret. Comput. Sci.*, 2004. Full Paper in `www.doc.ic.ac.uk/~ae/papers/scalar.ps`.
11. A. Edalat and A. Lieutier. Domain theory and differential calculus (Functions of one variable). In *Seventh Annual IEEE Symposium on Logic in Computer Science*. IEEE Computer Society Press, 2002. Full paper in `www.doc.ic.ac.uk/~ae/papers/diffcal.ps`.
12. A. Grzegorczyk. Computable functionals. *Fund. Math.*, 42:168–202, 1955.
13. A. Iserles. *Numerical Analysis of Differential Equations*. Cambridge Texts in Applied Mathematics. CUP, 1996.
14. Ker-I Ko. On the computational complexity of ordinary differential equations. *Inform. Contr.*, 58:157–194, 1983.
15. A. N. Kolmogorov and S. V. Fomin. *Introductory Real Analysis*. Dover, 1975.
16. R.E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, 1966.
17. N. Th. Müller and B. Moiske. Solving initial value problems in polynomial time. In *In Proceedings of the 22th JAIIO - Panel'93*, pages 283–293, Buenos Aires, 1993.
18. N. S. Nedialkov, K. R. Jackson, and G. F. Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 105:21–68, 1999.
19. M. B. Pour-El and J. I. Richards. *Computability in Analysis and Physics*. Springer-Verlag, 1988.