Final Report on EPSRC grant GR/S31396/01 Principal Investigator: Abbas Edalat

Domain-theoretic Solution of Differential equations

http://www.doc.ic.ac.uk/exact-computation/

Abbas Edalat and Dirk Pattinson
Department of Computing, Imperial College London

November 29, 2006

1 Introduction

The major problem with standard computational methods in solving initial value problems (IVP), defined by ordinary differential equations (ODE's), is that estimating the global error, which arises from initial, local as well as round-off errors due to floating point arithmetic, is extremely involved and the error bound determined for the approximate solutions can be generally very conservative. Using such bounds to meet a given precision can lead to impractical results which are essentially useless [17, page 7]. This problem is more aggravated in the case of stiff or even mildly stiff differential equations. Consequently, the lingering question often encountered which cannot be effectively answered is: How accurate is the approximate numerical solution obtained? (see [21, Section 3.5 and page 127].)

Validated numerical algorithms provide a numerical solution together with a guaranteed error bound. Validated real number computation in general, and therefore also validated numerical solutions of differential equations in particular, have traditionally be the focus of *interval analysis*. In the approach of interval analysis, real numbers are accounted for using intervals with machine representable endpoints, and outward rounding is applied whenever the result of an arithmetic operation is not machine representable. Interval analysis begins with the publication of Ramon Moore's book [24] that adapts a large number of numerical techniques to an interval setting. While methods of interval analysis are always guaranteed to produce an interval that contains the real value of the solution of a particular problem, there is no control over outward rounding, which is relegated to the floating point implementation of the particular system that runs the algorithm. For the same reason, interval techniques do not provide an adequate framework to reason about computability and complexity of problems in real analysis.

Domain theory [1], on the other hand, is faithful to the notion of computability and – applied to problems in numerical analysis – provides a very fine grained control over the round-off error. This is achieved by embedding the objects of interest (e.g. real numbers, continuous or differentiable multivariable functions) into an appropriate domain, a directed complete partial order where the ordering signifies increase in information content. By constructing the embeddings such that they take values in ω-continuous Scott domains, one can borrow from the rich theory of computability over domains, and is also in a position to derive proper data types that serve as the basis for the implementation of the algorithms in a high-level programming language.

Following the construction of domain-theoretic models for measure and integration theory, iterated function systems, exact real number computation, solid modelling and computational geometry, and their applications in reliable numerical and geometric computation since the early 1990's [3, 7, 5], we have embarked on obtaining domain-theoretic solution of IVP's in this project.

1.1 Background on ODE solving in Interval Analysis

The first treatment of initial value problems in the context of interval analysis dates back to Moore's book [24] in 1966. In this book, Moore translates Euler's Method for solving initial value problems into the context of interval analysis. This results in algorithms that always give guaranteed bounds of the solution, but no estimates for the speed of convergence are possible. Moore's algorithms have been refined in subsequent years, and have been implemented in the software packages AWA [23, 22] in the 1980s and, more recently, in VNODE [27] and ADIODES [30]. The main body of research on validated methods for solving differential equations in the context of interval analysis has been concerned with optimizations of the basic principle outlined by Moore [24]. Sophisticated methods have been devised to control the size of the time-steps on which the method relies and the so-called wrapping effect [15, 25, 26]. While this work certainly pushes further the boundary of problems amenable to the interval analysis, it is only the domain theoretic view that allows us to state and prove assertions about the convergence speed of a particular method. Moreover, domain theory gives rise to directly implementable data types that can be used in implementations of the ensuing algorithms in high-level programming languages.

From a practical point of view, all the implementations known to us [23, 30, 27] assume that the vector field that defines the differential equation is smooth and rely on automatic differentiation techniques to compute interval enclosures of the true solution. This contrasts with the domain theoretic approach, where we use a bare minimum of assumptions: here, vector fields are only assumed to be Lipschitz continuous, and we have outlined how to use the domain theoretic derivative [9] to obtain approximate information about derivatives to improve convergence speed.

2 Research Achievements

This project has been aimed at solving ordinary differential equations up to any guaranteed degree of accuracy. It overcomes a major difficulty of existing numerical methods, where tight and practical error estimates, that moreover incorporate the incurred round-off errors, are extremely hard to obtain. All papers and implementations of the project are available on-line at the project's webpage [16].

2.1 Domain of differentiable functions

We have developed a domain-theoretic model for differential calculus that allows to tackle IVP's. Our framework is centred around a domain for continuously differentiable multi-variable functions. Elements of this domain represent consistent partial information about a (differentiable) function and its first derivative. The consistency predicate is shown to be decidable on the basis elements of the domain, namely step functions. (As in classical mathematics, step functions in domain theory take only finitely many values, in this case intervals). Moreover, this newly introduced domain carries an effective structure and therefore provides an adequate framework for questions related to computability and complexity and gives rise to proper data types to represent differentiable functions in high-level programming languages, all of which go well beyond what interval analysis provides.

We have constructed a computable, domain-theoretic extension of Picard's operator [6] on this domain and shown that, for any computable vector field, it has a computable least fixed point, i.e., a computable interval-valued function which contains all classical solutions to the IVP, thus providing lower and upper bounds for the solutions. For a computable Lipschitz vector field, this least fixed point will be precisely the unique solution of the IVP and our computability result then is reduced to the classical result in computable analysis [29, 31].

Function information is represented using the domain of interval valued functions of a finite number of real variables. This simple domain is employed to solve an IVP in an exact framework. It provides the data type required both to represent the vector field with which the ordinary differential equation is defined and the solution to the IVP. It also enables us to construct a rigorous framework for the work in interval analysis on solving IVP's, in which we can study the soundness and completeness of the interval approach.

2.2 Soundness and completeness for solving IVP's

We have used the interval domain to develop a computable framework for solving differential equations such that any computable differential equation with a computable initial value has a computable solution. We have achieved this in two different ways: (i) by constructing a computable, domain-theoretic extension of Picard's theorem [10, 11], and (ii) by constructing a computable, domain-theoretic extension of Euler's method in solving ODE's [13]. The latter is closely related to the scheme used in interval analysis to solve IVP's [24] and formalizes this approach within domain theory.

In both frameworks, we have obtained algorithms for finding approximations to the solution of an initial value problem given by a differential equation. The vector field that specifies the IVP is defined in terms of elementary functions, which are themselves represented as least upper bounds of step functions. The approximation of the vector field in terms of step functions is obtained by employing existing packages of interval analysis to find the interval range of elementary functions. Our algorithms produce, at each stage of computation, two piecewise liner functions that provide lower and upper bounds to the solution.

The solution of the IVP can then be obtained as a piecewise linear function with the precision required by the user. Both frameworks are shown to be sound, i.e., the solution of the IVP is always contained within the approximating lower and upper piecewise linear functions, and complete, i.e., the sequences of lower and upper piecewise linear approximations respectively converge from below and above to the solution. Working with a domain-theoretic basis that is defined in terms of rational or dyadic numbers guarantees that completeness of the framework carries over to implementation.

2.3 Algebraic complexity and implementation

We have given lower bounds for the algebraic complexity of the algorithm. The complexity of representing the output is linear in terms of the complexity of the approximation of the vector field by step functions. The speed of convergence to the solution is exponentially fast if the approximating step functions converge exponentially fast to the vector field defining the IVP.

We have produced prototype implementations of our algorithms in C, based on interval analysis packages that approximate the range of elementary functions; these packages are only available in C. Two implementations have been undertaken, one using the domain-theoretic extension of Picard's theorem and one using the domain-theoretic extension of the Euler method. These are made available in the public domain [19].

The space complexity is linear for the domain-theoretic extension of the Euler method, but it explodes exponentially for the Picard method. Thus, the Euler method is the preferred framework for our implementation and it competes well with the corresponding method in interval analysis.

2.4 Other methods and classes of ODE's

Second order Euler method. The full domain of the differentiable functions with its consistency predicate is employed to develop a domain-theoretic extension of Euler's method of order two using the domain-theoretic derivative (equivalently the Clarke gradient [2]) of the vector field, which is a novel use of the Clarke gradient in solving initial value problems [1]. This allows us to obtain solutions of the IVP defined by vector fields which are non-smooth based on an interval version of the Taylor series expansion using the interval-valued derivative of the function, which as indicated in the Introduction gives a new method in interval analysis for solving IVP's.

Linear boundary value problem. We have obtained a domain theoretic framework for obtaining exact solutions of linear boundary value problems [28]. Based on the domain of compact real intervals, we show how to approximate both a fundamental system and a particular solution up to an arbitrary degree of accuracy. The boundary conditions are then satisfied by solving a system of imprecisely given linear equations at every step of the approximation. By restricting the construction to effective bases of

[1]The domain-theoretic derivative coincides with the Clarke gradient for locally Lipschitz functions on finite dimensional Euclidean spaces; see below.

the involved domains, we not only obtain results on the computability of boundary value problems, but also directly implementable algorithms, based on proper data types, that approximate solutions up to an arbitrary degree of accuracy. As these data types are based on rational numbers, no numerical errors are incurred in the computation process.

IVP's with imprecise initial value or vector field. This work will also enable us to solve differential equations whose vector fields are interval-valued and/or the initial value is uncertain, giving rise to new classes of solutions of differential equations. In the case of IVP's with uncertain initial values, our algorithms produce approximations, respectively from below and above, which converge to the least and greatest solutions possible with the uncertain input. In the case of imprecise vector fields, we obtain lower and upper bounds for any solution of the IVP.

2.5 Application to hybrid systems

We have used our data type for solving initial value problems to develop, for the first time, a denotational semantics for hybrid systems, i.e. systems that combine discrete and continuous behaviour, where the latter is governed by a family of initial value problems. The reachable states of a hybrid system that satisfies a non-zenoness condition, are obtained as the unique fixed point of an operator, called the forward action of the hybrid system, on the domain of (time dependent) states of hybrid system.

The denotational semantics is defined as the least fixpoint of the forward action operator on the continuous domain of functions of a non negative real variable, that take values in the lattice of compact subsets of n-dimensional Euclidean space. The semantic function assigns to every point in time t the set of states the automaton can visit at time t, starting from one of its initial states. Our denotational semantics for non-linear hybrid automata is related to the operational semantics given in terms of hybrid trajectories.

Our main results are the correctness and computational adequacy of the denotational semantics with respect to the operational semantics given in terms of hybrid trajectories. Moreover, we show that our denotational semantics can be effectively computed, which allows for the effective analysis of a large class of non linear hybrid automata. This enables us to effectively approximate the reachable state of such hybrid systems [14].

2.6 Implicit and inverse function theorems

As a byproduct of our work on the domain for differentiable functions we have obtained a domain-theoretic construction of the inverse and implicit functions theorems [12]. This has been achieved by formulating a domain-theoretic calculus for differentiable functions, which includes addition, subtraction and composition. Using the domain for differential functions, we have then developed a domain-theoretic version of the inverse function theorem and implicit function theorem, in which, subject to the usual conditions, the inverse function and the implicit function and their derivatives are obtained as fixed points of Scott continuous functionals and are approximated by step functions.

This means that from an increasing sequence of step functions converging to a function and its derivative in the domain of differentiable functions we can effectively obtain an increasing sequence of step functions converging in this domain to the inverse function and its derivative, and also effectively obtain an increasing sequence of polynomial step functions whose lower and upper bounds converge in the C^1 norm to the inverse function. A similar result holds for implicit functions. Combined with the domain-theoretic model for computational geometry [7, 5], this provides a robust technique for construction of curves and surfaces in geometric modelling and CAD.

2.7 Continuous derivative for functions

As a further area of research, the domain-theoretic derivative for locally Lipschitz real-valued functions of a finite dimensional Euclidean space has been shown to coincide with the Clarke gradient, which is used in non-smooth analysis and control theory. Since any such locally Lipschitz function together with its Clarke (or domain-theoretic) derivative is a maximal element of the Scott continuous domain of differentiable functions equipped with an effective structure, we have obtained a computable representation for the Clarke gradient. Moreover, the domain-theoretic derivative, or the L-derivative as we have called it, can be extended to real-valued functions on Banach spaces. It is not known if the L-derivative, defined using domain theory, and the Clarke gradient coincide on infinite dimensional Banach spaces. As with the Clarke gradient, the values of the L-derivative of a function are non-empty weak* compact and convex subsets of the dual of the Banach space. The L-derivative, however, is shown to be upper semi-continuous (equivalently Scott continuous), a result which is not known to hold for the Clarke gradient. Thus, the L-derivative provides, for the first time, a notion of derivative of real-valued functions which is continuous on any Banach space. We also formulate the notion of primitive maps dual to the L-derivative, an extension of Fundamental Theorem of Calculus for the L-derivative and a domain for computation of real-valued functions on a Banach space with a corresponding computability theory [4].

2.8 Future work

Second order Euler method. Based on the theoretical framework for a domain-theoretic version of the second Euler method using the Clarke gradient for the first non-trivial term of the Taylor series expansion of the vector field [9], we intend to develop the full details of the algorithm and implement it.
Boundary value problems via domain theory. We have already developed a domain-theoretic framework for linear boundary value problems [28], which we will extend to nonlinear problems.
Partial differential equation. PDE's provide an infinitely more challenging venture than ODE's. However, using a generalization of Picard's method we aim to construct a domain-theoretic framework for solving PDE's defined by analytic functions.
Euclidean manifolds. Based on the domain for multi-variable differentiable calculus, we seek to develop a domain for orientable, Euclidean and Lipschitz manifolds with applications in geometric modelling.

3 Project plan review and expenditure

In the early months of the project, as envisaged in the original proposal, we used a domain-theoretic extension of the Picard theorem on the domain of differentiable functions, which involved the notion of strong consistency of the function and derivative parts as well as updating both of these parts at each iterate of the associated Picard operator [8, 6]. Marko Krznarić's PhD thesis presents the development of this method [20].

However, in joint work with Dr Dirk Pattinson we soon developed a far simpler framework based on a domain-theoretic extension of the Picard operator on the function part of this domain, which allows us to obtain the solution of the IVP by updating only the function approximation without needing to update the derivative part or considering the strong consistency predicate. Unforeseen in the original proposal, this new framework has since been the focus of our work. It has allowed for a straightforward analysis of the algebraic and computational complexity of the resulting algorithms and has enabled us to have a more efficient implementation as well. Furthermore, the first order domain-theoretic Euler method developed in the new framework is closely related to the corresponding interval analysis method, thus allowing us to make a direct comparison between the two methods and discover the extra advantages of using domain theory in obtaining completeness of the framework and its complexity analysis.

Consequently, Dr Pattinson, who was a postdoc in Munich at the time spent a whole year at Imperial College on a DFG visiting fellowship (April 2004-March 2005). Later, he continued to visit us on a regular basis and worked on the project as a consultant and visiting researcher. Dr Khanban did the implementation of the project while he completed a part-time PhD on a related topic, namely a domain-theoretic algorithms for computational geometry [18].

We managed to obtain all manuals and software tools, and in particular the interval analysis packages, as open source which substantially cut down on our projected budget for consumables. On the other hand, instead of two PC's we purchased one PC and two laptops since Abbas Edalat was on sabbatical in 2005-2006 and had to work on the project while visiting overseas universities and Ali Khanban who was doing the implementation took a posdoctoral job after finishing his PhD in October 2005 and had to work from home.

References

[1] 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.

[2] F. H. Clarke. *Optimization and Nonsmooth Analysis*. Wiley, 1983.

[3] A. Edalat. Domains for computation in mathematics, physics and exact real arithmetic. *Bulletin of Symbolic Logic*, 3(4):401–452, 1997.

[4] A. Edalat. A continuous derivative for real-valued functions. Invited submission to New Computational Paradigms, 2006.

[5] A. Edalat, A. A. Khanban, and A. Lieutier. Computational geometry with imprecise input data. In *Lecture Notes in Computer Science*, volume 3526, pages 117–127, 2005.

[6] A. Edalat, M. Krznarić, and A. Lieutier. Domain-theoretic solution of differential equations (scalar fields). In *Proceedings of MFPS XIX*, volume 83 of *Electronuc Notes in Theoretical Computer Science*, 2003. Full paper in `www.doc.ic.ac.uk/~ae/papers/scalar.ps`.

[7] A. Edalat and A. Lieutier. Foundation of a computable solid modelling. *Theoretical Computer Science*, 284(2):319–345, 2002.

[8] A. Edalat and A. Lieutier. Domain theory and differential calculus (Functions of one variable). *Mathematical Structures in Computer Science*, 14(6):771–802, December 2004.

[9] A. Edalat, A. Lieutier, and D. Pattinson. A computational model for multi-variable differential calculus. In V. Sassone, editor, *Proc. FoSSaCS 2005*, volume 3441, pages 505–519, 2005.

[10] A. Edalat and D. Pattinson. A domain theoretic account of Picard's theorem. In *Proc. ICALP 2004*, volume 3142, pages 494–505, 2004. Full paper to appear in volume 9 of *London Mathematical Society Journal of Computation and Mathematics*. It is available in `www.doc.ic.ac.uk/~ae/papers/picard.icalp.ps`.

[11] A. Edalat and D. Pattinson. Domain theoretic solutions of initial value problems for unbounded vector fields. In M. Escardó, editor, *Proc. MFPS XXI*, volume 155, pages 565–581, 2005.

[12] A. Edalat and D. Pattinson. Inverse and implicit functions in domain theory. In P. Panangaden, editor, *Proc. 20th IEEE Symposium on Logic in Computer Science (LICS 2005)*, pages 417–426, 2005.

[13] A. Edalat and D. Pattinson. A domain theoretic account of Euler's method for solving initial value problems. In J. Dongarra, K. Madsen, and J. Wasniewski, editors, *Proc. PARA 2004*, volume 3732, pages 112–121, 2006. Full version available in `www.pst.informatik.uni-muenchen.de/~pattinson/Publications/`.

[14] A. Edalat and D. Pattinson. Denotational semantics of hybrid automata. In L. Aceto and A. Ingofsdottir, editors, *Proc. FoSSaCS 2006*, volume 3921, pages 231–245, 2006.

[15] P. Eijgenraam. The solution of initial value problems using interval arithmetic. Technical Report Mathematical Centre Tracts 144, Stichting Mathematisch Centrum, Amsterdam, 1981.

[16] IC exact arithmetic page, 2006. http://www.doc.ic.ac.uk/exact-computation/.

[17] A. Iserles. *Numerical Analysis of Differential Equations*. Cambridge Texts in Applied Mathematics. CUP, 1996.

[18] A. A. Khanban. *Basic Algorithms of Computational Geometry with Imprecise Input*. PhD thesis, Imperial College London, 2005.

[19] A. A. Khanban. IC-ODE-solvers, 2006. http://www.doc.ic.ac.uk/exact-computation/.

[20] M. Krznarić. *Solving Differential Equations in Exact Real Arithmetic*. PhD thesis, Imperial College London, 2005.

[21] J. D. Lambert. *Computational Methods in Ordinary Differential Equations*. John Wiley & Sons, 1973.

[22] R. Lohner. Enclosing the solution of ordinary initial and boundary value problems. In E. Kaucher, U. Kulisch, and C. Ullrich, editors, *Computer Arithmetic: Scientific Computation and Programming Languages*, Wiley-Teubner Series in Computer Science. 1987.

[23] R. Lohner. *Einschließung der Lösung gewöhnlicher Anfangs- und Randwertaufgaben und Anwendungen*. PhD thesis, Inst. f. Angew. Mathematik, Universität Karlsruhe, 1988.

[24] R. E. Moore. *Interval Analysis*. Prentice-Hall, Englewood Cliffs, 1966.

[25] N. Nedialkov, K. Jackson, and G. Corliss. Validated solutions of initial value problems for ordinary differential equations. *Applied Mathematics and Computation*, 155:21–68, 1999.

[26] N. Nedialkov, K. Jackson, and D. Pryce. An effective high-oder interval method for validating existence and uniqueness of the solution of an IVP for an ODE. *Reliable Computing*, 7:449–465, 2001.

[27] N. Nedialkov and K.Jackson. The design and implementation of an object-oriented validated ODE solver. Draft, available via `http://www.cas.mcmaster.ca/~nedialk/`

[28] D. Pattinson. Domain theoretic formulation of linear boundary value problems. In B. Loewe, editor, *Proc. CiE 2005*, number 3526, pages 385–395, 2005.

[29] M. B. Pour-El and J. I. Richards. *Computability in Analysis and Physics*. Springer-Verlag, 1988.

[30] Ole Stauning. *Automatic Validation of Numerical Solutions*. PhD thesis, Technical University Dennmark, Lyngby, Denmark, 1997.

[31] K. Weihrauch. *Computable Analysis (An Introduction)*. Springer, 2000.