A Verified Enclosure for the Lorenz Attractor (Rough Diamond)

Fabian Immler*

Institut für Informatik, Technische Universität München
immler@in.tum.de

Abstract. A rigorous numerical algorithm, formally verified with Isabelle/HOL, is used to compute an accurate enclosure for the Lorenz attractor.

Accurately enclosing the attractor is highly relevant: a similar non verified computation is part of Tucker's proof that the Lorenz attractor is chaotic in a rigorous mathematical sense. This proof settled a conjecture that Fields medalist Stephen Smale has put on his list of eighteen important mathematical problems for the twenty-first century.

Keywords: Isabelle/HOL, Ordinary Differential Equation, Lorenz Attractor, Rigorous Numerics

1 Introduction

The Lorenz system of ordinary differential equation (ODEs) has become famous as a classical example of chaotic dynamics since its introduction as a model for atmospheric flows by Edward Lorenz in 1963. Numerical experiments suggested chaotic behavior; in dynamical systems parlance, the existence of a *strange attractor*. However, the existence of a strange attractor for the Lorenz equations could not be proved until 1999 – shortly after Fields medalist Stephen Smale put it on his list of eighteen unsolved mathematical problems for the 21st century.

The proof was accomplished by Warwick Tucker [8], and an interesting aspect is that his proof relies on the output of a numerical program. His program computes enclosures for the attractor and analytical properties of solutions on the attractor.

These programs were written in C++ and not formally verified, Tucker even discovered (and fixed) some bugs in it [7]. Formally verifying the numerical results needed for the proof is therefore a worthwhile goal.

The contribution of this work is the computation of an accurate enclosure for the Lorenz attractor with a formally verified ODE solver. The development is available in the Archive of Formal Proofs [5].

* Supported by the DFG RTG 1480 (PUMA)

2 The Lorenz Attractor

We consider (like Tucker) the classical parameter values, for which the Lorenz equations in Jordan normal form are approximately given by the right hand side $f(x,y,z) = (11.8x - 0.29(x+y)z, -22.8y + 0.29(x+y)z, -2.67z + (x+y)(2.2x - 1.3y))$. It can be shown that the properties of interest are robust under small perturbations of the parameters.

A solution $\varphi : \mathbb{R} \to \mathbb{R}^3$ is any function with derivative $\dot{\varphi}(t) = f(\varphi(t))$. With an *initial condition* $\varphi(0) = x_0$, the solution is unique. We denote with *flow* the solution $\varphi(x_0, t)$ depending on initial condition x_0 at time t.

Numerical simulations suggest the existence of a set $\mathcal{A}$, the *Lorenz attractor* (enclosures of which is depicted in Figure 1) with the following 3 properties, which make it a *strange* attractor.

Property 1. Solutions (of the Lorenz system) tend towards $\mathcal{A}$.

The dynamics on $\mathcal{A}$ can be described as follows: solutions starting from $\Sigma = \{(x,y,z) \mid z = 27 \wedge x \in [-5.5; 5.5]\}$ flow downwards (towards lower values of z) and enter either the left or right branch of the attractor in order to circle around the "holes" around $(\pm 6, \cdot, 27)$ and return back to Σ. Depending on where they return, they either stay in the same branch or switch to the other side in the next revolution. Small initial sets approaching the fixed point $(0,0,0)$ exhibit strong expansion in the x-direction, which causes Property 2.

Property 2. Solutions on $\mathcal{A}$ exhibit sensitive dependence on initial conditions.

The expansion is strong enough for Property 3.

Property 3. Small initial sets eventually spread over the whole attractor $\mathcal{A}$.

A standard approach in the analysis of dynamical systems is to provide sufficient conditions for properties 1,2,3 by studying a so-called *Poincaré map* R, which simplifies reasoning about the three-dimensional flow to reasoning about discrete iterations of the two-dimensional map R. To define R for Tucker's proof, consider the *return plane* Σ as defined before. For any point $x \in \Sigma$, $\tau(x)$ is the first time when x flows through Σ from above. The Poincaré map R is then defined as $R(x) := \varphi(x, \tau(x))$. The significance is that the equivalents of properties 1,2,3 for the iterations of the map R and its attractor $\mathcal{A} \cap \Sigma$ carry over to the flow φ and $\mathcal{A}$. Tucker proved those with a combination of rigorous numerics and analytical reasoning locally around the origin $(0,0,0)$.

Rigorous Numerics. Property 1 can be shown by exhibiting a forward invariant (i.e. $R(N) \subseteq N$) subset N of the return plane that contains the attractor: $\mathcal{A} \cap \Sigma \subseteq N \subseteq \Sigma$.

Tucker proves this by computing enclosures for R with *rigorous numerics*: a function $\mathit{step}(X)$ computes some set (by safely including e.g. round-off errors into the result) that is reachable via the flow.

$$\forall x \in X.\ \exists h > 0.\ \varphi(x,h) \in \mathit{step}(X)$$

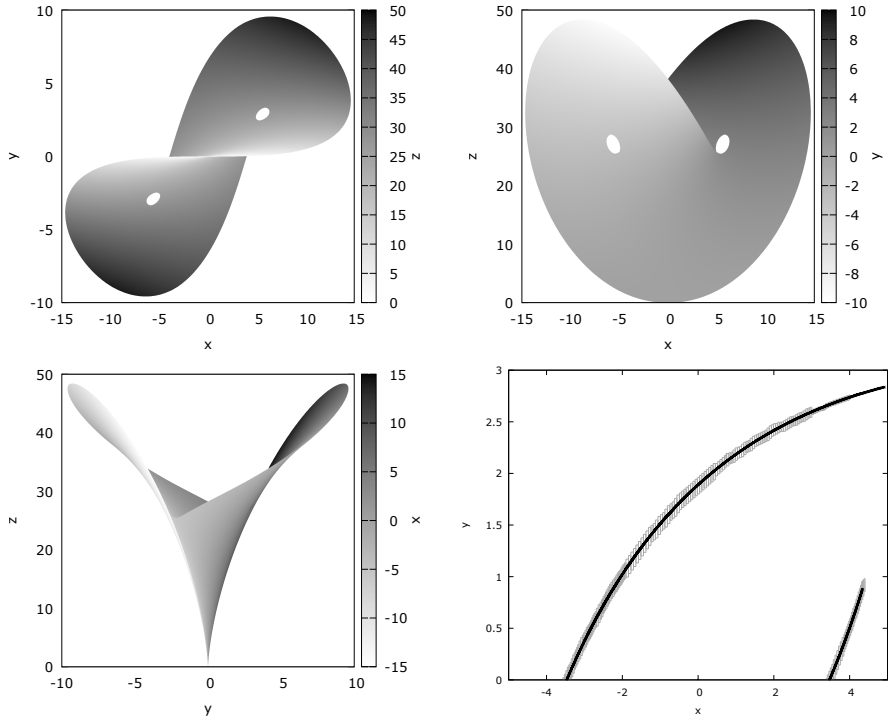

Fig. 1. *Top left, top right, bottom left*: projections of enclosures for Lorenz attractor. *Bottom right*: forward invariant subset N in black of the return plane Σ with Tucker's enclosure in gray; one half is omitted due to the symmetry $(x, y, z) \in N \longleftrightarrow (-x, -y, z) \in N$

Overapproximations to the Poincaré map R can then be obtained by iterating $\mathit{step}(X)$ until the return plane Σ is reached:

$\mathit{poincare}(X) :=$
  let $Y = \mathit{step}(X)$ **in if** $Y \cap \Sigma \neq \emptyset$ **then** $\mathit{step\text{-}to\text{-}sigma}(X)$ **else** $\mathit{poincare}(Y)$

Here *step-to-sigma* is like *step* but needs to satisfy $\mathit{step\text{-}to\text{-}sigma}(X) \subseteq \Sigma$. Now if $X \subseteq \Sigma$, then $R(X) \subseteq \mathit{poincare}(X)$. Tucker represents a candidate for the forward invariant subset as a union of small rectangles $N = \bigcup_{i \leq k} N_i$, for which computations confirm that $\mathit{poincare}(N_i) \subseteq N$ for all i and therefore $R(N) \subseteq N$.

Tucker implements *step* as a function that propagates axis-parallel rectangles by overapproximating the flow with the Euler method. Tucker implements *step-to-sigma* by choosing an appropriate interval for the step size and projecting the result onto the plane.

To quantify the dependence on initial conditions for Property 2, it is a standard approach to study the derivative DR of R. This can be done by adding to *step*, which computes overapproximations to the flow φ, overapproximations

for the partial derivatives $\frac{\partial\varphi}{\partial x}$, $\frac{\partial\varphi}{\partial y}$, $\frac{\partial\varphi}{\partial z}$, $\frac{\partial\varphi}{\partial t}$ of the flow. The overapproximations on DR can be used to prove that the N_i are expanded in some directions and contracted in others (via a "forward invariant cone field" for DR). Tucker can quantify the magnitude of the expansion as sufficiently large to establish Property 3.

Local Theory around the Origin. There is, however, one obstruction for the numerical methods: some solutions tend towards the origin $(0, 0, 0)$, and they do so in infinite time. Any time discretization algorithm would therefore need to take smaller and smaller steps but never reach the origin. Therefore, Tucker derived a coordinate change (in about 25 pages in his article) that makes the flow approximately linear in a cube with width 0.1 around the origin. Computations can be interrupted upon reaching that cube. The solution inside the cube can be propagated via an explicit formula, and the numerical computations can be continued afterwards.

3 ODEs and Numerical Solutions in Isabelle/HOL

In Isabelle/HOL [6], ordinary differential equations are formalized together with basic theorems for local existence/uniqueness (the Picard-Lindelöf theorem), global unique solutions and basic properties of the flow, like continuity with respect to initial conditions. Differentiability with respect to initial conditions is still missing and would be needed for reasoning about DR.

We use the formalization in Isabelle/HOL of rigorous numerical algorithms for ODEs [4]. The method we employ for *step* is slightly different from Tucker's approach: instead of the Euler method, we use the method of Heun, a two-stage Runge-Kutta method (where the error in one step is cubic in the step size) with adaptive step size control. Instead of rectangles we use zonotopes (our algorithm is based on affine arithmetic [1] instead of interval arithmetic). Numerical computations are carried out with software floating point numbers $m \cdot 2^e$ for (unbounded) integers $m, e \in \mathbb{Z}$. Explicit round-off operations restrict the size of m during the computations.

As detailed in the earlier paper [4], it turns out that reducing the reachable sets to two dimensions from time to time is crucial for maintaining precise enclosures and acceptable performance. For these reductions as well as for *step-to-sigma*, it is necessary to compute intersections of reachable zonotopes with intermediate planes or the return plane Σ, for which we use our formalization [3] of Girard/Le Guernic's geometric algorithm [2]. Tucker's algorithm achieves that implicitly because it propagates rectangles exclusively from plane to plane.

Tucker's and our implementation have in common that reachable sets are split when their size exceeds some given threshold.

The following theorem for partial correctness, proved in Isabelle/HOL assures that if *poincare* returns a result, this result is a safe overapproximation for the flow of the Lorenz equations:

Theorem 4. $\mathit{poincare}(X) \subseteq \Sigma \wedge \forall x \in X\ \exists t > 0.\ \varphi(x,t) \in \mathit{poincare}(X)$

4 Computing a Verified Enclosure for the Lorenz Attractor

We only tackle the numerical computations needed for Property 1: we verify a forward invariant set N for R (and in the process an enclosure for the Lorenz attractor $\mathcal{A}$, forward invariant under φ).

The set N used for our computations is plotted in the bottom right of Figure 1 in black. The enclosure that was verified by Tucker is depicted with gray rectangles and one can see that our computations are at least as accurate. In our case, N is a collection of 14816 squares N_i with width 2^{-8}.

4.1 Parallelization on Supercomputer

The overall computation is embarassingly parallel, since the computations for the 14816 initial rectangles N_i are independent. We extracted code for our verified algorithm poincare to Standard ML (SML) and compiled it with MLTon. With this setup, integers $\mathbb{Z}$ from the formalization are mapped to the arbitrary-precision integers from *The GNU Multiple Precision Arithmetic Library* (GMP).

Then we distributed the program for the different input data on 1024 cores of the computer cluster SuperMUC. With a wall-clock time limit of 7 hours, this amounts to a total computation time of around 7000 hours. Tucker's original computations (more than fifteen years ago) have been distributed on 20 computers for about 100 hours.

During reachability analysis, the program outputs a trace containing information like enclosures during propagation, which allowed us to plot enclosures of the Lorenz attractor (see Figure 1) generated from the formally verified program. Since every rectangle N_i returns within N, N is verified as forward invariant under R.

Note that a small part of the attractor is missing: we interrupt computations close to the origin (as is necessary in Tucker's proof as well), but we do not continue with a symbolic propagation from there. This only affects 16 reachable sets (which do not expand much anymore after leaving the cube around the origin) so the impact on overall computation time is negligible.

4.2 Parallelization with Isabelle/ML

If one wants to avoid running independent instances compiled outside of Isabelle, it is also possible to compile and evaluate the code from within Isabelle. Then Isabelle trusts the outcome of the computations after reconstructing Isabelle/HOL terms from the result of the SML program. To exploit the parallelization, Isabelle/HOL's library provides special combinators, for which the generated code uses parallel combinators of Isabelle/ML (i.e. `Par_List.map`). Using these combinators, we tested running the initial rectangles N_i for $0 \leq i \leq 32$ on a 8 core

machine, which gave a speedup of factor 6.1 compared to serial execution. Evaluating in Isabelle gives us the following theorem (trusting the code generation oracle).

Theorem 5. $\forall x \in \bigcup_{i \leq 32} N_i.\ \exists t > 0.\ \varphi(x,t) \in N$

5 Conclusion

We took a first step towards a formal verification of the numerical part of Tucker's proof. However we do not track the derivative DR, because we have no formalization of differentiability of the flow (and therefore R). It should increase the computational efforts only by a constant factor (since in every step, one propagates in addition to the flow on a reachable set just the derivative of the flow on a cone field). Furthermore, we ignore the symbolic propagation at the origin but this does not impact the overall computational effort too much.

Nevertheless we managed to obtain formally verified results on an important and computationally intensive part of the proof, which we hope to be able to extend with reasonable effort towards a propagation of DR.

Acknowledgments. I would like to thank Florian Haftmann for providing the theories for parallelization with Isabelle/HOL and Makarius Wenzel for the underlying infrastructure for parallel combinators in Isabelle/ML.

References

1. de Figueiredo, L., Stolfi, J.: Affine arithmetic: Concepts and applications. Numerical Algorithms 37(1-4), 147–158 (2004)
2. Girard, A., Le Guernic, C.: Zonotope/hyperplane intersection for hybrid systems reachability analysis. In: Egerstedt, M., Mishra, B. (eds.) Hybrid Systems: Computation and Control, LNCS, vol. 4981, pp. 215–228. Springer (2008)
3. Immler, F.: A verified algorithm for geometric zonotope/hyperplane intersection. In: Proceedings of the 2015 Conference on Certified Programs and Proofs. pp. 129–136. CPP '15, ACM, New York, NY, USA (2015)
4. Immler, F.: Verified reachability analysis of continuous systems. In: Baier, C., Tinelli, C. (eds.) Tools and Algorithms for the Construction and Analysis of Systems, Lecture Notes in Computer Science, vol. 9035, pp. 37–51. Springer Berlin Heidelberg (2015), http://dx.doi.org/10.1007/978-3-662-46681-0_3
5. Immler, F., Hölzl, J.: Ordinary differential equations. Archive of Formal Proofs (Aug 2015), Formal proof development, http://afp.sf.net/devel-entries/Ordinary_Differential_Equations.shtml
6. Nipkow, T., Paulson, L.C., Wenzel, M.: Isabelle/HOL: A proof assistant for higher-order logic. LNCS, Springer (2002)
7. Tucker, W.: My thesis: The lorenz attractor exists, http://www2.math.uu.se/~warwick/main/pre_thesis.html
8. Tucker, W.: A rigorous ODE solver and Smale's 14th problem. Foundations of Computational Mathematics 2(1), 53–117 (2002)