Towards a computer-assisted proof for chaos in a forced damped pendulum equation ⋆

Tibor Csendes and Balázs Bánhelyi

University of Szeged, Institute of Informatics, Szeged, Hungary

László Hatvani

University of Szeged, Bolyai Institute, Szeged, Hungary

Abstract

We report on the first steps made towards the computational proof of the chaotic behavior of the forced damped pendulum. Although, chaos for this pendulum was being conjectured for long, and it has been plausible on the basis of numerical simulations, there is no rigorous proof for it. In the present paper we provide computational details on a fitting model and on a verified method of solution. We also give guaranteed reliability solutions showing some trajectory properties necessary for complicate chaotic behaviour.

Key words: Differential equations, Chaos, Pendulum, Interval method
AMS Subject Classification: 65G30, 65G40

1 Introduction

One important question while studying computational approximations of solutions of differential equations is whether the given equation has chaotic solutions. That would imply that the numerical simulation must be carried out carefully, considering fitting measures against possible distraction due to accumulated rounding errors. Unfortunately, the recognition of chaotic behaviour has remained a hard to recognize feature that is usually studied by theoretical means [4]. There are a few exceptions such as Neumaier et al. [7,9] and the Polish team of Zglicynski [3,10].

⋆ This work has been partially supported by the Grant OTKA T 034350 and the Analysis Research Group of the Hungarian Academy of Science.

Email address: csendes@inf.u-szeged.hu (Tibor Csendes and Balázs Bánhelyi).

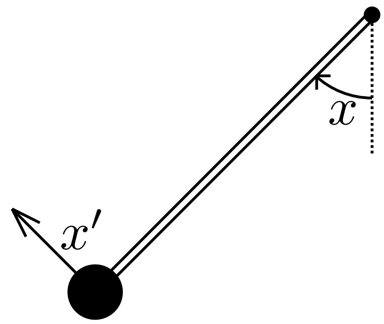

Fig. 1. An illustration of the studied forced damped pendulum, the angle x and the corresponding speed x' are indicated.

In our recent studies [1,2] we investigated the chaotic regions of some Hénon systems. We were able to present new chaotic regions for some Hénon systems, and also we could provide positive measure sets of mapping parameters that allow chaos within the earlier reported chaotic regions obtained for single parameter values only.

In the present paper we attack the problem of forced damped pendulum (see [4] and the references therein). This pendulum has been conjectured to be chaotic without a formal proof. Now we present a fitting verified numerical technique capable to prove extraordinarily complicated and unstable behaviour. For example, we can make our pendulum go through any specified sequence of gyrations by correctly choosing the initial conditions (see the details later).

Consider the forced damped pendulum, which is a mechanical system of one degree of freedom consisting of a mass point of mass m hung with a weightless solid rod of length l. This means that the point is forced to move along a vertical circle of radius l (see Figure 1) under the action of a gravitational field of force g, some friction proportional to the velocity, and the periodic external force $A \cos t$ ($A =$ const.). It is known [4] that motions of this system are described by the second order differential equation

$$mlx'' = -mg \sin x - \gamma l x' + A \cos t,$$

where t denotes the time, x is the angle of the pendulum, x' is the angle velocity, and $0 < \gamma =$ const. denotes the damping coefficient. Suppose that the parameters are chosen so that the equation of motion is

$$x'' = -\sin x - 0.1x' + \cos t.$$

The most effective way of studying the behaviour of this dynamical system is to take a "snapshot" of the $x - x'$ plane at each period 2π of the driving force. Such a "snapshot" is called a Poincaré section (see the illustration on Figure 2 and [8]). We take the Poincaré sections at the moments $t = 2n\pi$, where n is an integer. Mathematically, we iterate the so called Poincaré map $P : R^2 \to R^2$

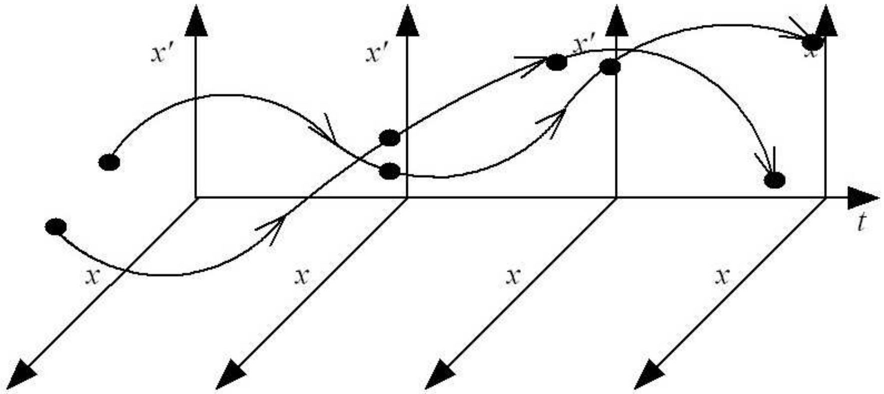

Fig. 2. The Poincaré sections for the forced damped pendulum problem. These are periodic intersections of the extended phase space t - x - x' projected to the x - x' plane.

defined by $(x(0), x'(0)) \mapsto (x(2\pi), x'(2\pi))$. Fixed points of the Poincaré map correspond to 2π-periodic solutions of our system of differential equations. An attracting fixed point of the Poincaré map is called a sink. It corresponds to a 2π-periodic solution such that in its neighbourhood all the solutions of the system tend to this periodic solution as $t \to \infty$. If a 2π-periodic solution $\tilde{x}$ has the property that some solutions tend to $\tilde{x}$ as $t \to \infty$, and some other solutions tend to $\tilde{x}$ as $t \to -\infty$, then $\tilde{x}$ (and the corresponding fixed point of the Poincaré map) is called a saddle. Saddle points play an important role in the detection of chaos; e.g., it was also the case for the Hénon mapping studied earlier by the authors [1,2] but also by others (e.g. [4,10]).

Let -1, 0, and 1, respectively, denote the event that, during one time interval $I_k = [2k\pi, 2(k+1)\pi]$, the pendulum crosses the downward position exactly once clockwise, it does not crosses the downward position, and it crosses the downward position once counterclockwise, respectively. One form of chaos is when we may arbitrarily prescribe the consecutive events of these types for a motion. More precisely, for every bi-infinite sequence $\ldots, \epsilon_{-2}, \epsilon_{-1}, \epsilon_0, \epsilon_1, \epsilon_2, \ldots$, where $\epsilon_k \in \{-1, 0, 1\}$, there exists a point (x_0, x'_0) such that, to the solution determined by these initial conditions, event ϵ_k happens during the time interval $I_k = [2k\pi, 2(k+1)\pi]$.

Following the notion applied in [4] we define three rectangles Q_{-1}, Q_0, and Q_1 in the $x - x'$ plane, each containing the k-th saddle point ($k = -1, 0, 1$). We illustrate the Q_k rectangles on Figure 3. These are compressed in the direction of the stable manifold and stretched in the direction of the unstable manifold.

To make the first step to prove the property mentioned above, we aim to prove now that for an arbitrary sequence of three indices i_1, $i_2 = 0$, and i_3 ($i_1, i_3 \in \{-1, 0, 1\}$) there exists a point $(x_0, x'_0) \in Q_0$ such that the trajectory determined by (x_0, x'_0) is exactly in Q_{i_1}, Q_0, and Q_{i_3} at $t = -2\pi$, 0, and 2π,

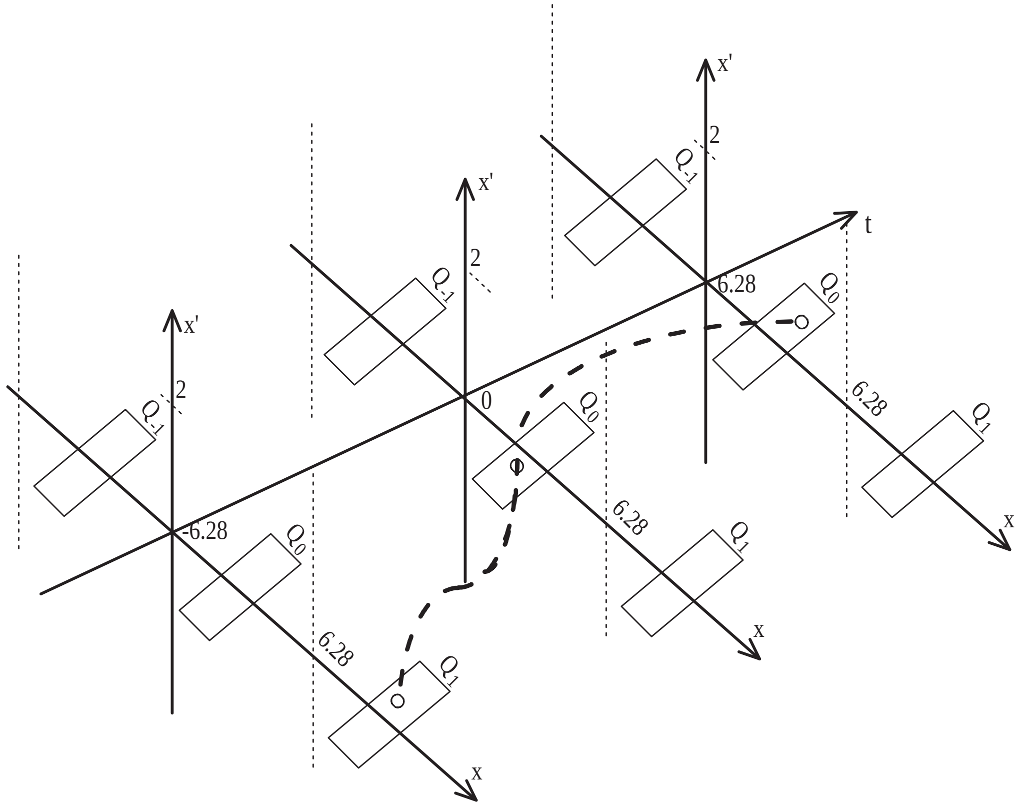

Fig. 3. The Poincaré sections with the Q_i rectangles studied in the present approach, together with the indicated trajectory related to the $Q_1 - Q_0 - Q_0$ sequence.

respectively. This is a kind of chaotic behaviour, and the later complete proof can be based on such a verified computer procedure for the realization of an arbitrary bi-infinite sequence.

2 An interval method to locate regions with chaotic features

In this section we introduce an inclusion function based verified procedure to prove the existence of points the related trajectories of which follow given patterns on the Poincaré sections. Our method applies a verified algorithm to solve initial value problems for systems of ordinary differential equations. This technique is responsible to guarantee that solutions starting from an interval stop inside a given aimed set at $t = 2\pi$. Once we are able to solve this problem (allowing overestimations and as a consequence, having uncertain answer for some larger intervals) we can compose a framework algorithm to search systematically for subintervals of the search domain that fulfill all the prescribed set theoretical containment conditions. This way of posing the problem is in full accordance with the problem setting used for the Hénon mapping chaos issue [1,2].

The applied algorithm encloses first the set Q_0 in a 2 dimensional interval I. This will be the starting interval. Then an adaptive subdivision technique gen-

Algorithm 1 : The Checking Routine

Inputs:
- ε: the user set limit size of subintervals,
- Q: the argument set to be proved,
- Q_-, Q_+: the aimed sets.

Calculate the initial interval I, that contains the regions of interest
Push the initial interval into the stack
while (the stack is nonempty)
  Pop an interval v out of the stack
  Calculate the width of v
  Determine the widest coordinate direction
  Calculate the transformed intervals $w_1 = T_-(v)$ and $w_2 = T_+(v)$
  if $v \subset Q$, $w_1 \subset Q_-$ and $w_2 \subset Q_+$, **then**
    print that the condition is fulfilled by v and **stop**
    else If the width of interval v is larger than ε, and
      $v \cap Q \neq \emptyset$ and $w_1 \cap Q_- \neq \emptyset$ and $w_2 \cap Q_+ \neq \emptyset$ **then**
      bisect v along the widest side: $v = v_1 \cup v_2$ and
        push the subintervals into the stack
    endif
  endif
end while
print that the search was unsuccessful and **stop**

erates such a subdivision of the starting interval that either for all subintervals (of a user set small size) one of the given conditions does not hold (at least for one point of the respective set), or it is shown that a subinterval exists, that complies with the given conditions. In Algorithm 1 the transformation T_- and T_+ shift the argument interval to intervals that enclose the solutions of the differential equation of the pendulum for a -2π and 2π long time intervals, respectively. Since the present algorithm locates a subinterval that fulfills some conditions (the complement of the Hénon chaos case), we can refer to the correctness and finiteness results proven for this algorithm in [2].

3 Numerical results and conclusion

We calculated the inclusion of a solution of the differential equation with the VNODE algorithm [6] and based on the Profil/BIAS interval environment [5]. To be able to represent the integration limits we transformed the differential equation. The applied form of the differential equation was:

```
INTERVAL a(1.0);
INTERVAL b = a/((REAL)10.0);
INTERVAL Pi = ArcSin(a)*((REAL)2.0);
yprime[0] = Pi;
yprime[1] = (y[2])*Pi;
yprime[2] = (-a*sin(y[1])-b*y[2]+cos(y[0]))*Pi;
```

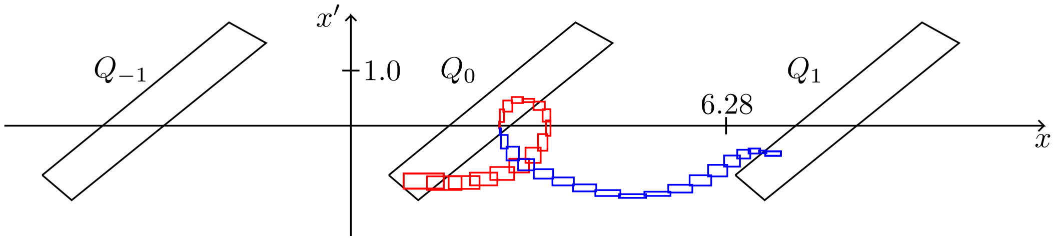

Fig. 4. The Q_i rectangles in the x - x' plane studied in the present approach, together with the enclosures of the trajectory related to the $Q_1 - Q_0 - Q_0$ sequence.

In this way we had to calculate the differential equation in time between -2 and 2 (both are computer representable).

A representative result for the Q_1–Q_0–Q_0 sequence is depicted on Figure 4. The Q_i rectangles in the x - x' plane studied in the present approach are displayed with the enclosures of the trajectory related to the Q_1–Q_0–Q_0 sequence. The trajectory starts from Q_0, its preimage is in Q_1, and its image in Q_0.

The full set of results obtained is comprised in Table 1. The lines contain the proven subintervals of Q_i, Q_0, and Q_j. Q_0 contains only such points which belong to a trajectory that passes the rectangles according to the given sequence (indicated in the first column). The Q_i and Q_j intervals include the preimage and image points that belong to Q_0. The interval bounds are rounded (to save space). The original bounds are validated. The total computation time required is given next in seconds. All the values in this column are acceptable, between 6 and 16 minutes on an average PC. The last column contains the number of subintervals generated and checked. These figures show again a moderate memory requirement. All the data together are quite encouraging for the complete proof of chaos.

Finally Figure 5 provides the result of the adaptive subdivision for the Q_1–Q_0–Q_0 sequence: the search interval is displayed containing the Q_0 rectangle together with the generated and checked subintervals. The dense set shows where the solution subinterval has been found (cf. Table 1).

Summarizing our experiences we can conclude that the described adaptive subdivision method could well utilize the verified differential equation solver VNODE to form an efficient tool that will be the core algorithm for the complete proof of chaotic behaviour in the case of the forced damped pendulum. The obtained result can be formalized as follows:

Assertion 1 *For the investigated forced damped pendulum all possible triplets of the form Q_-, Q_0, Q_+, where $Q_-, Q_+ \in \{Q_{-1}, Q_0, Q_1\}$, can be realized by proper selection of the starting point p within Q_0, such that $P^{-1}(p) \in Q_-$, $P(p) \in Q_+$.*

aimed sets	preimage / original / image intervals		CPUt (s)	No. of boxes
Q_0	[2.317648, 2.709306]	[−0.050055, 0.342587]	383	1390
Q_0	[2.628677, 2.629443]	[0.021484, 0.021973]		
Q_0	[1.076006, 1.253059]	[−1.048104, −0.964307]		
Q_1	[7.632088, 7.660986]	[−0.512440, −0.483095]	363	1615
Q_0	[2.696904, 2.698438]	[−0.047852, −0.046875]		
Q_0	[0.990481, 1.369249]	[−1.100671, −0.899117]		
Q_{-1}	[−2.894219, −2.710906]	[1.155728, 1.233331]	402	1595
Q_0	[2.607979, 2.609512]	[0.041992, 0.042969]		
Q_0	[1.028439, 1.405768]	[−1.086193, −0.880000]		
Q_0	[2.284252, 2.669446]	[−0.003131, 0.381040]	694	3178
Q_0	[3.751748, 3.752515]	[1.072266, 1.072754]		
Q_1	[7.173756, 7.188131]	[−1.052968, −1.048956]		
Q_1	[7.556686, 7.831820]	[−0.896126, −0.595137]	356	1364
Q_0	[3.875938, 3.888203]	[1.062500, 1.070313]		
Q_1	[7.265608, 7.524591]	[−1.042681, −0.922727]		
Q_{-1}	[−2.811017, −2.345166]	[1.059322, 1.274043]	561	2292
Q_0	[3.719551, 3.722617]	[1.078125, 1.080078]		
Q_1	[7.186581, 7.247708]	[−1.049464, −1.030382]		
Q_0	[2.342858, 2.729911]	[−0.079739, 0.311653]	1001	5258
Q_0	[1.421279, 1.422046]	[−0.881836, −0.880859]		
Q_{-1}	[−1.755311, −1.745855]	[1.595219, 1.602129]		
Q_1	[7.704982, 8.076489]	[−0.812236, −0.355093]	384	1462
Q_0	[1.520938, 1.533203]	[−0.968750, −0.953125]		
Q_{-1}	[−2.091369, −1.911043]	[1.343996, 1.492038]		
Q_{-1}	[−2.798045, −2.243367]	[0.960221, 1.221012]	652	2853
Q_0	[1.404414, 1.410547]	[−0.847656, −0.843750]		
Q_{-1}	[−1.877814, −1.815990]	[1.502639, 1.548777]		

Table 1
The obtained intervals the points of which coincide with trajectories that pass the respective, prescribed sets (Q_i, Q_0, Q_j) in the given order. The necessary CPU time in seconds and the checked number of subintervals are also provided.

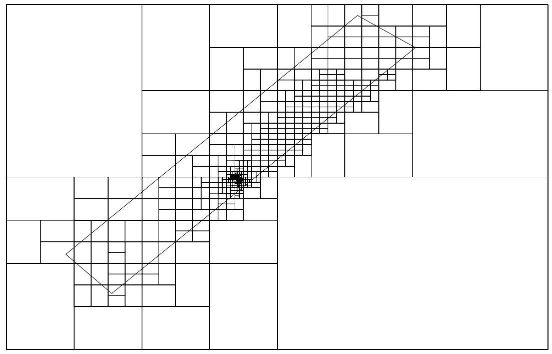

Fig. 5. The search interval containing the Q_0 rectangle with the subintervals generated and checked by our algorithm. The dense set indicates the solution subinterval that has been included in Table 1.

References

[1] B. Bánhelyi, T. Csendes, and B.M. Garay, Optimization and the Miranda approach in detecting horseshoe-type chaos by computer. In preparation, available at `www.inf.u-szeged.hu/~csendes/henon2.pdf`

[2] T. Csendes, B.M. Garay, and B. Bánhelyi, A verified optimization technique to locate chaotic regions of a Hénon system. In preparation. Available at `http://www.inf.u-szeged.hu/~csendes/henon.pdf`

[3] Z. Galias and P. Zgliczynski, Computer assisted proof of chaos in the Lorenz equations. Physica D, 115(1998) 165-188.

[4] J.H. Hubbard, The Forced Damped Pendulum: Chaos, Complication and Control. American Mathematical Monthly, 8(1999) 741-758.

[5] O. Knüppel, PROFIL — Programmer's Runtime Optimized Fast Interval Library. Bericht 93.4., Technische Universität Hamburg-Harburg, 1993.

[6] N. Nedialkov, VNODE, A Validated Solver for Initial Value Problems for Ordinary Differential Equations
`http://www.cas.mcmaster.ca/~nedialk/Software/VNODE/VNODE.shtml`

[7] A. Neumaier, and T. Rage, Rigorous chaos verification in discrete dynamical systems, Physica D 67(1993) 327-346.

[8] ODE ARHITECT, an ODE illustration program, J. Wiley & Sons Inc., 1999.

[9] T. Rage, T., A. Neumaier, and C. Schlier, Rigorous verification of chaos in a molecular model, Phys. Rev. E. 50(1994) 2682-2688.

[10] P. Zgliczynski, Computer assisted proof of the horseshoe dynamics in the Hénon map. Random & Computational Dynamics 5(1997) 1-17.