Lyapunov functions by integrating ODEs

Sigurdur Hafstein

Faculty of Physical Sciences, University of Iceland

9. February 2024

Outline

Outline of the talk:

- What is this talk about?
- **Numerical Analysis** course for undergraduates
- Exercises and Projects in general
- The project: Lyapunov functions by integrating ODEs
- Conclusions

What is this talk about?

- Designing good projects involving programming in numerical analysis for large groups of students with different backgrounds is a challenging task.
  - The assignment has to be manageable for the average student.
  - Should have some depth to inspire (some) students and let them to think about the subject.
- We describe a project that was assigned to the students of an introductory Numerical Analysis course at the University of Iceland.
- The assignment is to numerically compute the length of solution trajectories of ODEs with a stable equilibrium point.
  - While not difficult to do, the results are somewhat surprising and got (some) students interested in what was happening.
- We describe the project, its solution using Matlab, and the underlying mathematics in some detail.
- We discuss the pedagogical aspects and the results in terms of success and shortcomings.

The **Numerical Analysis** course

- Second year in three years undergraduate programs (BSc.)
- 6 ECTS units
- 2×80 minutes lectures and 1×80 minutes exercise class per week for 14 weeks
- 120-150 students ($2/3$ male, $1/3$ female) from
  - Math, Applied Math - additional theoretical 2 ECTS course
  - Physics
  - Engineering: Mechanical, Industrial, Chemical, Civil, Envr., Electrical
  - Elective for Geophysics, Computer Science, Software Engineering, Chemistry

The **Numerical Analysis** course

Prerequisites:

- Computer Science - Programming in Matlab or Python
- Linear Algebra
- Analysis I - calculus of one variable
- Analysis II - multivariable calculus

Recommended:

- Analysis III - ODEs, Fourier series and transform, complex analysis

The **Numerical Analysis** course

Course Description:

Fundamental concepts on approximation and error estimates. Solutions of systems of linear and non-linear equations. PLU decomposition. QR decomposition and SVD. Interpolating polynomials, spline interpolation and regression. Numerical differentiation and integration. Extrapolation. Numerical solutions of initial value problems of systems of ordinary differential equations. Multistep methods. Numerical solutions to boundary value problems for ordinary differential equations.

The **Numerical Analysis** course

Learning Outcomes:

Knowledge and understanding: know this and that blah, blah

Skills: *To complete this course the student should be able to*

1. *formulate a simple mathematical problem as a numerical problem, implement it on a computer, and compute an approximate solution,*
2. *estimate the error of numerical solutions,*
3. *use computer software, such as the Anaconda Python platform or Matlab, for programming, computing, and performing numerical experiments,*
4. *validate the results of numerical computations,*
5. *use the concepts and the results of the course to develop and advance algorithms for simple problems the student has not seen before.*

The **Numerical Analysis** course

Theory is covered in the lectures: Newton's Method to compute roots of a functions $f : \mathbb{R} \to \mathbb{R}$, formula $x_{k+1} = x_k - f(x_k)/f'(x_k)$

1. special case of a fixed-point iteration
2. quadratic convergence to simple roots
3. linear convergence to multiple roots
4. no guaranty of convergence
5. show use

Exercises are handed in weekly and are used to train the use of the methods discussed

1. apply Newton's Method to approximate the root of $f(x) = \sin x$ close to 3
2. write a program `r=NewtM(f,x0,tol)` that uses the Newton Method to find a root of the function f with initial value x_0 and returns $r = x_{k+1}$ when $|x_{k+1} - x_k| <$ tol

Exercises and Projects in general

What skills does a student get from studying theory and doing routine exercises?

1. solve real-world problems?
2. do routine exercises?

Solution: Two larger group projects in the course, 30% of the final grade

Exercises and Projects in general

Textbook: Numerical Analysis, Timothy Sauer (several editions usable)
Several **Reality Checks** are in the book, e.g.:

1. kinematics of the Steward Platform (frequently used in flight-simulators)
2. positioning using GPS
3. motion control in computer-aided modeling
4. simple audio codec

Good because:

1. inspiring for (some) students
2. gives sense to the methods studied
3. combination of different methods

The proposed project

I wanted to develop a project to raise the interest of (some) students in my field of research: **Dynamical Systems: Lyapunov functions**
Constraints:

1. not too long
2. not too difficult

Project: numerical study of the van der Pol (VDP) oscillator

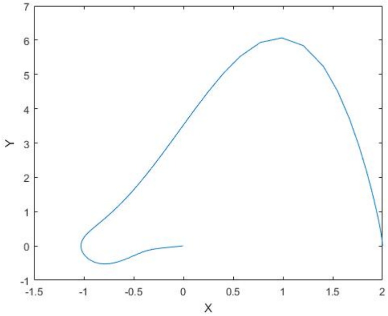
1. unstable equilibrium at the origin
2. stable limit cycle encircling the origin

The proposed project

Consider the ODE (time-reversed VDP)

$$\mathbf{x}' = \mathbf{f}(\mathbf{x}), \quad \text{where} \quad \mathbf{f}(x,y) = \begin{pmatrix} -y \\ -4(1-x^2)y + x \end{pmatrix}$$

Objective I Analyze the ODE by drawing solution trajectories, both forward and backwards in time, using Matlab's `ode45`



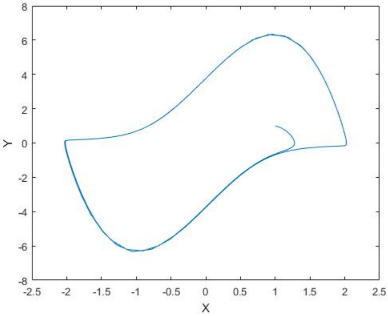


Figure: Left: forward time, Right: backward time

The proposed project

Objective II Implement the standard Runge-Kutta method of order 4 (`RK4`) and again, analyze the ODE by drawing solution trajectories, both forward and backwards in time

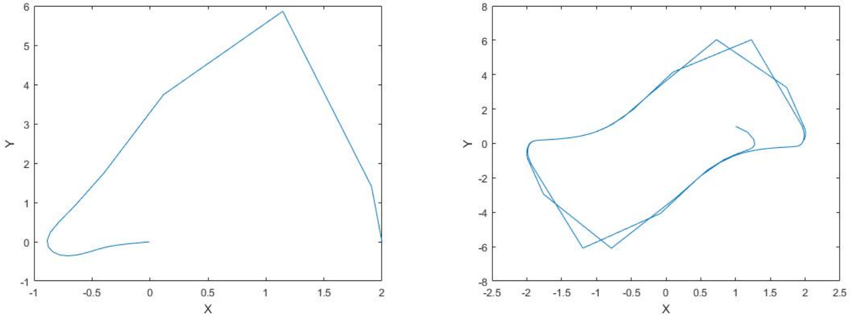


Figure: Left: forward time, Right: backward time

The proposed project

Objective III On a uniform 101×101 grid on $[-3,3] \times [-8,8]$, compute the length of the solution trajectories to the ODE integrated over a time-interval of length $T = 4$ and using `RK4` with 100 time-steps.
For the solution $\mathbf{x}(t)$ to

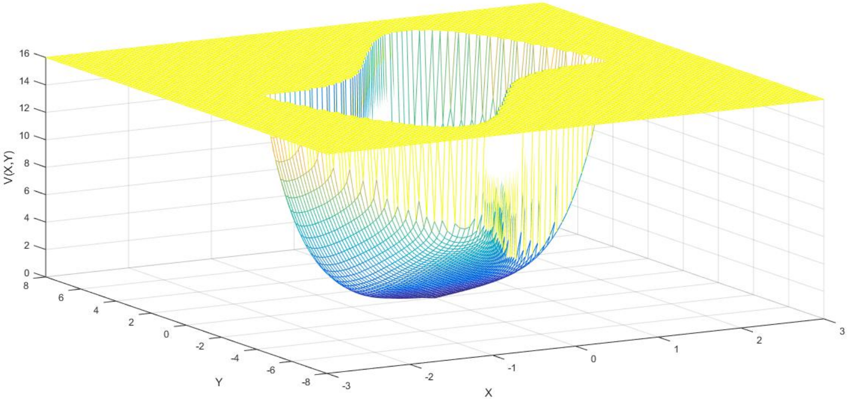
$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t)), \quad \mathbf{x}(0) = \boldsymbol{\xi}$$

the length is

$$L(\boldsymbol{\xi}) = \int_0^T \|\dot{\mathbf{x}}(t)\|_2 dt = \int_0^T \|\mathbf{f}(\mathbf{x}(t))\|_2 dt$$

The proposed project

Objective III On a uniform 101×101 grid on $[-3, 3] \times [-8, 8]$, compute the length of the solution trajectories to the ODE integrated over a time-interval of length 4 and using `RK4` with 100 time-steps.



The proposed project

The project and its solution are discussed in detail in

S.F. Hafstein.
Numerical Analysis Project in ODEs for Undergraduate Students.
In: Computational Science – ICCS 2019 19th International Conference, Faro, Portugal, June 12-14, 2019, Proceedings, Part V Series: Lecture Notes in Computer Science 11540 eds. J. Rodrigues, P. Cardoso, J. Monteiro, R. Lam, V. Krzhizhanovskaya, M. Lees, J. Dongarra, and P. Sloot Springer 2019. pp. 421-434.

The proposed project

The hope was that (some) students got interest in why the length of the trajectories as a function of initial-value looks like that.

1. solutions slow down close to the equilibrium that swallows them
2. the computed function is a Lyapunov function; decreasing along solution trajectories. Its level-sets are forward-invariant

The theory of complete Lyapunov functions tells us that every system given by an ODE possesses a complete Lyapunov function that goes a long way in characterizing the qualitative behaviour of the system. Indeed, this holds true for very general dynamical systems. A complete Lyapunov function is a scalar-valued function from the whole state-space that is non-increasing along along all solution trajectories and strictly decreasing where possible. Note that, e.g. for a periodic orbit, it cannot be strictly decreasing. In general it is strictly decreasing along all solution trajectories on the part of the state-space where the flow is gradient-like and constant on every transitive component of the chain-recurrent set.

Conclusions

The difference between the Reality Checks from the textbook and the proposed project:

1. the Reality Checks inspire (some) students to become interested in the technology behind the project, e.g. flight simulators, GPS, audio codecs, but not in mathematics.
2. the proposed project inspired some of the mathematics students to get interested in the theory of dynamical systems, a topic that they had only known indirectly from studying differential equations

Thank you for listening