Stiff differential equations.

Anne Kværnø

Nov 16, 2018

1 Introduction

When an ODE is solved by an adaptive solver we will expect that more steps are required for stricter tolerances. More specific: The step size control is based on the assumption that the local error estimate $\mathbf{le}_{n+1}$ satisfies

$$\|\mathbf{le}_{n+1}\| \approx D h_n^{p+1} \approx \text{Tol},$$

where p is the order of the lowest order method, and D is independent of the step size h, D depends on the solution point $(x, \mathbf{y})$, but it will not change much from one step to the next.

By solving a problem by some adaptive method, using different tolerances, Tol_1 and Tol_2, we will expect that the corresponding step sizes h_1 and h_2 near the same arbitrary solution point will behave like

$$\text{Tol}_1 \approx D h_1^{p+1}, \qquad \text{Tol}_2 \approx D h_2^{p+1},$$

so that

$$\frac{h_1}{h_2} \approx \left(\frac{\text{Tol}_1}{\text{Tol}_2}\right)^{\frac{1}{p+1}} \approx \frac{N_2}{N_1}.$$

where N_1 and N_2 are the total number of steps used for the two tolerances.

In the case of Heun-Euler $p = 1$. By reducing the tolerance by a factor 1/100 we will expect that the number of steps increases by a factor of 10.

Numerical example 1: Given the following system of 2 ODEs

$$\begin{aligned} y_1' &= -2y_1 + y_2 + 2\sin(x), & y_1(0) &= 2, \\ y_2' &= (a-1)y_1 - a y_2 + a\big(\cos(x) - \sin(x)\big), & y_2(0) &= 3, \end{aligned}$$

where a is some positive parameter. The exact solution, which is independent of the parameter, is

$$y_1(x) = 2e^{-x} + \sin(x), \qquad y_2(x) = 2e^{-x} + \cos(x),$$

check it yourself. Solve this problem with some adaptive ODE solver, for instance the Heun-Euler scheme.

Now try $\text{Tol} = 10^{-2}$, 10^{-4}, 10^{-6}, and do the experiment with two different values of the parameters, $a = 2$ and $a = 999$.

For $a = 2$ the expected behaviour is observed. But the example $a = 999$ requires much more steps, and the step size seems almost independent of the tolerance, at least for $\text{Tol} = 10^{-2}$, 10^{-4}.

The example above with $a = 999$ is a typically example of a *stiff ODE*. When a stiff ODE is solved by some explicit adaptive method an unreasonable large number of steps is required, and this number seems independent of the tolerance. In the remaining part of this note we will explain why this happens, and how we can overcome the problem. For simplicity, the discussion is restricted to linear problems, but also nonlinear ODEs can, and often will be, stiff.

Exercise 1: Repeat the experiment on the Van der Pol equation

$$\begin{aligned} y_1' &= y_2, & y_1(0) &= 2, \\ y_2' &= \mu(1-y_1^2)y_2 - y_1, & y_2(0) &= 0. \end{aligned}$$

Use $\mu = 2$, $\mu = 5$ and $\mu = 50$.

1.1 Linear stability analysis

Motivation. Given a system of m differential equation of the form

$$\mathbf{y}' = A\mathbf{y} + \mathbf{g}(x). \tag{*}$$

Such systems have been discussed in Mathematics 3, and the technique for finding the exact solution will shortly be repeated here:

Solve the homogenous system $\mathbf{y}' = A\mathbf{y}$, that is, find the eigenvalues λ_i and the corresponding eigenvectors $\mathbf{v}_i$ satisfying

$$A\mathbf{v}_i = \lambda_i \mathbf{v}_i, \qquad i = 1, 2, \ldots, m. \tag{**}$$

Assume that A has a full set of linear independent eigenvectors $\mathbf{v}_i$. Let $V = [\mathbf{v}_1, \ldots, \mathbf{v}_m]$, and $\Lambda = \operatorname{diag}\{\lambda_1, \ldots, \lambda_m\}$. In this case V is invertible and

$$AV = V\Lambda \qquad \Leftrightarrow \qquad V^{-1}AV = \Lambda.$$

The ODE (*) can be rewritten by

$$V^{-1}\mathbf{y}' = V^{-1}AVV^{-1}\mathbf{y} + V^{-1}\mathbf{g}(x).$$

Let $\mathbf{z} = V^{-1}\mathbf{y}$ and $\mathbf{q}(x) = V^{-1}\mathbf{g}(x)$ such that the equation can be decoupled into a set of independent scalar differential equations

$$\mathbf{z}' = \Lambda\mathbf{z} + \mathbf{q}(x) \qquad \Leftrightarrow \qquad z_i' = \lambda_i z_i + q_i(x), \quad i = 1, \ldots, m.$$

The solution of such equations has been discussed in Mathematics 1. When these solutions are found, the exact solution is given by

$$\mathbf{y}(x) = V\mathbf{z}(x),$$

and possible integration constants are given by the initial values.

As it turns out, the eigenvalues λ_i are the key to understand the behaviour of the adaptive integrators. So we will discuss the stability properties of the very simplified linear test equation

$$y' = \lambda y.$$

For general problems the eigenvalues and the eigenvectors may be complex. In that case, the solutions will appear in complex conjugate pairs which can be combined to trigonometric functions. In this note, we will for simplicity only consider $\lambda \in \mathbb{R}$. The discussion below is also relevant for nonlinear ODEs, $\mathbf{y}'(x) = \mathbf{f}(x, \mathbf{y}(x))$, in which case λ is considered as an eigenvalue of the Jacobian $\mathbf{f}_\mathbf{y}$ of $\mathbf{f}$ with respect to $\mathbf{y}$.

Example 1: Return to the introductory example. The ODE can be written as

$$\mathbf{y}' = A\mathbf{y} + \mathbf{g}(x),$$

with

$$A = \begin{pmatrix} -2 & 1 \\ a-1 & -a \end{pmatrix}, \qquad \mathbf{g}(x) = \begin{pmatrix} \sin(x) \\ a(\cos(x) - \sin(x)) \end{pmatrix}.$$

The eigenvalues of the matrix A are $\lambda_1 = -1$ and $\lambda_2 = -(a+1)$. The general solution is given by

$$\mathbf{y}(x) = c_1 \begin{pmatrix} 1 \\ 1 \end{pmatrix} e^{-x} + c_2 \begin{pmatrix} -1 \\ a-1 \end{pmatrix} e^{-(a+1)x} + \begin{pmatrix} \sin(x) \\ \cos(x) \end{pmatrix}.$$

In the introductory example, the initial values were chosen such that $c_1 = 2$ and $c_2 = 0$. But for a large, the term $e^{-(a+1)x}$ will go to 0 almost immediatly, even if $c_2 \neq 0$. It is still this term that creates problems for the numerical solution.

Stability functions and stability intervals. Given the linear test equation

$$y' = \lambda y, \qquad y(0) = y_0, \qquad \lambda \in \mathbb{R}, \qquad \lambda < 0,$$

with exact solution

$$y(x) = e^{\lambda x}\, y_0.$$

Since $\lambda < 0$ the solution $y(x) \to 0$ when $x \to \infty$. We want a similar behaviour for the numerical solution, that is $|y_n| \to 0$ when $n \to \infty$. But do we get it?

One step of some Runge–Kutta method applied to the linear test equation can always be written as

$$y_{n+1} = R(z) y_n, \qquad z = \lambda h.$$

The function $R(z)$ is called the *stability function* of the method.

Taking the absolute value on each side of this expression, we see that there are three possible outcomes:

$$\begin{array}{llllll} |R(z)| < 1 & \Rightarrow & |y_{n+1}| < |y_n| & \Rightarrow & y_n \to 0 & \text{(stable)} \\ |R(z)| = 1 & \Rightarrow & |y_{n+1}| = |y_n| & & & \\ |R(z)| > 1 & \Rightarrow & |y_{n+1}| > |y_n| & \Rightarrow & |y_n| \to \infty & \text{(unstable)} \end{array}$$

The *stability interval* of a method is defined by

$$\mathcal{S} = \{z \in \mathbb{R} \,:\, |R(z)| \leq 1\}.$$

To get a stable numerical solution, we have to choose the step size h such that $z = \lambda h \in \mathcal{S}$.

Numerical example 2: Euler's method applied to the linear test equation is

$$y_{n+1} = y_n + h\lambda y_n = (1 + h\lambda)y_n,$$

so the stability function $R(z)$ and the stability interval $\mathcal{S}$ is

$$R(z) = 1 + z, \qquad \mathcal{S} = [-2, 0].$$

Next, solve the introductory problem

$$\mathbf{y}' = \begin{pmatrix} -2 & 1 \\ a-1 & -a \end{pmatrix} \mathbf{y} + \begin{pmatrix} \sin(x) \\ a(\cos(x) - \sin(x)) \end{pmatrix}, \qquad \mathbf{y}(0) = \begin{pmatrix} 2 \\ 3 \end{pmatrix}, \qquad a > 0.$$

by Euler's method. We know that the eigenvalues of the matrix A are

$\lambda_1 = -1$ and $\lambda_2 = -(1 + a)$.

For the numerical solution to be stable for both eigenvalues, we have to require that

$$h \leq \frac{2}{1 + a}.$$

Try $a = 9$ and $a = 999$. Choose step sizes a little bit over and under the stability boundary, and you can experience that the result is sharp. If h is just a tiny bit above, you may have to increase the interval of integration to see the unstable solution.

It is the term corresponding to the eigenvalue $\lambda_2 = -(a + 1)$ which makes the solution unstable. And the solution oscillate since $R(z) < -1$ for $h > 2/(1 + a)$.

Exercise 2:

1. Find the stability function and the stability interval for Heun's method.
2. Repeat the experiment in Example 2 using Heun's mehod.

NB! Usually the error estimation in adaptive methods will detect the unstability and force the step size to stay inside or near the stability interval. This explains the behaviour of the experiment in the introduction of this note.

2 $A(0)$-stable methods.

In an ideal world, we would prefer the stability interval to satisfy $\mathcal{S} \supset \mathbb{R}^-$, such that the method is stable for all $\lambda < 0$ and for all h. Such methods are called $A(0)$-stable. For all explicit methods, like Euler's and Heun's, the stability function will be a polynomial, and $|R(z)| \to \infty$ as $z \to -\infty$. Explicit methods can not be $A(0)$-stable. We have to search among implicit methods. The simplest of those is the implicit, or backward Euler's method, given by

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}).$$

Applied to the linear test equation $y' = \lambda y$:

$$y_{n+1} = y_n + h\lambda y_{n+1} \qquad \Rightarrow \qquad y_{n+1} = \frac{1}{1 - h\lambda} y_n \qquad \Rightarrow \qquad R(z) = \frac{1}{1 - z}.$$

The method is $A(0)$-stable since $|R(z)| \leq 1$ for all $z \leq 0$.

2.1 Implementation of implicit Euler's method

For simplicity, we will only discuss the implementation of implicit Euler's method for linear systems of the form

$$\mathbf{y}' = A\mathbf{y} + \mathbf{g}(x),$$

where A is a constant matrix. In this case, one step of implicit Euler is given by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + hA\mathbf{y}_{n+1} + h\mathbf{g}(x_{n+1}).$$

A linear system

$$(I - hA)\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{g}(x_{n+1})$$

has to be solved with respect to $\mathbf{y}_{n+1}$ for each step.

In the implementation below, the right hand side of the ODE is implemented as a function `rhs`, returning the matrix A and the vector $\mathbf{g}(x)$ for each step. The function `implicit_euler` does one step with implicit Euler. It has the same interface as the explicit method, so that the function `ode_solve` can be used as before.

Numerical example 3: Solve the test equation with

$$A = \begin{pmatrix} -2 & 1 \\ a-1 & -a \end{pmatrix}, \qquad \mathbf{g}(x) = \begin{pmatrix} \sin(x) \\ a(\cos(x) - \sin(x)) \end{pmatrix},$$

by implicit Euler. Choose $a = 2$ and $a = 999$, and try different stepsizes like $h = 0.1$ and $h = 0.01$. Are there any stability issues in this case?

Exercise 2: The trapezoidal rule is an implicit method which for a general ODE $\mathbf{y}'(x) = f(x, \mathbf{y}(x))$ is given by

$$\mathbf{y}_{n+1} = \mathbf{y}_n + \frac{h}{2}\Big(\mathbf{f}(x_n, \mathbf{y}_n) + \mathbf{f}(x_{n+1}, \mathbf{y}_{n+1})\Big).$$

1. Find the stability function to the trapezoidal rule, and prove that it is $A(0)$-stable.
2. Implement the method, and repeat the experiment above.

2.2 Adaptive methods.

Implicit Euler is a method of order 1, and the trapezoidal rule of order 2. Thus, these can be used for error estimation: Do one step with each of the methods, use the difference between the solutions as an error estimate, and use the solution from the trapezoidal rule to advance the solution. This has been implemented in the function `trapezoidal_ieuler`. The interface is as for the embedded pair `heun_euler`, so the adaptive solver `ode_adaptive` can be used as before.

Numerical example 4: Repeat the experiment from the introduction, using `trapezoidal_euler`.

We observe that there are no longer any step size restriction because of stability. The algorithm behaves as expected.

Comment: Implicit methods can of course also be applied for nonlinear ODEs. Implicit Euler's method will be

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}(x, \mathbf{y}_{n+1}),$$

which is a nonlinear system which has to be solved for each step. Similar for the trapezoidal rule. Usually these equations are solved by Newton's method or some simplification of it.

Summary.

Linear test equation:

$$y' = \lambda y, \qquad \lambda < 0.$$

Stability function $R(z)$, given by the method applied to the test problem:

$$y_{n+1} = R(z)y_n, \qquad z = \lambda h.$$

Stability interval $\mathcal{S}$:

$$\mathcal{S} = \{z \in \mathbb{R}, \quad |R(z)| \leq 1\}.$$

$A(0)$-stability:

$$\mathcal{S} \supset \mathbb{R}^-,$$

which is the same as

$$|R(z)| \leq 1 \qquad \text{for all } z \leq 0.$$