Prof. Dr. M. Günther
Igor Kossaczký, M.Sc.
Winter Term 2014/15

Bergische Universität Wuppertal
Fachbereich C – Mathematik und Naturwissenschaften
Angewandte Mathematik / Numerische Analysis

Lab Exercises for Numerical Analysis and Simulation I: ODEs

Laboratory 1 - Consistency and Convergence

Presentation of exercises: 10. - 14. 11. 2014

Each working group (1-3 persons) shall present and explain their programmes. Please contact Igor Kossaczký (`kossaczky@math.uni-wuppertal.de`) for a date arrangement.

We consider the initial value problem (IVP)

$$y'(x) = f\big(x, y(x)\big), \quad y(x_0) = y_0 \qquad (y : \mathbb{R} \to \mathbb{R}^n,\ f : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}^n)$$

and want to solve it numerically in some interval $x \in [x_0, x_{\text{end}}]$. For this purpose, an adaptive method with step size control shall be constructed in this lab exercise.

With the explicit Euler scheme

$$u_1 = u_0 + hf(x_0, u_0)$$

and the explicit Euler scheme with half step size

$$v_1 = v_{\frac{1}{2}} + \tfrac{h}{2} f(x_{\frac{1}{2}}, v_{\frac{1}{2}}) \quad \text{with} \quad v_{\frac{1}{2}} = v_0 + \tfrac{h}{2} f(x_0, v_0),$$

two methods are given to compute new approximations u_{n+1} and v_{n+1} by means of known values. Both methods exhibit consistency order $\mathcal{O}(h)$. By an appropriate combination of these schemes ($w_1 = 2v_1 - u_1$) one achieves the modified Euler scheme according to Collatz

$$w_1 = w_0 + hf\left(x_0 + \frac{h}{2}, w_0 + \frac{h}{2} f(x_0, w_0)\right),$$

which has consistency order $\mathcal{O}(h^2)$.

Task 1: Implement the Euler scheme in a function of the following structure:

```
Function name: euler

Input parameters:
  fname        name of right-hand side f
  x0           starting point
  y0           initial value vector (dimension n)
  h            step size

Output parameters:
  x            independent variable
  y            approximations at point x
```

Implement the explicit Euler scheme with half step size and the modified Euler scheme according to Collatz in separate functions `halfeuler` and `collatz`, which make use of the implemented function `euler`. The input and output parameter shall be the same as for the function `euler`.

Task 2: Furthermore, implement a method for solving the initial value problem with step size control based on the explicit Euler scheme and the modified Euler scheme according to Collatz. Use the estimated local error

$$e := u_1 - w_1 = \mathcal{O}(h^2)$$

and the formula for step size prediction

$$h_{\text{opt}} = h_{\text{used}} \cdot \sqrt{\frac{1}{\text{ERR}}}$$

with the error norm

$$\text{ERR} := \sqrt{\frac{1}{n}\sum_{j=1}^{n}\left(\frac{e_j}{atol + z_j \cdot rtol}\right)^2}\,, \quad \text{where } z_j := \max\{|u_{0,j}|,\, |u_{1,j}|\}\,.$$

Consequently, the step is accepted if ERR < 1 holds.

Include the safety factor $\rho = 0.9$ for scaling the resulting h_{opt}, i.e.

$$h_{\text{new}} = \rho \cdot h_{\text{opt}}.$$

Moreover, use the bounding parameters $\sigma = 0.2$ and $\theta = 5$ for decreasing/increasing the step size to avoid oscillating behavior, meaning the new step size has to fulfill

$$h_{\text{used}} \cdot \sigma \leq h_{\text{new}} \leq h_{\text{used}} \cdot \theta\,.$$

If the step size becomes smaller than $h_{\min} := 10^{-6}$, integration shall be terminated with an according information in the output parameter.

Implement the method with step size control in a separate function

```
Function name: functionname

Input parameters:
  fname        name of right-hand side f
  xspan        vector [x0,xend]
  y0           initial value vector (dimension n)
  h0           initial step size
  rtol         relative tolerance
  atol         absolute tolerance

Output parameters:
  x            vector with points of independent variable
  y            matrix with approximations at points x
  istat        statistics
               istat = [number of evaluations of right-hand side f,
               number of accepted steps, number of rejected steps ]

  idid = 1:   xend was reached
       = 0:   step size too small
```

Test examples:

a) *Scalar ODE*

The initial value problem

$$y'(x) = \frac{y}{1+x^2}, \qquad y(x_0) = y_0$$

exhibits the exact solution

$$y(x) = y_0 \cdot \mathrm{e}^{\arctan(x)-\arctan(x_0)}.$$

Solve the problem in the interval $x \in [-10, 20]$ with initial condition $y(-10) = 1$. Apply step size control with the initial step size $h_0 = 1$. Choose the fixed absolute tolerance $atol = 10^{-6}$, whereas the relative tolerances $rtol = 10^{-l}$ for $l = 3, 4, 5, 6$ should be applied. Compare the four numerical approximations to the exact solution. Try also the method with constant step sizes in the three cases $h = 1, 0.5, 0.1$.

b) *Three-body problem*

We consider a three-body problem with the earth, the moon and a satellite in a two-dimensional space (ξ_1, ξ_2). The unknowns are the position and velocity of the satellite: $y_1 := \xi_1, y_2 := \xi_2, y_3 := \xi_1', y_4 := \xi_2'$. The system of ODEs reads

$$\begin{aligned}
y_1'(x) &= y_3 \\
y_2'(x) &= y_4 \\
y_3'(x) &= y_1 + 2y_4 - (1-\mu)\tfrac{y_1+\mu}{r_1^3} - \mu\tfrac{y_1-1+\mu}{r_2^3} \\
y_4'(x) &= y_2 - 2y_3 - (1-\mu)\tfrac{y_2}{r_1^3} - \mu\tfrac{y_2}{r_2^3}
\end{aligned}$$

with the parameter $\mu = \frac{1}{82.45}$ and

$$r_1 := \sqrt{(y_1+\mu)^2 + y_2{}^2}, \qquad r_2 := \sqrt{(y_1-1+\mu)^2 + y_2{}^2}.$$

Solve the system with the initial values

$$y(0) = (1.2, 0, 0, -1.049358)^\top$$

in the time interval $x \in [0, T]$ with $T := 6.1921693$. Thereby, select the initial step size $h_0 = T/1500$. Apply constant step sizes as well as step size control with the tolerances $rtol = atol = 10^{-6}$. Plot the resulting satellite's orbits in (ξ_1, ξ_2) phase diagrammes. Compare the results of the two simulations.

c) *Lorenz - equation*

Given is the ODE $y' = A(y) \cdot y$ with

$$A(y) = \begin{pmatrix} -\frac{8}{3} & 0 & y_2 \\ 0 & -10 & 10 \\ -y_2 & 28 & -1 \end{pmatrix}$$

the initial value $y_0 = (20, 5, -5)^\top$ and the perturbed initial values $\tilde{y}_0 = (20, 5, -5.1)^\top$. Use $rtol = atol = 10^{-4}$ and the initial step size $h_0 = 0.01$. Visualize the numerical solution for $t \in [0, 10]$ using a phase plot in (y_1, y_2, y_3) (`plot3`). Moreover, plot each component of the solutions corresponding to the different initial values in (t, y_i) to estimate the differences.

Comments and Remarks

We highly recommend using `Matlab` for solving the lab exercises, but you are free to use other high level programming languages, e.g. `Octave`, `Python`, `C`, `C++`, `Fortran`, ...

Remarks for `Matlab`:

- The dimension n of the ODE system can be determined inside the functions by the initial vector `y0` using the command `length` or `size`.
- The evaluation of the varying function `fname` inside the function `euler` can be done using the command `feval`, for example via: `f = feval(fname,x,y)`. Type `doc feval` in `Matlab` for more information.
- If the right-hand side f is given in the file `fname.m`, then the integrator is called by: `[...]=functionname(@fname,...)` or `[...]=functionname('fname',...)`.
- The size of the output parameters `x` and `y` is not known a priori. The vector and the matrix can be enlarged in each step. However, this extension may take a long time for larger sizes. At the beginning, fix `x` and `y` using the command `zeros` to a maximum number of steps, for example 10000. Cut off the unused part at the end of the integration.