KTH Engineering Sciences

Computer Exercise 1

Numerical Solution of Initial Value Problems

In this lab initial value problems (IVPs) are solved numerically and the following items are studied:

- accuracy and stability
- constant stepsize and adaptive (variable) stepsize
- stiff and non-stiff problems
- parameter study of the solutions of a system of ODEs

Part 1: Accuracy of a Runge-Kutta method

Make a numerical experiment to find the order of accuracy of the following Runge-Kutta method:

$$\begin{aligned}
k_1 &= f(t_{k-1}, u_{k-1}),\\
k_2 &= f(t_{k-1} + h, u_{k-1} + hk_1),\\
k_3 &= f(t_{k-1} + h/2, u_{k-1} + hk_1/4 + hk_2/4),\\
u_k &= u_{k-1} + \frac{h}{6}(k_1 + k_2 + 4k_3), \quad t_k = t_{k-1} + h, \quad k = 1, 2, \dots, N.
\end{aligned}$$

Implement the method for van der Pol's differential equation

$$\frac{d^2y}{dt^2} + (y^2 - 1)\frac{dy}{dt} + y = 0, \qquad y(0) = 1, \quad \frac{dy}{dt}(0) = 0.$$

Run the problem with constant stepsizes using $N = 10$, 20, 40, 80, 160 and 320 steps with t in the interval $[0, 1]$. Estimate the error at $t = 1$ by computing $e_N = y_N - y(1)$ for the different N values. Since $y(1)$ is not known exactly, use the approximation $y(1) \approx y_{N_{\max}}$, where $N_{\max} = 320$. Make a `loglog`-plot of $|e_N|$ as a function of h, and estimate the order of accuracy from the graph.

Hint 1: Treat the problem as a system on *vector form*, both when you rewrite the 2nd order differential equation to a system of two first order ODEs and when you program the method.

Hint 2: Be careful to take the correct number of steps to reach $t = 1$. If you get the order of accuracy to be one, there is some mistake in your MATLAB-code!

Part 2: Stability investigation of a Runge-Kutta method

The absolute stability of a numerical method for IVPs is important when we want to solve *stiff* problems. The following ODE-system modeling the kinetics of a set of three reactions, known as Robertson's problem, is studied here:

$$A \to B \quad (k_1) \qquad B + C \to A + C \quad (k_2) \qquad 2B \to B + C \quad (k_3)$$

In the reactions above k_1, k_2 and k_3 denote the *rate constants* of the three reactions. The following set of ODEs describe the evolution of (scaled) concentrations of A, B and C as a function of time t:

$$
\begin{aligned}
\frac{dx_1}{dt} &= -k_1 x_1 + k_2 x_2 x_3, && x_1(0) = 1, \\
\frac{dx_2}{dt} &= k_1 x_1 - k_2 x_2 x_3 - k_3 x_2^2, && x_2(0) = 0, \\
\frac{dx_3}{dt} &= k_3 x_2^2, && x_3(0) = 0.
\end{aligned}
$$

The rate constants have the following values: $k_1 = 0.04$, $k_2 = 10^4$ and $k_3 = 3 \cdot 10^7$.

Constant stepsize experiment

If Robertson's problem is solved with an explicit method the stepsize has to be very small to avoid numerical instability. Use the Runge-Kutta method given in Part 1 on Robertson's problem when the t-interval is $[0, 1]$. Run the problem with constant stepsizes corresponding to $N = 125$, 250, 500, 1000, 2000 steps and find the smallest number of steps (from the 5 given) needed to obtain a stable solution. Plot the solution trajectory in a `loglog`-diagram for the solution computed with the smallest step.

Adaptive stepsize experiment using MATLAB *functions*

There are several IVP-solvers in MATLAB. Use the command `help funfun` to see which are available. To get more information about one of them, say `ode23`, give the command `help ode23`. In order to control e.g. accuracy parameters you also need to read about the function `odeset`. When the problem is stiff you need a stiff IVP-solver, e.g. `ode23s`.

There are several ways to find demo examples in MATLAB. If you give the command `demo` you can study the *Differential Equations* examples under the MATLAB path. You can also directly run examples like `vdpode` and `rigidode` at the prompt. Note that you can look at the code for these demos with the command `type vdpode`, etc.

Make the following numerical experiments on Robertson's problem:

- Use the non-stiff IVP-solver `ode23` on the t-interval $[0, 1]$ for different relative tolerances: `RelTol`$=10^{-3}$, 10^{-4}, 10^{-5}, 10^{-6} and record the number of steps taken by `ode23`. Make a graph of the stepsize h as function of t for one of the tolerances.

- Run the stiff IVP-solver `ode23s` on the t-interval $[0, 1000]$ for the same relative tolerances as above and record the number of steps taken by `ode23s`. Make a graph of the stepsize h as function of t for one of the tolerances.

What conclusions can you draw?

Part 3: Parameter study of solutions of an ODE-system

Make a parameter study for the following two problems taken from applications. Choose a method (order of accuracy must be at least two) yourself. Present the result graphically in a suitable way. Think about the following possibilities and choose what you think is best:

- one or several graphs (using `subplot`) in the figure window?

- linear or logarithmic scales?

- in the graphs: title, x-label, y-label

Comment on the results. Do the systems behave as you expect?

Problem 1: Particle flow past a cylinder

A long cylinder with radius $R = 2$ is placed in an incompressible fluid streaming in the direction of the positive x-axis. The axis of the cylinder if perpendicular to the direction of the flow. The position $(x(t), y(t))$ of a flow particle at time t is determined by the start position $(x(0), y(0))$ and the ODE-system:

$$\frac{dx}{dt} = 1 - \frac{R^2(x^2 - y^2)}{(x^2 + y^2)^2}, \qquad \frac{dy}{dt} = -\frac{2xyR^2}{(x^2 + y^2)^2}.$$

At $t = 0$ there are four flow particles at $x = -4$ with the y-positions 0.2, 0.6, 1.0 and 1.6. Compute and make a graph of the flow curves of the particles in the t-interval $[0, 10]$. Use `axis equal` in the graph!

Problem 2: Motion of a particle

A particle is thrown from the position $(0, H)$ with an elevation angle α and the velocity $v_0 = 20$. The trajectory of the particle is given by the ODE system

$$\frac{d^2x}{dt^2} = -k\frac{dx}{dt}\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2},$$
$$\frac{d^2y}{dt^2} = -9.81 - k\,\frac{dy}{dt}\sqrt{\left(\frac{dx}{dt}\right)^2 + \left(\frac{dy}{dt}\right)^2},$$

where k is the drag coefficient that models the air resistance. The initial data depends on H, α and v_0 as

$$(x(0),\ y(0)) = (0,\ H), \qquad \left(\frac{dx(0)}{dt},\ \frac{dy(0)}{dt}\right) = (v_0 \cos\alpha,\ v_0 \sin\alpha),$$

For two different values of k, say $k = 0.020$ and $k = 0.065$, plot the solution trajectories for $\alpha = 30$, 45 and 60 (degrees). For the graphical presentation, observe that the model is valid only until the particle touches the ground, i.e. it is valid only while $y \geq 0$. The graph should show the motion in a xy-coordinate system with t as a parameter.