Spring fever

The second-order ODE

$$y'' + \gamma y' + 2y = \sin(t)$$

describes the simple harmonic oscillation of (for example) a mass suspended by a spring and subjected to a periodic force. The constant γ models damping due to friction. Neglecting the friction, the natural frequency of oscillation is $\omega_0 = \sqrt{2}$, and the total motion combines the natural oscillation with that of the driving external force. If $\gamma > 0$, however, the unforced solution dies off, and the oscillation is driven entirely by the external force.

One complication in assessing variable-step automatic integrators, such as the RK23 method described in the text, is that we mathematically describe convergence as being at the rate $O(h^p)$ for a fixed step size h. If we look instead at the total number n of steps taken by the method over time interval $[a, b]$, then the "average step size" is $(b-a)/n$ and we can hope for convergence of $O(n^{-p})$.

Goals

You will use the driven spring problem to assess the convergence of the `rk23` function given in the text.

Preparation

Read section 6.5. Write the ODE as the first-order system $\boldsymbol{u}' = \boldsymbol{f}(t, \boldsymbol{u})$. (That is, write out exactly what the function $\boldsymbol{f}$ is.)

Procedure

Download the script template and complete it to perform the following steps. Throughout this lab, use $0 \le t \le 20$ and $y(0) = 0$, $y'(0) = 0$.

1. Consider the ODE as a first-order system, $\boldsymbol{u}' = \boldsymbol{f}(t, \boldsymbol{u})$. In a separate file, write a function

   ```
   function f = myspring(t,u,gamma)
   ```

   that defines the ODE by returning the value of $\boldsymbol{f}$ for any given t and $\boldsymbol{u}$.

2. Define `u0` as the initial condition of the first-order system and let $\gamma = 10$. Create a reference solution using the syntax

   ```
   dudt = @(t,u) myspring(t,u,10);
   opt = odeset('reltol',1e-13,'abstol',1e-13);
   [t,u_ref] = ode15s(dudt,[0 20],u0,opt);
   ```

   where `u0` is defined appropriately. Plot the solution $y(t)$ as a function of t, using a title and labeled axes. In the title or as text on the plot, include the number of time steps that were taken by the solver.

3. Use the adaptive `rk23` solver from the text to solve the problem with the `tol` argument set to 10^{-4}. Make a phase plot of the solution (i.e., with y on the horizontal axis and y' on the vertical axis). Don't forget labels and a title.

4. Use `rk23` to solve the problem with each of the error tolerances 10^{-2}, 10^{-3}, …, 10^{-12} in turn. After each solution, record in two vectors the number of time steps taken, n, and the global error, $E_n = \|\boldsymbol{u}_{\text{ref}}(20) - \boldsymbol{u}(20)\|$. (*Important*: This is at the final time $t = 40$, *not* the solution in the 30th row of the output from `rk23`.) Output a table with each row giving the tolerance, n, and E_n.

5. On a new graph, make a log-log plot of E_n versus n. The points should mostly lie close to a straight line. Add to the plot two straight lines indicating perfect second- and third-order convergence, $E_n = n^{-2}$ and $E_n = n^{-3}$.

6. Now let $\gamma = 5000$. Repeat steps 2–3.

7. Repeat steps 4–5 (still with $\gamma = 5000$). The results will be very different.

Discussion

The time stepping in `rk23` is supposed to be third-order accurate. But that is a statement about the mathematical limit $h \to 0$, or $n \to \infty$. For particular finite values of n, we might observe something close to the ideal $E_n = Cn^{-3}$, but just about anything is possible.

In fact, for this spring problem with $\gamma = 5000$, the time step size adaptation is being constrained by something other than third-order accuracy. The mathematics behind this observation is considered in Chapter 10.