Numerical Solution of ODEs (MTH 452/552)

Lecture notes, January 13, 2017

We discussed the posted Matlab code for the Euler method and discussed some principles of programming, such as putting code that will change into separate files from code that is permanent, and labeling Matlab figures so it is clear what is displayed. The posted code provides an example for these techniques. Please go over the posted source code and make sure you do understand what it does.

Example. Convert the ODE $u'' = -u$ to a first order system.

We let $w_1 = u$ and $w_2 = u'$. Then $w_1' = u' = w_2$ and $w_2' = u''$. Now using the differential equation at hand we have $u'' = -u$, so $w_2' = -u = -w_1$. This gives the first order system

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix}' = \begin{bmatrix} w_2 \\ -w_1 \end{bmatrix}.$$

This system is programmed in the posted M-file myode1.m

Definition. (Big-O notation.) We write $f(k) = O(g(k))$ as $k \to 0$, if there is a constant C such that $|f(k)| \leq C|g(k)|$ for sufficiently small k.

Similarly, we write $F(N) = O(G(N))$ as $N \to \infty$, if there is a constant C such that $|F(N)| \leq |G(N)|$ for sufficiently large N.

In the context of this course we often have $k = \Delta t$ and $g(\Delta t) = (\Delta t)^p$ for some power p.

Definition. If p is the largest number such that the error of the numerical solution converges to zero as fast as $O((\Delta t)^p)$ as $\Delta t \to 0$, we speak of 'convergence of order p'.

Our numerical experiment with the Euler method indicated convergence of order $p = 1$.

Questions like the one in the example below occur very often in practice and need to be well understood.

Example. Question: Assume one knows that the method at hand has convergence of order $p > 0$. For a given Δt one observes an error of E_0. How should Δt be adjusted so that the new error can be expected to stay below a desired error tolerance E_T?

Answer: Let k_0 be the current value of Δt and k the desired new value. Then $|E_0| \simeq Ck_0^P$ and $|E_T| \simeq Ck^p$. It follows that

$$\left|\frac{E_T}{E_0}\right| \simeq \frac{Ck^p}{Ck_0^p} = \frac{k^p}{k_0^p}.$$

Solving for k gives

$$k = \left|\frac{E_T}{E_0}\right|^{\frac{1}{p}} k_0.$$

Measure of computational work. We want to assess the efficiency of numerical methods for solving ODEs of the form $u' = f(t, u)$. The amount of computational work required for a given problem depends on the function $f(t, u)$ which may be simple or complicated. Since we are interested in the general efficacy of a given method, we want a measure that is not as dependent on the particular function f. Therefore we will often use the number of required evaluations of $f(t, u)$ as our measure of computational effort.

Two of the possible ways to decrease the required number of function evaluations are:

1) Using a method with a higher order of convergence that allows for using a larger step size in order to achieve the desired accuracy.

2) Using variable stepsize Δt. Usually one has initial conditions at some time t_0 and is interested in the solution at some time $t_1 > t_0$. If the solution $u(t)$ does not change much for t in some some part of the interval $[t_0, t_1]$, a larger step size can be chosen for this part, than for parts of the interval where the solution changes rapidly.