Week 12: Monday, Apr 11

Problem du jour

What is the region of absolute stability for the trapezoidal rule?

$$y_{n+1} = y_n + \frac{h}{2}\left(f(t_n, y_n) + f(t_{n+1}, y_{n+1})\right)$$

Answer: Applied to the test problem $y' = \lambda y$, the trapezoidal rule gives us

$$y_{n+1} = \left(\frac{2 + h\lambda}{2 - h\lambda}\right) y_n.$$

Numerical solutions decay when $|2 + h\lambda| < |2 - h\lambda|$, i.e. when $\mathrm{Re}(\lambda) < 0$.

Logistics

- I've made a correction to HW 6.
- The prelim isn't graded yet...

Euler and trapezoidal rules

So far, we have introduced three methods for solving ordinary differential equations: forward Euler, backward Euler, and the trapezoidal rule:

$$\begin{aligned}
y_{n+1} &= y_n + hf(t_n, y_n) && \text{Euler} \\
y_{n+1} &= y_n + hf(t_{n+1}, y_{n+1}) && \text{Backward Euler} \\
y_{n+1} &= y_n + \frac{h}{2}\left(f(t_n, y_n) + f(t_{n+1}, y_{n+1})\right) && \text{Trapezoidal}
\end{aligned}$$

Each of these methods is *consistent* with the ordinary differential equation

$$y' = f(t, y).$$

That is, if we plug solutions to the exact equation into the numerical method, we get a small *local error*. For example, for forward Euler we have *consistency of order 1*,

$$\mathcal{N}_h y_h(t_{n+1}) \equiv \frac{y(t_{n+1}) - y(t_n)}{h_n} - f(t_n, y(t_n)) = O(h_n),$$

and for the trapezoidal rule we have second-order consistency

$$\mathcal{N}_h y_h(t_{n+1}) \equiv \frac{y(t_{n+1}) - y(t_n)}{h_n} - f(t_n, y(t_n)) = O(h_n^2).$$

Consistency + 0-stability = convergence

Each of the numerical methods we have described can be written in the form

$$\mathcal{N}_h y^h = 0,$$

where y^h denotes the numerical solution and $\mathcal{N}_h$ is a (nonlinear) difference operator. If the method is consistent of order p, then the true solution gives a small residual error as h goes to zero:

$$\mathcal{N}_h y = O(h^p).$$

We would like to conclude that since $\mathcal{N}_h y$ and $\mathcal{N}_h y^h$ are both small, y and y^h are close together. But in order to show this, we need one more property: *0-stability*.

A method is zero-stable if there are constants h_0 and K so for any mesh functions x^h and z^h on an interval $[0, T]$ with $h \leq h_0$,

$$d_n = |x_n - z_n| \leq K \left\{ |x_0 - z_0| + \max_{1 \leq j \leq N} \left| \mathcal{N}_h x^h(t_j) - \mathcal{N}_h z^h(t_j) \right| \right\}$$

for $1 \leq n \leq N$. Zero stability essentially says that the difference operators $\mathcal{N}_h$ can't become ever more singular as $h \to 0$: they are invertible, and the inverse is bounded by K.

If a method is consistent and zero stable, then the error at step n is

$$|y(t_n) - y^h(t_n)| = |e_n| \leq K \max_j |d_j| = O(h^p).$$

The proof is simply a substitution of y and y^h into the definition of zero stability. The only tricky part, in general, is to show that the method is zero stable. Let's at least do this for forward Euler, to see how it's done — but you certainly won't be required to describe the details of this calculation on an exam!

We assume without loss of generality that the system is autonomous ($y' = f(y)$). We also that f is *Lipschitz continuous*; that is, there is some L so that for any x and z,

$$|f(x) - f(z)| \leq L|x - y|.$$

It turns out that Lipschitz continuity of f plays an important rule not only in the numerical analysis of ODEs, but in the theory of existence and uniqueness of ODEs as well: if f is not Lipschitz, then there might not be a unique solution to the ODE. The standard example of this is $u' = 2\,\mathrm{sign}(u)\sqrt{|u|}$, which has solutions $u = \pm t^2$ that both satisfy the ODE with initial condition $u(0) = 0$.

We can rearrange our description of $\mathcal{N}_h$ to get

$$\begin{aligned} x_{n+1} &= x_n + hf(x_n) + \mathcal{N}_h[x](t_n) \\ z_{n+1} &= z_n + hf(z_n) + \mathcal{N}_h[z](t_n). \end{aligned}$$

Subtract the two equations and take absolute values to get

$$|x_{n+1} - z_{n+1}| \leq |x_n - z_n| + h|f(x_n) - f(z_n)| + |\mathcal{N}_h[x](t_n) - \mathcal{N}_h[z](t_n)|$$

Define $d_n = |x_n - z_n|$ and $\theta = \max_j |\mathcal{N}_h[x](t_j) - \mathcal{N}_h[z](t_j)|$. Note that by Lipschitz continuity, $|f(x_n) - f(z_n)| < Ld_n$; therefore,

$$d_{n+1} \leq (1 + hL)d_n + h\theta.$$

Let's look at the first few steps of this recurrence inequality:

$$\begin{aligned} d_1 &\leq (1 + hL)d_0 + h\theta \\ d_2 &\leq (1 + hL)^2 d_0 + [(1 + hL) + 1]\, h\theta \\ d_3 &\leq (1 + hL)^3 d_0 + \left[(1 + hL)^2 + (1 + hL) + 1\right] h\theta \end{aligned}$$

In general, we have

$$\begin{aligned} d_n &\leq (1 + hL)^n d_0 + \left[\sum_{j=0}^{n-1} (1 + hL)^j\right] h\theta \\ &\leq (1 + hL)^n d_0 + \left[\frac{(1 + hL)^j - 1}{(1 + hL) - 1}\right] h\theta \\ &\leq (1 + hL)^n d_0 + L^{-1}\left[(1 + hL)^j - 1\right] \theta \end{aligned}$$

Now note that

$$(1 + hL)^n \leq \exp(Lnh) = \exp(L(t_n - t_0)) \leq \exp(LT),$$

where T is the length of the time interval we consider. Therefore,

$$d_n \leq \exp(LT)d_0 + \frac{\exp(LT) - 1}{L} \max_j |\mathcal{N}_h[x](t_j) - \mathcal{N}_h[z](t_j)|.$$

While you need not remember the entire argument, there are a few points that you should take away from this exercise:

1. The basic analysis technique is the same one we used when talking about iterative methods for solving nonlinear equations: take two equations of the same form, subtract them, and write a recurrence for the size of the differenc.

2. The Lipschitz continuity of f plays an important role. In particular, if LT is large, $\exp(LT)$ may be very inconvenient, enough so that we have to take very small time steps to get good error results according to our theory.

As it turns out, in practice we will usually give up on global accuracy bounds via analyzing Lipschitz constant. Instead, we will use the same sort of local error estimates that we described when talking about quadrature: look at the difference between two methods that are solving the same equation with different accuracy, and use the difference of the numerical methods as a proxy for the error. We will discuss this strategy — and more sophisticated Runge-Kutta and multistep methods — next week.