

MAT 228B Notes

Sam Fleischer

January 18, 2016

1 ODE Solvers

Given an ODE $y' = f(y)$ with $y(0) = y_0$, an ODE Method returns a sequence $y^n \approx y(n\Delta t)$ with $y^0 = y_0$. Apply the method $y' = \lambda y$ for $\lambda \in \mathbb{C}$. Let $z = \lambda\Delta t$. z is in the *region of absolute stability* of the method if $y^n \rightarrow 0$ as $n \rightarrow \infty$.

1.1 Region of Absolute Stability for Forward-Euler

So $\frac{y^{n+1} - y^n}{\Delta t} = \lambda y^n$. This gives

$$\begin{aligned}y^{n+1} &= y^n + \Delta t \lambda y^n \\y^{n+1} &= (1 + z)y^n\end{aligned}$$

So $y^n = (1 + z)^n y^0$. Thus z is in the region of absolute stability, i.e. $y^n \rightarrow 0$, whenever $|1 + z| < 1$. On the real line, this corresponds to $z \in (-2, 0)$. So $-2 < \lambda\Delta t < 0$. For a given λ , this gives a timestep restriction. In the complex plane, this is the unit open ball around -1 , that is, $B((-1, 0))$. Ensuring that you are in the region of absolute stability ensures that something that should decay actually does decay.

1.2 Forward-Euler for Diffusion

1D, Dirichlet homogeneous boundary conditions, i.e. $u_t = Du_{xx}$ with $u(0, t) = u(1, t) = 0$ and $u(x, 0) = f(x)$.

$$\frac{u^{n+1} - u^n}{\Delta t} = \frac{D}{(\Delta x)^2} (u_{j-1}^n - 2u_j^n + u_{j+1}^n) = Lu^n$$

where L is some matrix. For this problem, we can just take the largest eigenvalue to get the maximum timestep. But this doesn't work in general since the regions of absolute stability won't be as nice.

We require $|1 + \Delta t \lambda| < 1$ for all eigenvalues λ of L . The eigenvalues of L are

$$\lambda_k = \frac{2D}{(\Delta x)^2} (\cos(k\pi\Delta x) - 1) = -\frac{4D}{(\Delta x)^2} \sin^2\left(\frac{k\pi\Delta x}{2}\right), \quad k = 1, \dots, N$$

where $\Delta x = \frac{1}{N+1}$. These are all negative and real. So the largest (in magnitude) eigenvalue is $\lambda_N \approx -\frac{4D}{(\Delta x)^2}$. So we are going to require that $-1 < 1 + \lambda_N \Delta t < 1$, so

$$\begin{aligned}-2 &< \frac{4D\Delta t}{(\Delta x)^2} < 0 \\0 &< \Delta t < \frac{(\Delta x)^2}{2D}\end{aligned}$$

This is the stability restriction for forward-Euler for the diffusion equation.

1.3 Forward-Euler for Advection with Centered-Difference

$$\begin{aligned}\frac{u_j^n - u_j^{n-1}}{\Delta t} + a \frac{u_{j+1}^n - u_{j-1}^n}{2\Delta x} &= 0 \\ \implies \frac{u_j^n - u_j^{n-1}}{\Delta t} &= Au^n\end{aligned}$$

for some matrix A . Compute the eigenvalues of A . the eigenvalues of the centered-difference operators on period domains are complex exponentials. The k th eigenfunction at the j th gridpoint is

$$v_j^k = \exp[\pi i k x_j]$$

Plugging this into the centered difference operator gives

$$\frac{\exp[i\pi k x_{j+1}] - \exp[i\pi k x_{j-1}]}{2\Delta x}$$

Note $x_{j+1} = (j+1)\Delta x = x_j + \Delta x$. So,

$$\frac{\exp[i\pi k x_{j+1}] - \exp[i\pi k x_{j-1}]}{2\Delta x} = \frac{\exp[i\pi k (x_j + \Delta x)] - \exp[i\pi k (x_j - \Delta x)]}{2\Delta x} = \underbrace{\left(\frac{e^{i\pi k \Delta x} - e^{-i\pi k \Delta x}}{2\Delta x} \right)}_{\text{eigenvalue}} e^{i\pi k x_j}$$

So the eigenvalues of the centered-difference operator are

$$\lambda_k = \frac{i \sin(\pi k \Delta x)}{\Delta x}$$

So for whatever k , λ_k is always on the imaginary axis, which is *always* outside the region of absolute stability. Might this be tolerable? If you run it and cut it off soon enough it might not be that big of a deal. But if you run it long enough, you are guaranteed to blow up.

1.4 Stiff Problems

The diffusion equation is numerically stiff. The negative real part eigenvalues with a large ratio (?). There are some negative eigenvalues and some big negative eigenvalues.

$$\frac{dy}{dt} = \lambda y \tag{1}$$

where λ is negative and real. It's a decay rate. For diffusion,

$$\lambda_k = -\frac{2D}{(\Delta x)^2} (\cos(k\pi \Delta x) - 1) \tag{2}$$

The smallest are $\mathcal{O}(1)$ and the largest are $\mathcal{O}\left(\frac{1}{(\Delta x)^2}\right)$. So there are $\mathcal{O}(1)$ timescales and $\mathcal{O}\left(\frac{1}{(\Delta x)^2}\right)$ timescales. With forward-Euler, you have to resolve all the time-scales (?).