Math 405: Topic 7a: PDEs and the method-of-lines

Forward Euler applied to the heat/diffusion equation $u_t = u_{xx}$: $v^{n+1} = v^n + kLv^n$. Here L is our favourite "1 -2 1" matrix. This scheme can be represented with a *finite difference stencil*:

```
t          o (j,n+1)
^          |
|          |
|    o------o------o
|         (j,n)    (j+1,n)
+----> x
```

Backward Euler applied to $u_t = u_{xx}$: $v^{n+1} = v^n + kLv^{n+1}$

```
                 (j+1,n+1)
     o------o------o
            |
            |
            o (j,n)
```

These are "full discretizations" of the PDE.

The Method of Lines

Alternatively, we can discretize in space first. To continue with the heat equation example, discretize in space with our favourite "1 -2 1" matrix.

This gives a large system of ODEs $\vec{v}(t)' = L\vec{v}(t)$. We can then apply time-stepping methods (*and analysis*) to the ODEs.

This idea is the "**method of lines**". The "lines" refers to the vertical lines $x = x_j$ at each grid point in a space-time diagram; we solve an ODE for $v_j(t)$ along each line (and each ODE is typically coupled to some neighbours).

Stability in PDE finite difference calculations

Time-step restrictions for the heat equation

The choice of h may effect k. We looked at eigenvalues of the L matrix before: they depend on h. Largest magnitude is $-4/h^2$. Need this inside the stability region.

Choose forward Euler: need $\lambda k > -2$. Leads to restriction on k for stability in time:

$$k \leq \frac{1}{2}h^2,$$

(for the heat equation in *this* dimension, with *this* spatial discretization, with *this* time discretization.)

The biharmonic problem $u_t = -u_{xxxx}$

How to discretize? Think $(u_{xx})_{xx}\ldots$, this leads to

$$v_j(t)' = -\frac{1}{h^4}\left(v_{j-2}(t) - 4v_{j-1}(t) + 6v_j(t) - 4v_{j+1}(t) + v_{j+2}(t)\right)$$

[demo_07_biharmonic.m] Note ridiculously small time steps required. Let's try to see why (stability issue) and what we can do about it (implicit A-stable ODE methods).

PDE stability in method-of-lines

As above, we follow the linear stability analysis for the ODE methods. The spatial discretization gives us the eigenvalues of the *semidiscrete* system (or we may be able to compute them numerically). Need these eigenvalues to lie inside the absolute stability region of the ODE method.

Note: this involves the eigenvalues of the semidiscrete system, not the original right-hand-side of the PDE!

Demo: run demo_07_biharmonic, then use 'eigs' to compute 'largest magnitude' eigenvalues of the *discretized* biharmonic operator: need k times these less than 2 for forward Euler stability. Note this gives almost the same restriction as observed in practice.

von Neumann stability analysis for the fully discrete problem

Another commonly used approach to stability in PDE problems is *von Neumann Analysis* of the finite difference formula. Also known as *discrete Fourier analysis*, invented in the late 1940s.

Consider $u_t = -u_{xxxx}$ again, discretized in space as above and with forward Euler in time:

$$v_j^{n+1} = v_j^n - \frac{k}{h^4}\left(v_{j-2}^n - 4v_{j-1}^n + 6v_j^n - 4v_{j+1}^n + v_{j+2}^n\right).$$

Suppose we have periodic boundary conditions and that at step n we have a (complex) sine wave $e^{i\xi x}$ for some wave number ξ, so that

$$v_j^n = \exp(i\xi x_j) = \exp(i\xi jh),$$

Note larger ξ is more oscillatory. We will analyze whether this wave grows in amplitude or decays (for each ξ). For stability, we want all waves to decay.

For the biharmonic diffusion equation, we substitute this wave into the finite difference scheme above, and factor out $\exp(i\xi h)$ to get

$$v_j^{n+1} = g(\xi)v_j^n,$$

with the *amplification factor*

$$g(\xi) = 1 - \frac{k}{h^4}\left(e^{-i2\xi h} - 4e^{-i\xi h} + 6 - 4e^{i\xi h} + e^{i2\xi h}\right).$$

This can be simplified to:

$$g(\xi) = 1 - \frac{16k}{h^4}(\sin(\xi h/2))^4.$$

As ξ ranges over various values sin is bounded by 1 so we have

$$1 - 16k/h^4 \leq g(\xi) \leq 1.$$

A mode will grow if $|g(\xi)| > 1$. Thus for stability we want $|g(\xi)| \leq 1$ for all ξ , i.e.,

$$1 - 16k/h^4 \geq -1, \qquad \text{or} \qquad \boxed{k \leq h^4/8.}$$

For $h = 0.025$, as in the demo code, this gives

$$k \leq 4.883 \times 10^{-8}.$$

This matches our experiment and ODE analysis convincingly, but confirms that this finite difference formula is not really practical: should use implicit.

Stability, Consistency, Convergence for PDE finite difference methods

Another version of our fundamental theorem:

Lax equivalence theorem: for linear PDEs, consistency + Lax-Richtmyer stability implies convergence.

Lax-Richtmyer Stability: Suppose we have a fixed relationship between k and h (e.g., $k = 0.4h^2$). Say we can write our fully-discrete system as

$$U^{n+1} = B(k)U^n + b^n(k)$$

where $B(k)$ is a matrix. This is the case for our heat example with "1 -2 1" and forward Euler for example.

Defn: Linear method in this form is *Lax–Richtmyer* stable if

$$||B(k)^n|| \le C_T.$$

where C_T constant indep of k, n but could depend on final time T.

MOL Example: Fisher-KPP Equation

Independent 1937 discoveries for biological applications (spread of species): Fisher; Kolmogorov, Petrovsky, and Piscounov:

$$u_t = \epsilon u_{xx} + u - u^2.$$

Solutions: traveling waves. Explicit finite difference model is similar to heat equation. Note have nonhomogeneous BC $u(0) = 1$, and $u(20) = 0$; the former is implemented by using an extra vector "BC". [demo_07_fisher_kpp.m]

```
     | u  |           |-2   1            | | u  |     |1/h^2|
     |  1 |           |                  | |  1 |     |     |
     | u  |           | 1  -2   1        | | u  |     |  0  |
     |  2 |           |                  | |  2 |     |     |
d    | .  |           |    .   .   .     | | .  |     |  .  |          2
--   | .  |  = ep/h^2 |     .   .   .    | | .  |  +  |  .  |  +  (u -u )
dt   | .  |           |      .   .   .   | | .  |     |  .  |        j  j
     |    |           |                  | |    |     |     |
     |    |           |       1   -2   1 | |    |     |     |
     | u  |           |                  | | u  |     |     |
     |  N |           |           1   -2 | |  N |     |  0  |

                                                      BC vector
```

(Note the nonlinear term "$u - u^2$" will need to be entered in Octave/Matlab using ".*" and ".^" etc).

Accuracy on heat equation

Both Forward and Backward Euler with the L matrix are consistent with an expected error of $O(k) + O(h^2)$ (space error + time error).

But with FE: $k = O(h^2)$ so error is $O(h^2) + O(h^2) = O(h^2)$. Maybe you only want first-order accuracy, is so, this extra work is wasteful. (Yet another "definition" of stiffness: if your choice of timestep k is motivated by stability rather than accuracy, you are probably dealing with a stiff problem.)

Higher-order in time

Even if we want second-order perhaps there might be better ways, use a more accurate ODE solver: trapezoidal rule in time + second order in space. When used on heat equation, this is called "Crank–Nicolson":

$$v^{n+1} = v^n + \frac{k}{2} L v^{n+1} + \frac{k}{2} L v^n.$$

or $Bv^{n+1} = Av^n$ for some matrices B and A.

Note the stencil of this scheme:

```
            (j+1,n+1)
o-------o-------o
        |
        |
o-------o-------o
      (j,n)   (j+1,n)
```

Caution

Sometimes hard to tell from numerical convergence study which terms are dominating. Can also do tests to isolate the error components in h and k. For example, try to find exact (or almost exact) solution of the MOL ODE system and compare the fully-discrete solution against that.