Time integration

Issues

- Accuracy
- Cost
  - CPU = cost/step×#steps,
  - storage,
  - programmer's time
- Stability

Spatial discretisation (typically FE or Spectral)

$$\rightarrow \qquad u_t = F(u, t)$$

- Treat by black-box time-integrator
- OR recognise spatial structure (typically only for FD)

Stability in time

1. Unstable algorithm – bad!
   – numerics blow up all Δt, usually rapidly, often oscillates
2. Conditionally stable – normal
   – stable if Δt not too big
3. Unconditionally stable – slightly dangerous
   – stable all Δt, inaccurate large Δt

'Stable' = ?

(i) numerics decays, even if physics does not

(ii) numerics do not blow up for all t

(iii) numerics do not blow up much, i.e. converge fixed t
e.g. need $\Delta t < a + b/t$

Lax equivalence theorem

For a well-posed linear problem,

a consistent approximation (local error $\rightarrow 0$ as $\Delta t \rightarrow 0$)

converges to the correct solution

if and only if the algorithm is stable

Stiffness, for $u_t = F(u, t)$

How do small disturbances grow/decay?
Linearise + freeze coefficients – occasionally wrong

$$\delta u_t = F'(u_0, t_0)\delta u$$

Find eigenvalues λ of $F'(u_0, t_0)$

Stiff if $\lambda_{\max} \gg \lambda_{\min}$, typically by 10^4

Stability controlled by largest $|\lambda|$, need

$$\Delta t < \frac{\mathrm{const}}{|\lambda|_{\max}}$$

– may represent boring time behaviour on fine scales
If so, use unconditionally stable algorithm with big Δt and inaccurate rending of boring fine details

Forward Euler – 1st order, explicit

For $u_t = \lambda u$

$$\frac{u^{n+1} - u^n}{\Delta t} = \lambda u^n$$

Hence

$$u^{n+1} = (1 + \lambda \Delta t)^{n = t/\Delta t} u^1$$
$$\to e^{\lambda t} u^1 \quad \text{as } \Delta t \to 0$$

Case λ real and negative: stable if $\Delta t < \frac{2}{|\lambda|}$

Forward Euler – 1st order, explicit

Case λ purely imaginary

$$|1 + \lambda \Delta t| = \left(1 + |\lambda|^2 \Delta t^2\right)^{1/2} > 1 \quad \text{all } \Delta t$$

so "unstable"

Now

$$\left(1 + |\lambda|^2 \Delta t^2\right)^{t/2\Delta t} \quad \xrightarrow{\Delta t \to 0} \quad e^{\frac{1}{2}|\lambda|^2 \Delta t\, t}$$

i.e. does not blow up much (ϵ) if

$$\Delta t < \frac{2 \ln \epsilon}{\lambda|^2 t}$$

Backward Euler – 1st order, implicit

For $u_t = \lambda u$

$$\frac{u^{n+1} - u^n}{\Delta t} = \lambda u^{n+1}$$

So

$$u^n = \left(\frac{1}{1 - \lambda \Delta t}\right)^n u_0$$

Very stable   just unstable in $|1 - \lambda \Delta t| < 1$

But inaccurate if Δt large
E.g. λ real and negative & large $\Delta t = 1/|\lambda|$ gives

$$u(t) \sim e^{\lambda t \ln 2} \quad \text{cf} \quad e^{\lambda t}$$

Mid-point Euler – 2nd order, explicit

Simple to recode the first-order Forward Euler to make second-order

$$\frac{u^* - u^n}{\frac{1}{2}\Delta t} = F(u^n, t_n)$$
$$\frac{u^{n+1} - u^n}{\Delta t} = F(u^*, t_{n+\frac{1}{2}})$$

Same stability as Forward Euler

Crank-Nicolson – 2nd order implicit

For $u_t = \lambda u$

$$\frac{u^{n+1} - u^n}{\Delta t} = \lambda \frac{u^{n+1} + u^n}{2}$$

NB: RHS uses unknown u^{n+1}, not a problem for this simple linear problem. Solution

$$u^n = \left(\frac{1 + \frac{1}{2}\lambda\Delta t}{1 - \frac{1}{2}\lambda\Delta t} \right)^n u^0$$

Case $Re(\lambda) < 0$ stable all Δt

Case λ imaginary amplitude correctly constant all Δt although phase drifts

Leap frog - 2nd order, explicit

$$\frac{u^{n+1} - u^{n-1}}{2\Delta t} = \lambda u^n$$

Two-term recurrence relation

$$u^{n+1} - 2\lambda\Delta t u^n - u^{n-1} = 0$$

has solutions $u^n = A\theta_+^n + B\theta_-^n$ with $\theta_\pm = \lambda\Delta t \pm \sqrt{1 + \lambda^2\Delta t^2}$

So

$$u^n \sim e^{\lambda n\Delta t} + \epsilon(-1)^n e^{-\lambda n\Delta t}$$

Spurious solution blows up if $Re(\lambda) < 0$

But stable for purely imaginary λ & $\Delta t < 1/|\lambda|$

Runge-Kutta

E.g. standard 4th order RK, for $u_t = F(u, t)$

$$\begin{aligned}
du^1 &= \Delta t F(u^n, t^n) \\
du^2 &= \Delta t F(u^n + \tfrac{1}{2}du^1, t^n + \tfrac{1}{2}\Delta t) \\
du^3 &= \Delta t F(u^n + \tfrac{1}{2}du^2, t^n + \tfrac{1}{2}\Delta t) \\
du^4 &= \Delta t F(u^n + 1du^3, t^n + 1\Delta t) \\
u^{n+1} &= u^n + \tfrac{1}{6}(du^1 + 2du^2 + 2du^3 + du^4)
\end{aligned}$$

NB: 4 function calls per step – very expensive

Can vary Δt after each step – adaptive

Good stability, need $\Delta t \lesssim \frac{3}{|\lambda|}$

Error control for RK4

Take 2 steps of Δt from u^n

$$u^{n+2} = A + 2b\Delta t^5 + \dots$$

Take 1 step of $2\Delta t$ from u^n

$$u^* = A + b(2\Delta t)^5 + \dots$$

Extrapolating, 5th order estimate of answer

$$\frac{16}{15}u^{n+2} - \frac{1}{15}u^*$$

Estimate of error

$$\tfrac{1}{30}(u^* - u^{n+2})$$

– decide if to decrease/increase Δt

Implicit Runge-Kutta

$$du^1 = \Delta t F\left(u^n + \tfrac{1}{4}du^1 + (\tfrac{1}{4} - \tfrac{\sqrt{3}}{6})du^2, t^n + (\tfrac{4}{1} - \tfrac{\sqrt{3}}{6})\Delta t\right)$$
$$du^2 = \Delta t F\left(u^n + (\tfrac{1}{4} + \tfrac{\sqrt{3}}{6})du^1 + \tfrac{1}{4}du^2, t^n + (\tfrac{1}{4} + \tfrac{\sqrt{3}}{6})\Delta t\right)$$
$$u^{n+1} = u^n + \tfrac{1}{2}du^1 + \tfrac{1}{2}du^2$$

Iterate to find du^1 and du^2 – very expensive

Stable all Δt if $Re(\lambda) \leq 0$

Multi-step methods – use information from previous steps

AB3 Adams-Bashforth, 3rd order, explicit

$$u^{n+1} = u^n + \frac{\Delta t}{12}(23F_n - 16F_{n-1} + 5F_{n-2})$$

AM4 Adams-Moulton, 4th order, implicit

$$u^{n+1} = u^n + \frac{\Delta t}{24}(9F_{n+1} + 19F_n - 5F_{n-1} + F_{n-2})$$

NB uses 1 function evaluation per step – good
NB difficult to start or change step size Δt – bad
NB Stable $\Delta t \lesssim 1/|\lambda|$

Predictor-corrector
AB3 sufficiently good estimate for u^{n+1} to use in AM4 F_{n+1},
but then 2 function evaluations per step

Sympletic integrators

For Hamiltonian (non-dissipative) systems

$$\dot{p}_i = -\frac{\partial H}{\partial q_i}, \qquad \dot{q}_i = \frac{\partial H}{\partial p_i}$$

conserve H and projections of volume of phase-space
NB Important for integration to long times.

Sympletic integrators have same conservations properties for a numerical approximation to the Hamiltonian $H^{\mathrm{num}}(\Delta t)$

NB must keep Δt fixed

E.g. Störmer-Verlet (sort of leap-frog) – for molecular dynamics

$$p^{n+\frac{1}{2}} = p^n + \tfrac{1}{2}\Delta t F(r^n)$$
$$r^{n+1} = r^n + \Delta t \tfrac{1}{m} p^{n+\frac{1}{2}}$$
$$p^{n+1} = p^{n+\frac{1}{2}} + \tfrac{1}{2}\Delta t F(r^{n+1})$$

Navier-Stokes – different methods for different terms

For $u_t + uu_x = u_{xx}$ (no pressure, yet)

$$\frac{u^{n+1} - u^n}{\Delta t} = -(uu_x)^{n+\frac{1}{2}} + \frac{u_{i+1}^{n+1} - 2u_i^{n+1} + u_{i-1}^{n+1} + u_{i+1}^n - 2u_i^n + u_{i-1}^n}{2\Delta x^2}$$

implicit on diffusion for stability at boring fine scales

AB3 explicit on safe advection

$$(uu_x)^{n+\frac{1}{2}} = \tfrac{1}{12}\left(23\,(uu_x)^{n-\frac{1}{2}} - 16\,(uu_x)^{n-\frac{3}{2}} + 5\,(uu_x)^{n-\frac{5}{2}}\right)$$

Iserles Zig-Zag – 2nd order and sort of upwinding

$$(uu_x)^{n+\frac{1}{2}} = \frac{u_i^{n+1} + u_i^n}{2}\left(\frac{u_{i+1}^{n+1} - u_i^{n+1}}{2\Delta x} + \frac{u_i^n - u_{i-1}^n}{2\Delta x}\right) \quad \text{if} \quad u_i^n > 0$$

Lagrangian methods in $\mathbf{u}\cdot\nabla\mathbf{u}$ dominant

Pressure update - 2nd order, exact projection to $\nabla \cdot \mathbf{u} = 0$

Split time-step

$$\frac{u^* - u^n}{\Delta t} = -\left(u u_x\right)^{n+\frac{1}{2}} - \nabla p^{n-\frac{1}{2}} + \nu \nabla^2 \left(\frac{u^* + u^n}{2}\right)$$

Projection

$$u^{n+1} = u^* + \Delta t \nabla \phi^{n+1}$$

with

$$\nabla^2 \phi^{n+1} = -\nabla \cdot u^* / \Delta t \qquad \text{with BC} \quad \Delta t \frac{\partial \phi^{n+1}}{\partial n} = u_n^{\mathrm{BC}} - u_n^*$$

Update

$$\nabla p^{n+\frac{1}{2}} = \nabla p^{n-\frac{1}{2}} - \nabla \left(\phi^{n+1} - \tfrac{1}{2}\nu \Delta t \nabla^2 \phi^{n+1}\right)$$

Tangential BC

$$u^*_{\mathrm{tang}} = u^{\mathrm{BC}}_{\mathrm{tang}} - \Delta t \nabla \phi^n$$