LECTURE 16: ORDINARY DIFFERENTIAL EQUATIONS

- ODE: higher order differential equations can always be rewritten as a series of 1st order:

$$\frac{d^2y}{dx^2} + q(x)\frac{dy}{dx} = r(x)$$

$$\frac{dy}{dx} = z(x)$$

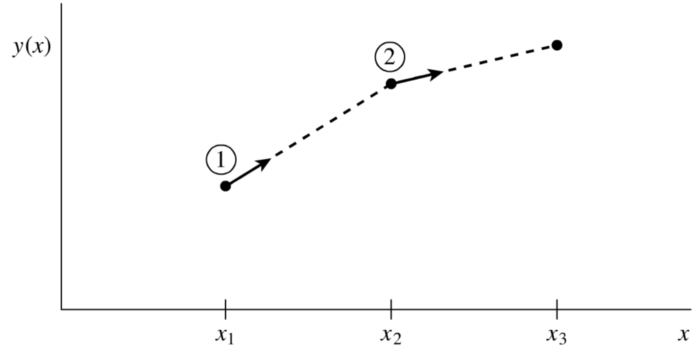
$$\frac{dz}{dx} = r(x) - q(x)z(x)$$

- We also need to specify boundary conditions. Typical case is initial value problem: we specify at initial time. For example, specify initial position and velocity of a particle and then use Newton's law to solve for its time evolution

Euler Method, 2nd Order Midpoint …

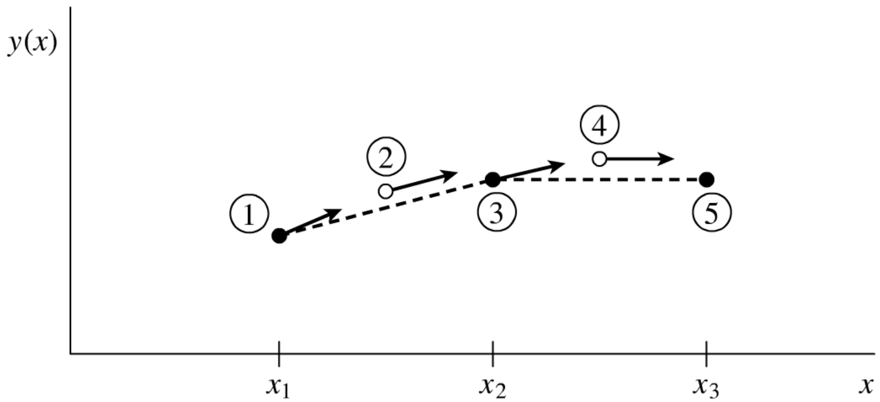
- We start with the simplest method, 1st order (explicit) Euler: *dy/dx = f(x,y), dx = h*

- $$y_{n+1} = y_n + hf(x_n, y_n)$$



- 2nd order extension (midpoint, or 2nd order Runge-Kutta)

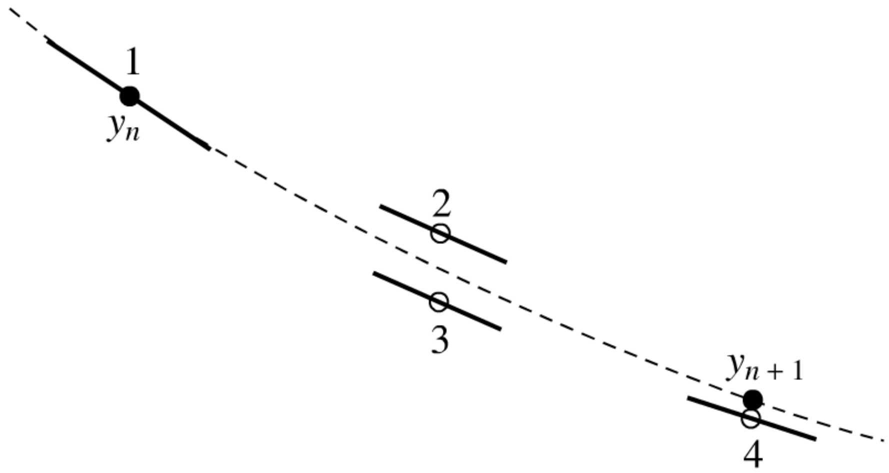
$$\begin{aligned} k_1 &= hf(x_n, y_n) \\ k_2 &= hf\left(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1\right) \\ y_{n+1} &= y_n + k_2 + O(h^3) \end{aligned}$$



4^{th} Order Runge-Kutta

- Historically often the method of choice

$$
\begin{aligned}
k_1 &= hf(x_n, y_n) \\
k_2 &= hf(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_1) \\
k_3 &= hf(x_n + \tfrac{1}{2}h, y_n + \tfrac{1}{2}k_2) \\
k_4 &= hf(x_n + h, y_n + k_3) \\
y_{n+1} &= y_n + \tfrac{1}{6}k_1 + \tfrac{1}{3}k_2 + \tfrac{1}{3}k_3 + \tfrac{1}{6}k_4 + O(h^5)
\end{aligned}
$$



4^{th} Order Runge-Kutta

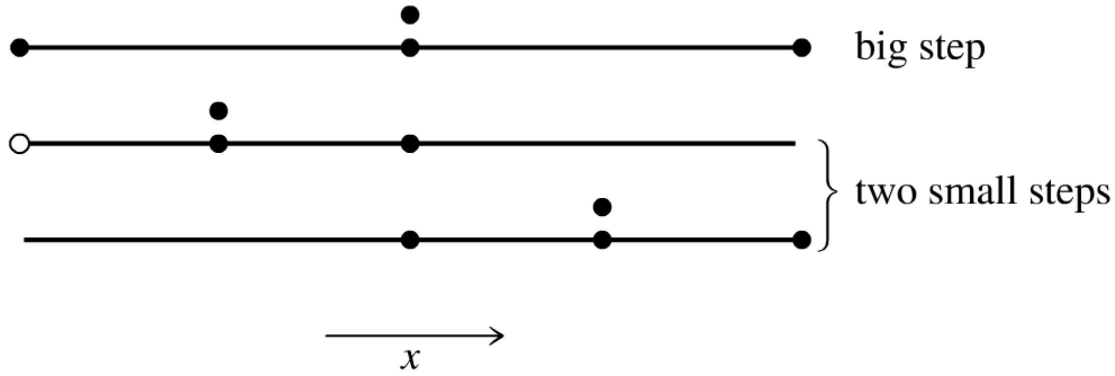
- Add adaptive stepsize control, doubling the step.
- Richardson extrapolation adds one more order

$$y(x+2h) = y_1 + (2h)^5\phi + O(h^6) + \ldots$$

$$y(x+2h) = y_2 + 2(h^5)\phi + O(h^6) + \ldots$$

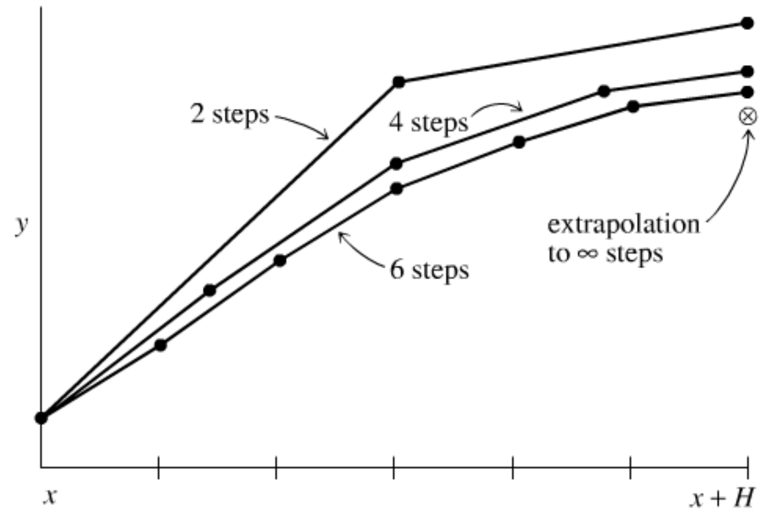
$$y(x+2h) = y_2 + \frac{\Delta}{15} + O(h^6)$$

$$\Delta \equiv y_2 - y_1$$



Bulirsch-Stoer method: “infinite” order extrapolation

- Uses Richardson’s extrapolation again (we also used it for Romberg integration): we estimate the error as a function of interval size h, then we try to extrapolate it to h=0
- As in Romberg we need to have the error to be in terms of h^2 instead of h
- Can use polynomial or rational function extrapolation: we discussed both for interpolations

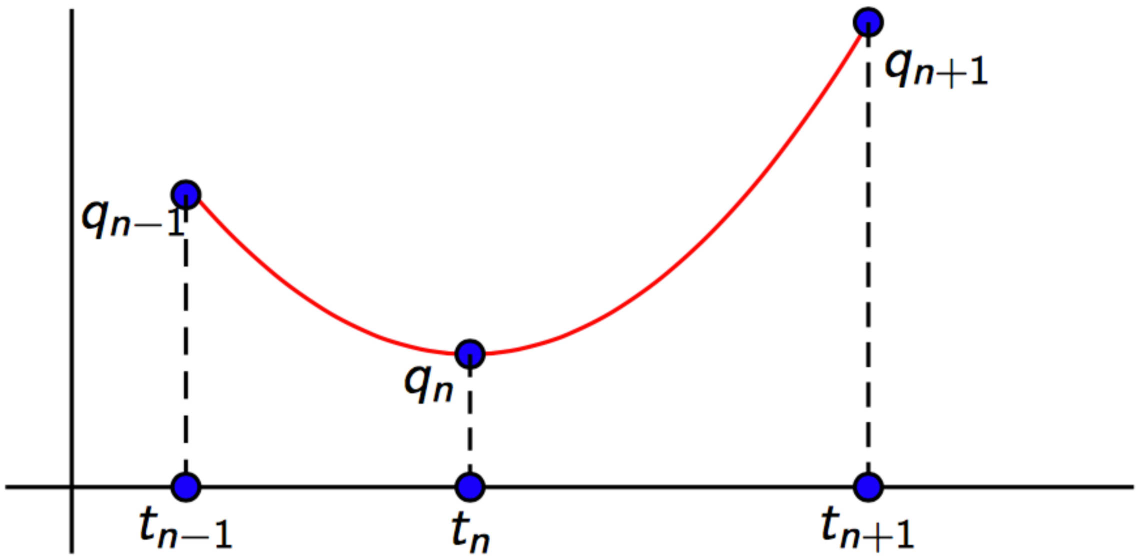


2^{nd} Order Conservative Equations

$$\ddot{q} = f(q)$$

- Stormer-Verlet with two step formulation: we are interpolating parabola through 3 points
- Gains a factor of 2

$$q_{n+1} - 2q_n + q_{n-1} = h^2 f(q_n)$$



One Step Formulation: Leap-frog

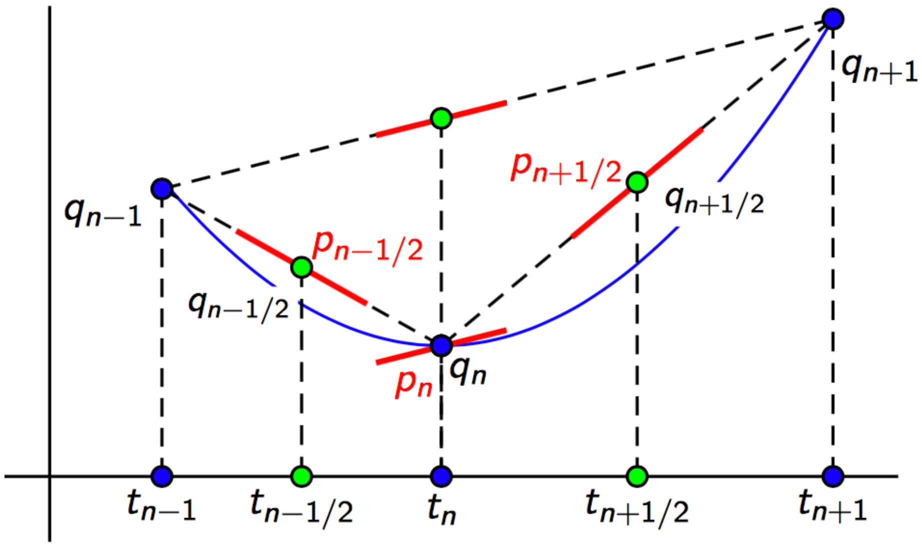
- We introduce momentum $p = \dot{q}, \quad \ddot{q} = f(q)$

$$\dot{q} = p, \qquad \dot{p} = f(q)$$

$$p_{n+1/2} = p_n + \frac{h}{2} f(q_n)$$

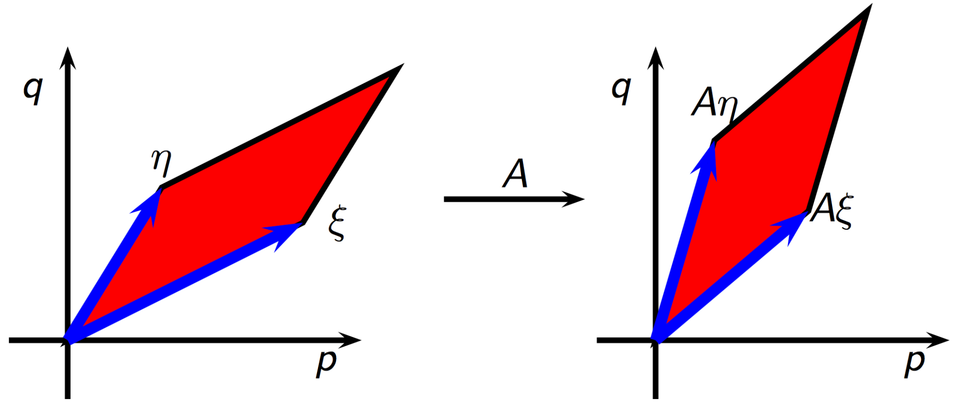
$$q_{n+1} = q_n + h p_{n+1/2}$$

$$p_{n+1} = p_{n+1/2} + \frac{h}{2} f(q_{n+1})$$



Generalization: Symplectic Integrators

- Symplectic integrators preserve phase space (p,q) volume: p,q must be canonical variables
- Symplectic transformation preserves phase space area (p,q) (Lioville's theorem)



- Hamiltonian is not conserved, but a related quantity is and one does not accumulate amplitude error, only phase error
- Useful if one needs to integrate a system for a long time (e.g. planet orbits etc)

Leapfrog is Symplectic

Hamiltonian problem $\dot{p} = -H_q(p, q)$, $\dot{q} = H_p(p, q)$

Theorem. The Störmer-Verlet method

$$p_{n+1/2} = p_n - \frac{h}{2} H_q(p_{n+1/2}, q_n)$$

$$q_{n+1} = q_n + \frac{h}{2}\Big(H_p(p_{n+1/2}, q_n) + H_p(p_{n+1/2}, q_{n+1})\Big)$$

$$p_{n+1} = p_{n+1/2} - \frac{h}{2} H_q(p_{n+1/2}, q_{n+1})$$

is symplectic.

Euler can be made symplectic

applied to $\dot{p} = -H_q$, $\dot{q} = H_p$:

$$\begin{aligned} p_{n+1} &= p_n - hH_q(p_{n+1}, q_n) \\ q_{n+1} &= q_n + hH_p(p_{n+1}, q_n) \end{aligned} \qquad \text{(SE1)}$$

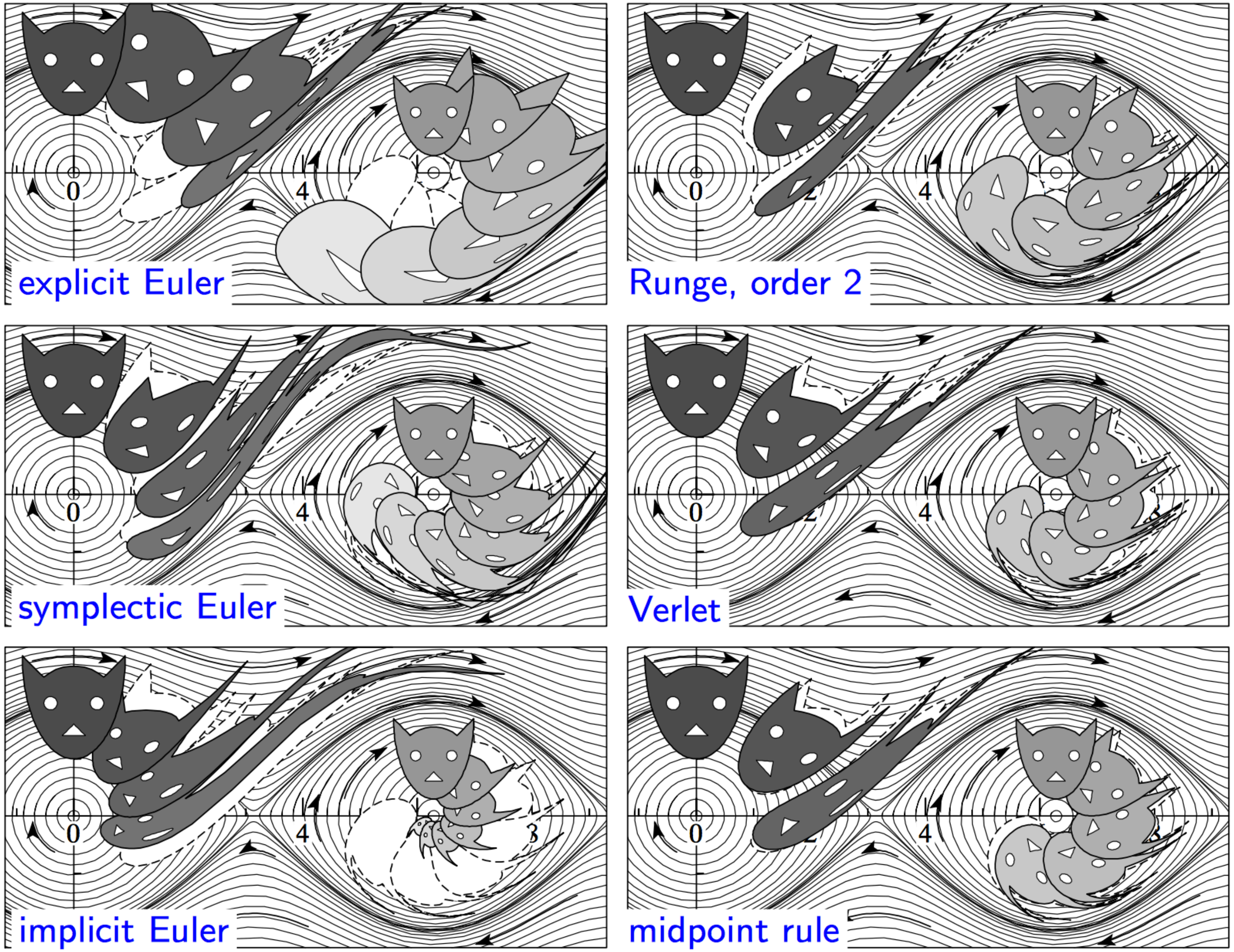
or

$$\begin{aligned} q_{n+1} &= q_n + hH_p(p_n, q_{n+1}) \\ p_{n+1} &= p_n - hH_q(p_n, q_{n+1}) \end{aligned} \qquad \text{(SE2)}$$

Theorem. (de Vogelaere, 1956)
The symplectic Euler method is symplectic.

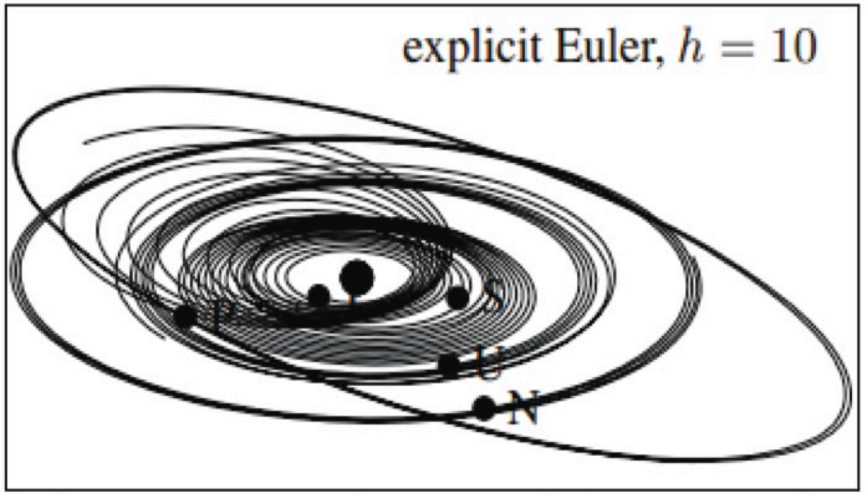
Theorem. The implicit midpoint rule is symplectic.

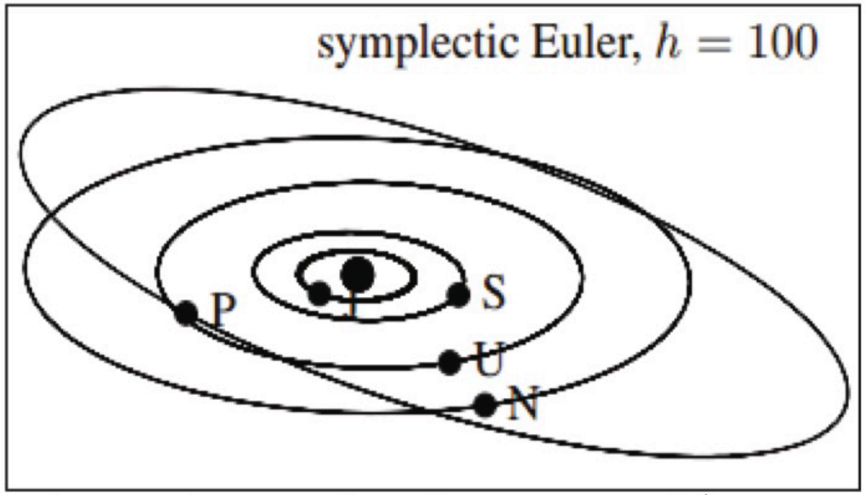
Phase Space Flow



Example: Planetary Orbit Integration

- Explicit Euler's orbits decay. This is not cured by higher order (Runge-Kutta, B-S…)
- Symplectic integrators preserve the orbit amplitude (but not the phases, not shown)





Hamiltonian Monte Carlo

- Remember the HMC discussion: we take a few steps integrating Hamiltonian along the path, then resample the momentum. Acceptance rate is 1 if Hamiltonian is conserved, otherwise it drops. So we'd like to move as far as possible to reduce the correlation of samples, while preserving the Hamiltonian.
- In hierarchical Bayesian models we work with many latent variables that we marginalize over: it is important that HMC solver conserves H in very high dimensions
- Symplectic integrators are ideal for this purpose. Typically we use leap-frog (2^{nd} order, symplectic)
- Codes like Stan are currently best on the market
- However, we need gradient H_q: we will discuss automatic differentiation in the next lecture

Stiff Equations

- Explicit (forward) Euler:

$$y' = -cy$$

$$y_{n+1} = y_n + hy'_n = (1 - ch)y_n$$

- Unstable if $h > 2/c$, since y goes to infinity
- Example:

$$u' = 998u + 1998v \qquad u(0) = 1 \qquad v(0) = 0$$

$$v' = -999u - 1999v$$

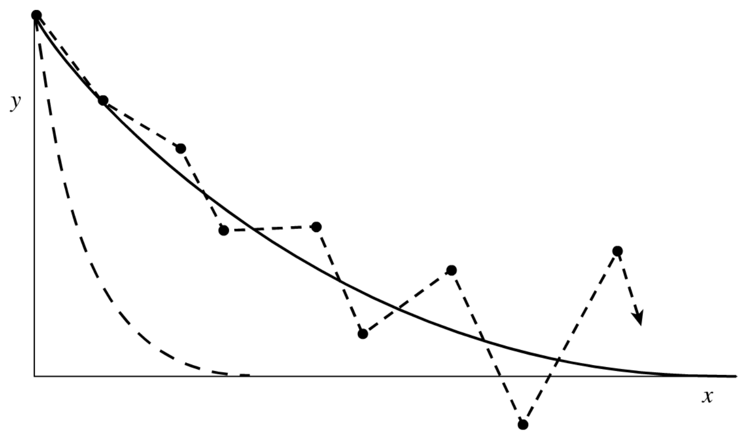
$$u = 2e^{-x} - e^{-1000x}$$

$$u = 2y - z \qquad v = -y + z \qquad v = -e^{-x} + e^{-1000x}$$

- But the system is unstable if h > 1/1000
- Solution: implicit (backward Euler)

$$y_{n+1} = y_n + hy'_{n+1}$$

$$y_{n+1} = \frac{y_n}{1 + ch}$$



General Appraoch

- If we are solving a linear system: $\mathbf{y}' = -\mathbf{C} \cdot \mathbf{y}$

  $\mathbf{T}^{-1} \cdot \mathbf{C} \cdot \mathbf{T} = \mathrm{diag}(\lambda_0 \ldots \lambda_{N-1})$ $\qquad$ $\mathbf{z}' = -\mathrm{diag}(\lambda_0 \ldots \lambda_{N-1}) \cdot \mathbf{z}$

  $\mathbf{z} = \mathrm{diag}(e^{-\lambda_0 x} \ldots e^{-\lambda_{N-1} x}) \cdot \mathbf{z}_0$

- Exact solution: $\mathbf{y} = \mathbf{T} \cdot \mathrm{diag}(e^{-\lambda_0 x} \ldots e^{-\lambda_{N-1} x}) \cdot \mathbf{T}^{-1} \cdot \mathbf{y}_0$

- Explicit scheme: $\mathbf{y}_0 = \sum_{i=0}^{N-1} \alpha_i \boldsymbol{\xi}_i \qquad \mathbf{y}_n = \sum_{i=0}^{N-1} \alpha_i (1 - h\lambda_i)^n \boldsymbol{\xi}_i$

- Stability condition: $|1 - h\lambda_i| < 1 \qquad i = 0, \ldots, N-1 \qquad h < \frac{2}{\lambda_{\max}}$

- Implicit scheme: $\mathbf{y}_{n+1} = (\mathbf{1} + \mathbf{C}h)^{-1} \cdot \mathbf{y}_n$

- Always stable: $|1 + h\lambda_i|^{-1} < 1 \qquad i = 0, \ldots, N-1$

Stiff Nonlinear Equations

- In general, implicit scheme hard to solve

$$\mathbf{y}' = \mathbf{f}(\mathbf{y})$$

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\mathbf{f}(\mathbf{y}_{n+1})$$

- Linearize f :
- (Newton's method)

$$\mathbf{y}_{n+1} = \mathbf{y}_n + h\left[\mathbf{f}(\mathbf{y}_n) + \left.\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\right|_{\mathbf{y}_n} \cdot (\mathbf{y}_{n+1} - \mathbf{y}_n)\right]$$

- Invert Jacobian: $\mathbf{y}_{n+1} = \mathbf{y}_n + h\left[\mathbf{1} - h\frac{\partial \mathbf{f}}{\partial \mathbf{y}}\right]^{-1} \cdot \mathbf{f}(\mathbf{y}_n)$

- This is semi-implicit Euler method
- There are also stiff versions of higher order ODE

Partial Differential Equations

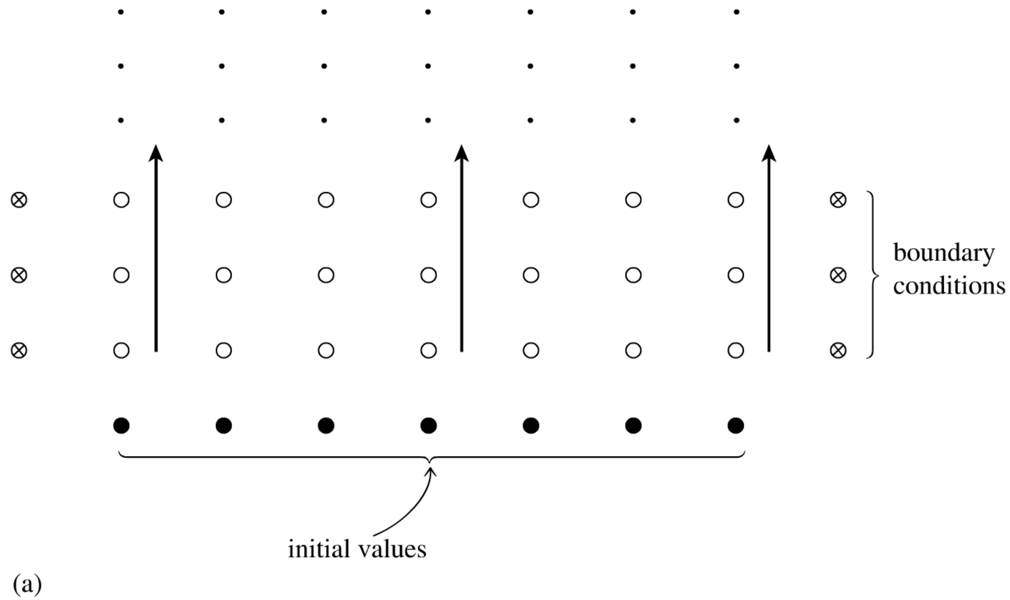
- This is a vast subject, and we will only mention its existence
- Hyperbolic, e.g. wave equation: $$\frac{\partial^2 u}{\partial t^2} = v^2 \frac{\partial^2 u}{\partial x^2}$$
- Parabolic, e.g. diffusion equation: $$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x}\left(D \frac{\partial u}{\partial x}\right)$$
- Both of these are initial value (Cauchy) problems
- Boundary value problem: elliptic, Elliptic, e.g. Poisson equation

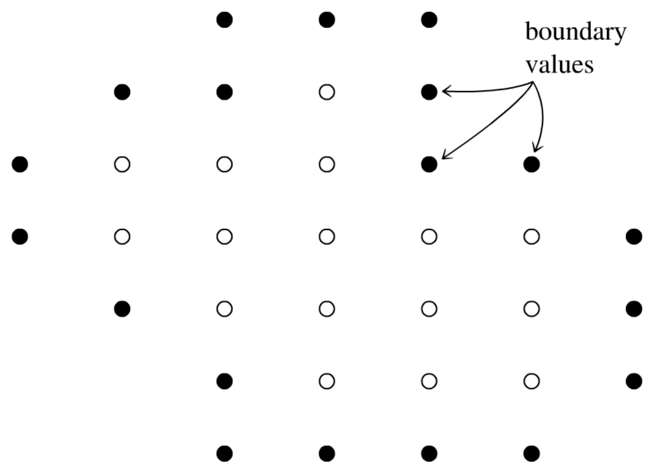
$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = \rho(x, y)$$

- If source ρ=0 this is Laplace equation

Finite Difference Method

- Discretize on a grid…





Summary

- ODEs and PDEs are central to numerical analysis in physical sciences, engineering…
- ODEs have a relatively stable methods
- PDEs have a vast array of approaches: relaxation, finite differences, finite elements, spectral methods, matrix methods, multi-grid, Monte Carlo, variational…

Literature

- *Numerical Recipes*, Press et al., Chapter 17-20