12-04-17, Mon

Multistep methods:

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

$$\int_{t_n}^{t_{n+1}} y'(t)\,dt = \int_{t_n}^{t_{n+1}} f(t, y(t))\,dt.$$

Recall from last class that we defined the trapezoid multistep method as

$$y_{n+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right].$$

Scheme is implicit (y_{n+1} appears on both sides).

Q: How do we handle this?

A: You could as an equation

$$G(x) = 0$$

and find the root, e.g. use Newton's method.

Question: When will this type of iteration converge converge to the true solution of the DE?

Answer: Let $y_{n+1}^{(0)}$ be a good initial guess of the solution y at point x_{n+1}. Then y_{n+1}^{j+1} is the solution estimate at iteration $j+1$, say of Newton.

approx ① $y_{n+1}^{j+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{j})\right], \; j = 0, 1, \ldots$

true ② $y_{n+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y_{n+1})\right]$

Subtract ② − ① to get error

$$y_{n+1} - y_{n+1}^{j+1} = \frac{h}{2}\left[f(x_{n+1}, y_{n+1}) - f(x_{n+1}, y_{n+1}^{j})\right].$$

If we assume the RHS of the DE (namely f) satisfies a Lipschitz condition, i.e.

$$|f(x, y_1) - f(x, y_2)| \le k|y_1 - y_2|, \text{ for some } k \ge 0.$$

Then

$$|y_{n+1} - y_{n+1}^{j+1}| \le \frac{hk}{2}[y_{n+1} - y_{n+1}^{j}].$$

So if

$$\frac{hk}{2} < 1$$

then the iterates $y_{n+1}^{j} \to y_{n+1}$ as $j \to \infty$.

Constant k can be found using the mean value theorem.

$$y_{n+1} - y_{n+1}^{j} \le \frac{h}{2} f_y(x_{n+1}, y_{n+1})(y_{n+1} - y_{n+1}^{j}).$$

Note that the trapezoid rule gives a truncation error of $\mathcal{O}(h^3)$. To maintain this order of accuracy, we require the approximations going into the solution scheme to be $\mathcal{O}(h^3)$ locally ($\mathcal{O}(h^2)$ locally). Otherwise, we lose accuracy of trapezoid method.

We need a guess for y_{n+1} (RHS). If we use Euler's method

$$y_{n+1}^{(0)} = y_n + hf(x_n, y_n)$$

has error $\mathcal{O}(h^2)$. So

$$y_{n+1} - y_{n+1}^{(0)} = \mathcal{O}(h^2).$$

If we want to ensure approx on RHS of trapezoid rule, multistep method is at least $\mathcal{O}(h^3)$.

Then from Lipschitz condition,

$$|y_{n+1} - y_{n+1}^{j+1}| \le \frac{hk}{2} |y_{n+1} - y_{n+1}^{j}|$$

would require 2 iterations of Euler.

Note:

$$|y_{n+1} - y_{n+1}^{2}| \le \frac{hk}{2} |y_{n+1} - y_{n+1}^{1}| \le \frac{hk}{2}\left[\frac{hk}{2} |y_{n+1} - y_{n+1}^{(0)}|\right]$$

$$= \frac{h^2k^2}{4} |y_{n+1} - y_{n+1}^{0}|.$$

Midpoint method is more accurate. Error is $\Theta(h^3)$. One iteration will suffice to give RHS approx of y_{n+1}.

Predictor - corrector methods:

The formulas that gives the guesses at y_{n+1} (like Euler $y_{n+1}^{(0)} = y_n + hf(x_n, y_n)$)

or midpoint

$$y_{n+1}^{(0)} = y_{n-1} + 2hf(x_n, y_n)$$

are called predictor formulas. When the trapezoid rule is then applied to complete the estimation on y_{n+1}:

$$y_{n+1}^{j+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{j})], \quad j = 0, 1, \ldots$$

These are called the correction formulas.

Final exam:

- Bring a simple calculator (trig or log formulas)

* ① Study notes

② Work problems (from book)

③ Read books

Review:

- Section 6.1 - Chebyshev polynomials
  - unevenly spaced nodes optimize placement and reduce oscillations that you see with Lagrange, Newton
  - Formulas, properties, error $|f-p| \le \frac{1}{2^n(n+1)!} \max |f^{(n+1)}(\xi)|$.

Section 6.4: Spline interpolation
- properties of splines
- how to find them

Section 6.8: Best approximation, least-squares theory

Want to minimize

$$\max_{a \le x \le b} |f(x) - p(x)|$$

over whole interval.

* Theorem characterizing best approximation

g is the best approximation to f in G iff

$f - g \perp G$.

- proof

* How this idea is used in least squares theories

- Idea of orthogonal polynomials
  - Theory of recurrence formulas for these polynomials

- Theorem on coeffecients for best approximation

$$f \approx \sum_i c_i g_i = \langle f, g_i \rangle$$ (Gram-Schmidt orthogonalization)

Section 6.12: Trigonometric interpolation

- Fourier series (definition and coeffecients in expansion)
- Note that a_k, b_k, etc. are the best possible choice (in L^2 sense)
- Properties of Fourier transform (Parseval's formula)

Section 6.13: Fast Fourier Transform

- Definition of discrete Fourier Transform (find coeffecients in FT interpolation of function at evenly spaced nodes around unit circle).

$$Fc = f$$

- Finding c, cost of finding F^{-1} is $\Theta(n^3)$ where n = # rows or columns of F.
- Fourier matrix form

$$F^{-1} = \frac{1}{n} F$$

- Then finding coeffecients in $Fc = f$ reduces to a matrix-vector product

$$c = \frac{1}{n} F f.$$

$$\boxed{Fc = f}$$

data

Fourier matrix

coeffecients we want to find

- This is cost $\Theta(n^2)$

- FFT is $O(n\log n)$.

Key observation is if $n=2m$ then

$\omega_n^2 = \omega_m$ where $\omega_n = e^{2\pi i/n}$ (n^{th} root of unity)

squaring ω_n, angle is doubled.

- So divide vector f_n into even and odd components and do 2 matrix-vector multiplications of length $n/2$ each.
- Recursively break down vector using products of 2.

- Proof of cost of FFT.
- Application of FFT to solutions of PDEs.

Chapter 7: Numerical differentiation and integration

Section 7.1: Numerical differentiation

- Taylor series approximations to derivatives
- error in these formulas come from remaining higher order terms in TS
- issue of roundoff error corrupting solution as $h \to 0$
- Tradeoff between roundoff error and truncation error
- Method of undetermined coeffecients for finding formulas of errors

Richardson extrapolation:

- more accurate formulas found by combining formulas h, $h/2$ step sizes.
- Optimize multiple of one formula to make errors as small as possible

$$4L = 4\,\mathcal{O}(h/2) + a_2 h^2 + a_4 h^4/4 + \dots$$
$$-\quad L = \mathcal{O}(h) + a_2 h^2 + a_4 h^4 + \dots$$

Section 7.2: Numerical integration based on interpolation

- Trapezoid rule - use linear Lagrange interpolating polynomials
- error
- composite rule
- general Newton-Cote's formula
- Simpson's rule
- Simple formulas, may not do best job

Section 7.3: Gaussian quadrature

Idea: Choose both nodes and weights in an optimal way to reduce error

Goal: We want to integrate exactly as high a degree polynomial as possible with a fixed number of function evaluations

Nodes and weights can be found from solving sets of nonlinear equations

eqns

$$\int_{-1}^{1} 1\,dx = w_1 f_1(x) + \dots$$

$$\int_{-1}^{1} x\,dx = w_1 f_1(x) + \dots$$

$$\vdots$$

and so on. Can be hard.

- In fact the weights

  A_i = integral of weighted Lagrange polynomials, and nodes are zeroes of Legendre polynomials
  (if $\langle f, g \rangle = \int_{-1}^{1} fg\,dx$)
  otherwise use correct weight inner products and orthogonal polynomials.

- GQ does converge as $n \to \infty$.

$$\int_a^b f(x)\,w(x)\,dx = \lim_{n\to\infty} \sum_{i=0}^{n} A_i f_i(x).$$

Section 7.5: Adaptive quadrature

Place nodes to reflect varying local behavior of the integrand.

- adaptive Simpson and way we estimate error over subintervals.

Chapter 8: Numerical methods for solving ODE's

Section 8.2: Taylor series methods

- Euler (derivations, local truncation error, global error)
  - linear TS
- higher order TS methods (requires upfront calculations of analytic derivatives)

Section 8.3: Runge-Kutta methods

- No such upfront calculation
- Higher accuracy is achieved by doing more function evaluations
- Idea behind derivation (relies on multidimensional TS and average calculations)

Section: 8.4 - Multistep methods

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

Use different quadrature schemes for RHS.

- Midpoint method (explicit) - Stability (Diff. eqn. theory)
  difference
- prediction - correction idea for implicit methods.