Difference equations: (Chapter 1)

We are interested in linear operators $T: V \to V$.

Example: Shift operator E

$$Ex = (x_2, x_3, \dots), \quad x = (x_1, x_2, \dots).$$

L is a polynomial in E. $L = P(E)$.

Theorem: If P is a polynomial and λ is a root of P then one solution of the difference equation

$$P(E)x = 0$$

is $(\lambda, \lambda^2, \dots)$. If all the roots of P are simple and non-zero then all the solutions of the difference equation are linear combinations of these solutions.

Example:

$$(E^2 - 3E^1 + 2E^0)x = 0$$

$$\Updownarrow$$

$$P(\lambda) = \lambda^2 - 3\lambda + 2\lambda = 0.$$

Theorem 3: (on stable difference equations)

For a polynomial P satisfying $p(0) \neq 0$, these properties are equivalent.

1. The difference equation
$$P(E)x = 0$$
is stable.

2. All roots of P satisfy $|z| \leq 1$ and all multiple roots satisfy $|z| < 1$.

Example:

$$\begin{cases} x_{n+1} = \frac{13}{3} x_n - \frac{4}{3} x_{n-1}, & n \geq 1 \\ x_0 = 1 \\ x_1 = 1/3 \end{cases}$$

Postulate: soln $= \left(\frac{1}{3}\right)^n$.

Proof: Use induction on n.

Because $x_0 = 1 = \left(\frac{1}{3}\right)^0$ and $x_1 = \frac{1}{3} = \left(\frac{1}{3}\right)^1$.

Induction step:

$$x_{m+1} = \frac{13}{3} x_m - \frac{4}{3} x_{m-1}$$

$$x_{m+1} = \frac{13}{3}\left(\frac{1}{3}\right)^m - \frac{4}{3}\left(\frac{1}{3}\right)^{m-1} =$$

$$= \left(\frac{1}{3}\right)^{m-1}\left[\frac{13}{9} - \frac{4}{3}\right] = \left(\frac{1}{3}\right)^{m-1}\left[\frac{1}{9}\right] =$$

$$= \left(\frac{1}{3}\right)^{m+1}. \quad \square$$

On a 32-bit computer, we get

$x_0 = 1.000000$

$x_1 = 0.333333$ 7 correct digits

$x_2 = 0.111112$ 6 correct digits

$x_3 = 0.0370373$ 5 correct digits

$x_4 = 0.0123466$ 4 correct digits (true: 0.012345679)

$\vdots$

$x_7 = 0.0005131$ 1 correct digit (true $4.57 \cdot 10^{-4}$)

$x_8 = 0.0003757$ 0 correct digit (true $1.52415 \cdot 10^{-4}$)

Algorithm is unstable.

Any error present in x_n is multiplied by $\frac{13}{3}$ to get x_{n+1}. So an error in x_1 propagates into x_{15} with a factor of $(\frac{13}{3})^{14}$.

Error of $x_1 \approx 10^{-8}$. So

$$(\tfrac{13}{3})^{14} \approx 823207590 \approx 1 \cdot 10^{9}.$$

This equation

$$x_{n+1} = \frac{13}{3} x_n - \frac{4}{3} x_{n-1}$$

is a difference equation. So

$$x_{n+1} - \frac{13}{3} x_n + \frac{4}{3} x_{n-1} = 0.$$

$$u_{k+1} = A u_k$$

$$\begin{bmatrix} x_{k+1} \\ x_k \end{bmatrix} = \begin{bmatrix} \frac{13}{3} & -\frac{4}{3} \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ x_{k-1} \end{bmatrix}$$

$$(\tfrac{13}{3} - \lambda)(-\lambda) + \frac{4}{3} = 0$$

$$\lambda^2 - \frac{13}{3}\lambda + \frac{4}{3} = 0.$$

$$(\lambda - \tfrac{1}{3})(\lambda - 4) = 0.$$

So $\lambda = \frac{1}{3}, 4$. So solution is a linear combination of powers of these 2 solutions. So general solution is

$$x_n = A(\tfrac{1}{3})^n + B(4)^n$$

where A, B depend on initial conditions x_0, x_1.

True solution is

$$x_n = (\tfrac{1}{3})^n \Rightarrow A = 1, B = 0.$$

But the difference equation solution gotten from this recurrence relation is contaminated by the term

$B(4)^n$.

Eventually, in fact, the $(4)^n$ term dominates the true solution $(\frac{1}{3})^n$.

Consider now

$x_0 = 1$, and $x_1 = 4$.

Then the true solution is $x_n = 4^n$.

Proof:

Base case: $x_0 = 1 = 4^0$, and $x_1 = 4 = 4^1$.

Induction step:

$$x_{m+1} = \frac{13}{3}(4)^m - \frac{4}{3}(4)^{m-1} =$$

$$= 4^m\left[\frac{13}{3} - \frac{1}{3}\right] = 4^m\left[\frac{12}{3}\right] = 4^{m+1}. \quad \square$$

And the numerical results will be stable.

$x_1 = 4.0000$

$x_{10} = 1.048576 \cdot 10^6$ (true 1048576)

Here there is still an error coming from $(\frac{1}{3})^n$ but it is relatively small compared to true solution.

Stability = "continuos dependence on initial data".

Back to multistep methods:

Midpoint method

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

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

$$y(t_{n+1}) - y(t_{n-1}) = (t_{n+1} - t_{n-1}) f(t_n, y_n)$$

Midpoint method

$$y(t_{n+1}) = y(t_{n-1}) + 2h f(t_n, y_n).$$

This is a multi step method because it uses more than one prior value of solution y. Euler only used previous value. Using prior info leads to more accurate numerical solution.

This is an explicit method because we know all the values of y that we need to get new values $y(t_{n+1})$. The order of convergence globally is $\Theta(h^2)$. Local truncation error is $\Theta(h^3)$.

Lets look at model problem:

$$\begin{cases} y'(x) = \lambda x \\ y(0) = 1 \end{cases}$$

By separation of variables,

$$\frac{dy}{y} = \lambda dx$$

$$\ln y = \lambda x + C$$

$$y = Ce^{\lambda x}.$$

$$y(0) = 1$$

$$Ce^{\lambda(0)} = 1$$

$$C = 1.$$

$$\boxed{y(x) = e^{\lambda x}.}$$

Now apply midpoint method for this problem. We get

$$y(x_{n+1}) = y(x_{n-1}) + 2hf(t_n, x_n)$$

$$y_{n+1} = y_{n-1} + 2h\lambda y_n.$$

This is a linear difference equation.

$$u_{k+1} = A u_k$$

$$\begin{bmatrix} y_{n+1} \\ y_n \end{bmatrix} = \begin{bmatrix} 2h\lambda & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} y_n \\ y_{n-1} \end{bmatrix}$$

$$\det \begin{bmatrix} 2h\lambda - \nu & 1 \\ 1 & -\nu \end{bmatrix} = 0$$

$$(2h\lambda - \nu)(-\nu) - 1 = 0.$$

$$\nu^2 - 2h\lambda\nu - 1 = 0$$

roots: $\nu = \dfrac{2h\lambda \pm \sqrt{(2h\lambda)^2 + 4}}{2} = h\lambda \pm \sqrt{(h\lambda)^2 + 1}.$

General solution to difference equation is

$$\textcircled{1} \quad y_n = A\underbrace{\left(h\lambda + \sqrt{(h\lambda)^2+1}\right)}_{v_0}^n + B\underbrace{\left(h\lambda - \sqrt{(h\lambda)^2+1}\right)}_{v_1}^n, \qquad n \geq 0.$$

Write instead

$$y_n = A v_0^n + B v_1^n.$$

Case 1: $0 < \lambda < \infty$ (for all h).

$v_0 > |v_1| > 0$

$v_0 > 1$.

So v_0^n dominates the solution.

Case 2: $-\infty < \lambda < 0$

$v_1 < -1$, $h > 0$

$0 < v_0 < 1$.

So in this case, v_1^n will eventually dominate the solution.

In fact the numerical method (midpoint method) is unstable for these values of λ. One of the eigenvalues has magnitude > 1.

Example: Consider the problem

$$y'(x) = \lambda y(x).$$

Let $\lambda = -1$. So,

$$\begin{cases} y'(x) = -y(x) \\ y(0) = 1 \end{cases}.$$

x_n	y_n = midpoint method solution	error
0.25	0.7500	
0.50	0.6250	0.0288
0.75	0.4375	-0.0185
1.00	0.4063	0.0349
1.25	0.2344	-0.0384
1.50	0.2891	0.0521
⋮		-0.0659
2.25	-0.0322	0.1376

Note that at $x_n = 2.25$, we get a negative value for solution to DE. But true solution is $y = e^x$ which is never negative.

The trapezoid method is another multistep method.

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

$$y(t_{n+1}) = y(t_n) + \int_{t_n}^{t_{n+1}} f(t, y(t))\,dt$$

Apply trapezoid rule for the integral to get

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

This is a single-step implicit method.
The unknown solution $y_{n+1} = y(t_{n+1})$
appears on both sides of the method.