BLG 202E

Week 13 – 03.05.2016

ODEs

Chapter 16 Differential Equations :

Diffusion :

C : concentration

$$\underbrace{J}_{\text{Flux}} = D\frac{\partial c}{\partial x}$$

$$\frac{\partial J}{\partial x} = \frac{\partial c}{\partial t}$$

$$\frac{\partial c}{\partial t} = D\frac{\partial^2 c}{\partial x^2}$$

PDE

ODE : Ordinary Diff Eqns.

$$\boxed{\frac{dy}{dt} = y' = f(t, y(t))} \longrightarrow y(t_0) = y_0$$

Initial-value ODES

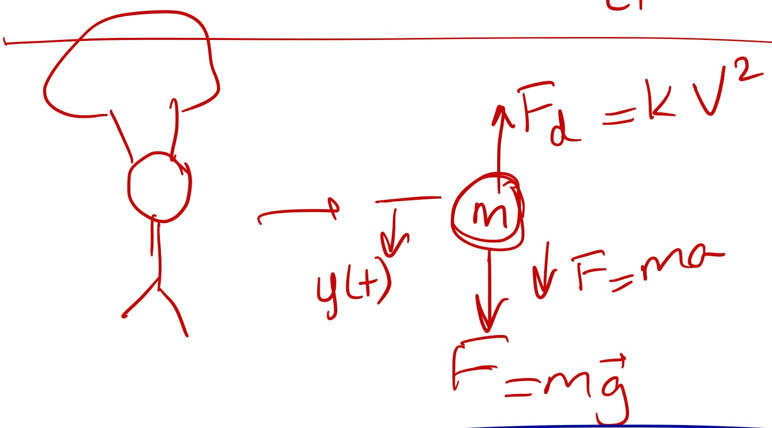
$$\int_{t_i}^{t_{i+1}} \frac{dy}{ds}\,ds = \int_{t_i}^{t_{i+1}} f(s, y(s))\,ds$$

$$y(t_{i+1}) - y(t_i) = \int_{t_i}^{t_{i+1}} f(s, y(s))\,ds$$

$y(t, x_0) = a$
Boundary-value

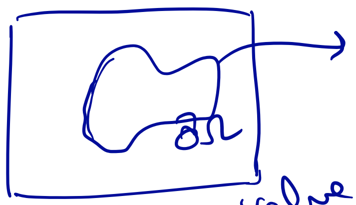
we cannot numerical quadrature.

~~y is implicitly defined~~



$$\cancel{m}\vec{g} + \cancel{m}\,\vec{a}(t) - \frac{k}{m}V^2(t) = 0$$

$$\frac{d^2 y(t)}{dt^2} = -g + \frac{k}{m}\left(\frac{dy(t)}{dt}\right)^2$$



Boundary-value

$V(x,y)$

$V(x,y)\big|_{\partial\Omega} = V_0(x,y)$

$$\frac{\partial V}{\partial t} = K|\nabla V|$$

K: Curvature

PDE

Initial Value ODE problem : Find $y(t)$ that satisfies

$$y'(t) = \frac{dy(t)}{dt} = f(t, y(t))$$

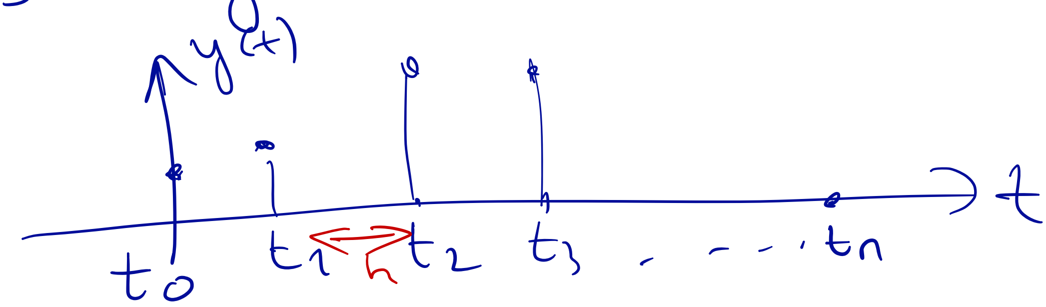
with initial condition

$$y(t_0) = y_0$$

makes the soln. unique.

- $y(t)$ is defined implicitly thru the differential eqn.
- Numerical soln to this problem generates a sequence of values $t_0, t_1, \dots, t_n, \dots$ and a corresponding seq of values $y_0, y_1, \dots, y_n, \dots$ so that

$$y_n \approx y(t_n), \qquad n = 0, 1, 2, \dots$$



$\boxed{h = t_{i+1} - t_i}$: fixed step size.

- ODEs : system of differential equations:

$$\frac{d\underline{y}}{dt} = f(t, \underline{y}(t)) \qquad \underline{y}(t) = \begin{pmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \end{pmatrix}$$

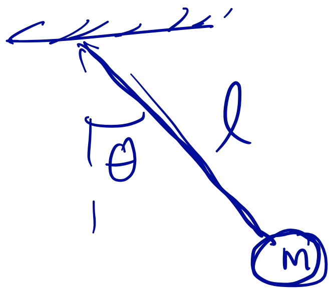
$$\underline{y}(t_a) = \underline{y}_a ,$$

t : independent variable.

$y(t)$: trajectory (solution) of the ODE for Initial value ODE.

Note on theory: The initial value has a unique soln. provided f is "regular enough" (Lipschitz cont.) We'll not cover here.).

Ex: Pendulum : Angular acceleration : (Newton's laws forces)



$$\frac{d^2\theta}{dt^2} = -\frac{g}{\ell}\sin\theta$$

constant of gravity

2nd order ODE :

$$= f(t, \underset{\theta(t)}{y(t)})$$

Write as first-order ODE system :

$$y_1(t) = \theta(t) \Rightarrow y_1'(t) = y_2(t)$$
$$y_2(t) = \theta'(t) \Rightarrow y_2'(t) = -\frac{g}{\ell}\sin\theta = f(t, \theta(t)) = f(\theta(t))$$

ODE system for the pendulum

$$\underline{f}(t, \underline{y}) = \begin{pmatrix} y_2 \\ -\frac{g}{\ell}\sin(y_1) \end{pmatrix}, \quad \underline{y} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}$$

$$\begin{cases} \dfrac{d\underline{y}}{dt} = \underline{f}(t, \underline{y}) \\ y_1(t_0) = \theta(t_0) \\ y_2(t_0) = \theta'(t_0) \end{cases}$$

vector notation

initial values are given.

Ex: Mathematical ecology: populations of prey, and predators.

Prey-predator model: $u(t)$: prey (rabbit) population
$v(t)$: predator (fox) population

$\boxed{\alpha, \beta, \gamma, \delta > 0}$

$$\begin{cases} u' = \alpha u - \beta v u = (\alpha - \beta v) u \\ v' = -\gamma v + \delta u v = (\delta u - \gamma) v \end{cases}$$

The soln. to this nonlinear system of equations

$$\underline{y}' = \begin{pmatrix} u' \\ v' \end{pmatrix} = \underline{f} \; ; \; \underline{y} = \begin{pmatrix} u \\ v \end{pmatrix}, \; \underline{f}(\underline{y}) = \begin{pmatrix} (\alpha - \beta v) u \\ (\delta u - \gamma) v \end{pmatrix}$$

can be obtained numerically. The solns are periodic (See Figure).

Simplest Method to solve numerically an initial-value ODE

Forward Euler Method:

$$\left. \begin{aligned} y' &= f(t, y) \\ y(t_0) &= y_0 \end{aligned} \right]$$

Grid (mesh) points $t_0, t_1, \ldots, t_N$

$h = t_{i+1} - t_i$.

fixed step size.

Find Approximate soln. $y(t_i) \approx y_i \leftarrow$

* Apply forward-difference:

$$\underbrace{y'(t_i)}_{f(t_i, y(t_i))} = \frac{y(t_{i+1}) - y(t_i)}{h} - \frac{h}{2} y''(\xi)$$

$$\Rightarrow \; y(t_{i+1}) = y(t_i) + h f(t_i, y(t_i)) + \frac{h^2}{2} y''(\xi)$$

$\Longrightarrow$

The initial value problem is then solved by:

$$\underline{y}_{i+1} = \underline{y}_i + h\, f(t_i, \underline{y}_i)$$
$$t_{i+1} = t_i + h$$
$$\underline{y}_0 = \underline{y}(t_0)$$

Forward Euler method

Ex: $\boxed{\begin{array}{l} y' = y \\ y(0) = 1 \end{array}}$ $\boxed{y(t) = e^t \text{ exact soln}}$ → Forget about this, go to numerical soln

⇒ Apply forward Euler:

$$y_0 = 1$$
$$y_{i+1} = y_i + h \cdot y_i$$
$$\left.\begin{array}{l} y_{i+1} = (1+h)\, y_i \\ y_0 = 1 \end{array}\right\} \quad i = 0, 1, 2, \ldots$$

Backward Euler Method:

Instead of forward, use the backward difference:

$$y'(t_{i+1}) = \frac{y(t_{i+1}) - y(t_i)}{h} + \frac{h}{2} y''(\xi)$$

Recall T.S.
$y(t+h) = \ldots$
$y(t-h) = \ldots$

Derivation:

$$y(t_{i-1}) = y(t_i) - h\, y'(t_i) + \frac{h^2}{2} y''(\xi)$$

$$y'(t_i) = \frac{y(t_i) - y(t_{i-1})}{h} + \frac{h}{2} y''(\xi)$$

$$y'(t_{i+1}) = \frac{y(t_{i+1}) - y(t_i)}{h} + \frac{h}{2} y''(\xi)$$

So, set : $$\left.\begin{aligned} y_{i+1} &= y_i + h f(t_{i+1}, y_{i+1}) \\ t_{i+1} &= t_i + h \end{aligned}\right\}$$

Note that y_{i+1} (unknown) appears implicitly!

For stepping procedure, solve a (nonlinear) system of eqns

$$y_{i+1} - y_i - h f(t_{i+1}, y_{i+1}) = 0$$

$\therefore$ More costly than forward Euler method.

* $\left.\begin{array}{l}\text{Forward Euler is explicit} \\ \text{Backward Euler is implicit.}\end{array}\right\}$ methods.

---

* Absolute Stability:

During computations, step size choice becomes important.

– Consider a simple test equation:

$$\left.\begin{aligned} y' &= \lambda y \\ y(0) &= y_0 \end{aligned}\right\}$$

soln: $y(t) = y_0 e^{\lambda t}$

$\lambda > 0$: models growth,

$\lambda < 0$: models decay (death)

Apply forward Euler method:

$$y_{i+1} = y_i + h f(t_i, y_i) = y_i + h \lambda y_i$$

$$y_{i+1} = (1 + h\lambda) y_i$$

Require that approx soln does not grow : $|y_{i+1}| \leq |y_i|$

$|y_{i+1}| \le |y_i| \qquad y_{i+1} = (1+\lambda h)\, y_i \qquad \boxed{y' = \lambda y}$

$\Rightarrow |1+\lambda h| \le 1$ for stability.

$-1 \le \underline{1+\lambda h \le 1} \qquad y' = \lambda y$

$y(t) = y_0 e^{\lambda t}$ ✓

$\lambda > 0$: models growth, numerically also grows.

$\boxed{\lambda < 0}$: $\underline{-1 \le 1 + h\lambda} \Rightarrow h\lambda \ge -2$ $\qquad \boxed{h \le \frac{2}{-\lambda}}$

$\boxed{h \le -\frac{2}{\lambda}}$

$\lambda < 0$, Absolute stability requirement $\boxed{h \le \frac{2}{|\lambda|}}$

Bound on the Step size h

$\Rightarrow$ for $\lambda < 0$, we must require h to satisfy this bound for stable solns.

$\boxed{\text{Ex:}}$ (★)

$y' = \underline{-1000\,(y - \cos(t))} - \sin(t)$

$\underline{y' = -1000\, y} + \underline{1000} \ \underline{\cos(t)} - \underline{\sin(t)}$

$y(0) = 1$

$y(t) = \cos(t)$

If we want to apply forward Euler method.

we have to choose $h \le \frac{2}{1000} = \boxed{0.002}$

$\Rightarrow$ otherwise, you end up with a non-correct soln.

$\Rightarrow$ HW!

* Stability: Test eqn : $\boxed{y' = \lambda y}$

Apply the Backward Euler method :

$$y_{i+1} = y_i + h\lambda y_{i+1}$$

$$y_{i+1} = \frac{1}{1-h\lambda} y_i$$ $\Rightarrow$ we require $|y_{i+1}| \leq |y_i|$ for absolute stability

$$\Rightarrow \frac{1}{|1-h\lambda|} \leq 1$$

$$\boxed{-1 \leq \frac{1}{1-h\lambda} \leq 1}$$

$$-(1-h\lambda) \leq 1$$

$$h\lambda - 1 \leq 1$$

$$\boxed{h\lambda \leq +2}$$

$\lambda < 0$

$$\boxed{h \geq \frac{2}{\lambda}}$$

$$1-h\lambda \geq 1$$

$$\boxed{h\lambda \leq 0}$$

$$\lambda < 0$$

$$\boxed{h > 0}$$

$$\boxed{h > 0}, \quad \lambda < 0$$

No bound / (absolute stability criterion)

Ex (*) : $h > 0.002$ with Forward Euler method

$h = 0.01\pi \rightarrow$ the result blows up.

w/ Backward Euler $h = 0.01\pi$ gives error $= 3.2e{-9}$

" $h = 0.1\pi$ " $= 1.7e{-5}$

Forward & Backward Euler are $O(h)$.

Runge_Kutta Methods: higher order methods

The Euler methods we've seen are only 1st order accurate. Want higher accuracy.

* ODE: $y' = f(t,y)$, integrate the ODE from t_i to t_{i+1} (in 1 step): $\int_{t_i}^{t_{i+1}} \frac{dy}{ds}\, ds = \int_{t_i}^{t_{i+1}} f(s,y)\, ds$

$$\boxed{y(t_{i+1}) = y(t_i) + \int_{t_i}^{t_{i+1}} f(t,y(t))\,dt}$$

discretize this integral by a basic quadrature rule.

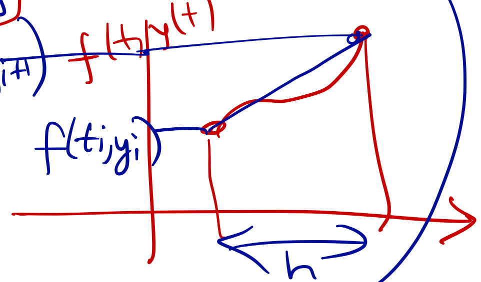
Replace $\int_{t_i}^{t_{i+1}} f(t,y(t))\,dt$

Implicit Trapezoidal rule:

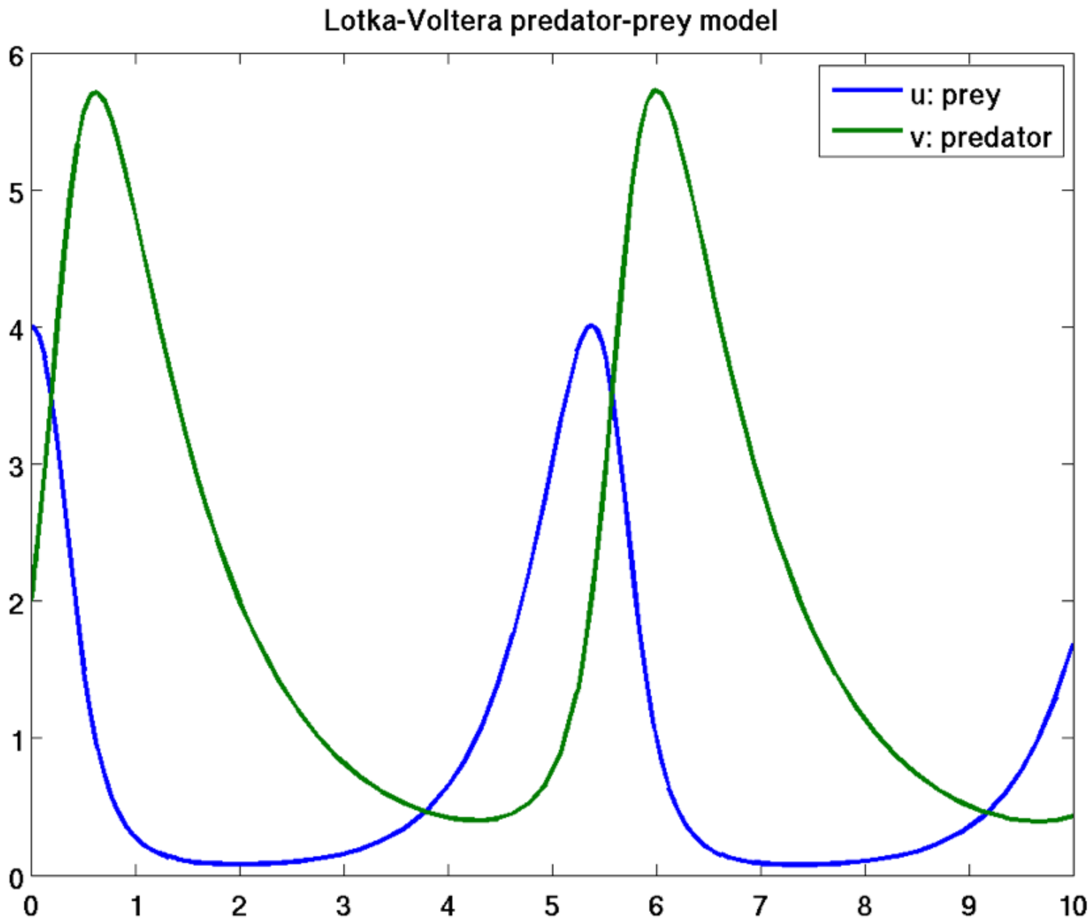
$$\boxed{y_{i+1} = y_i + \frac{h}{2}\left[f(t_i,y_i) + f(t_{i+1},y_{i+1})\right]}$$

Implicit!

$O(h^2)$



$$= \frac{h}{2}\left[f(t_i,y_i) + f(t_{i+1},y_{i+1})\right]$$



Unknown f:

$$E(f(x)) = \int (f - g)^2 dx + \underbrace{\int \left(\frac{\partial f}{\partial x}\right)^2 dx}$$

Cost fn.

$$\arg\min_f E(f)$$

$$\frac{\partial E}{\partial f} = 0 \Rightarrow \boxed{\frac{\partial f}{\partial t} = \cdots}$$

Diff equations are also used in optimizing functionals (cost functions).