Numerical Derivatives

Mahendra Verma

$$f'(x) = \lim_{h \to 0} \frac{f(x+h) - f(x)}{h}$$

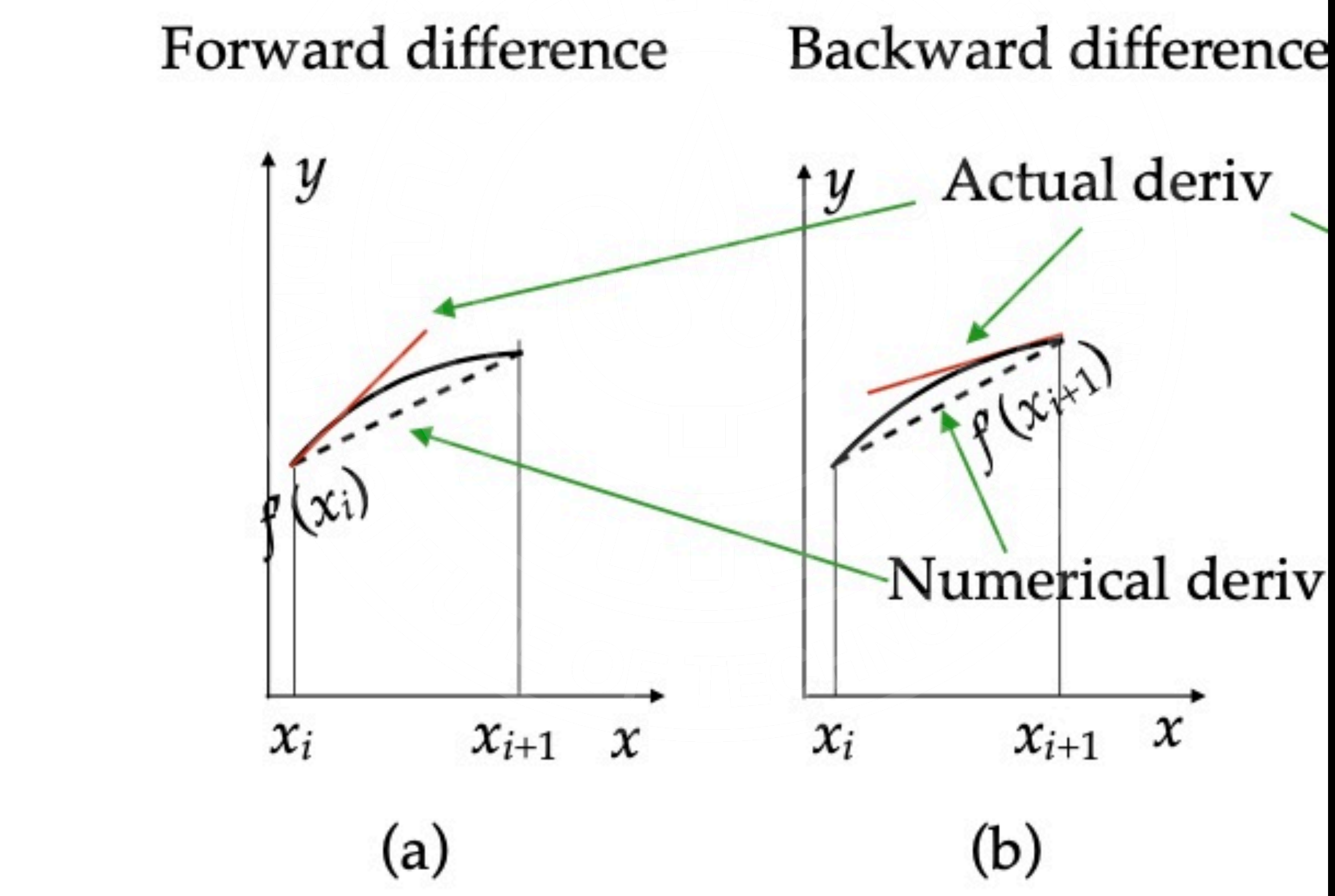
$$f(x) \approx P_2(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} f_i + \frac{x - x_i}{x_{i+1} - x_i} f_{i+1}$$

Forward difference: $f'(x_i) \approx P_2'(x_i) = D_+ f = \dfrac{f_{i+1} - f_i}{h_i}$

Backward difference: $f'(x_{i+1}) \approx P_2'(x_{i+1}) = D_- f = \dfrac{f_{i+1} - f_i}{h_i}$

Forward difference: $f'(x_i) \approx P_2'(x_i) = D_+ f = \dfrac{f_{i+1} - f_i}{h_i}$

Backward difference: $f'(x_{i+1}) \approx P_2'(x_{i+1}) = D_- f = \dfrac{f_{i+1} - f_i}{h_i}$



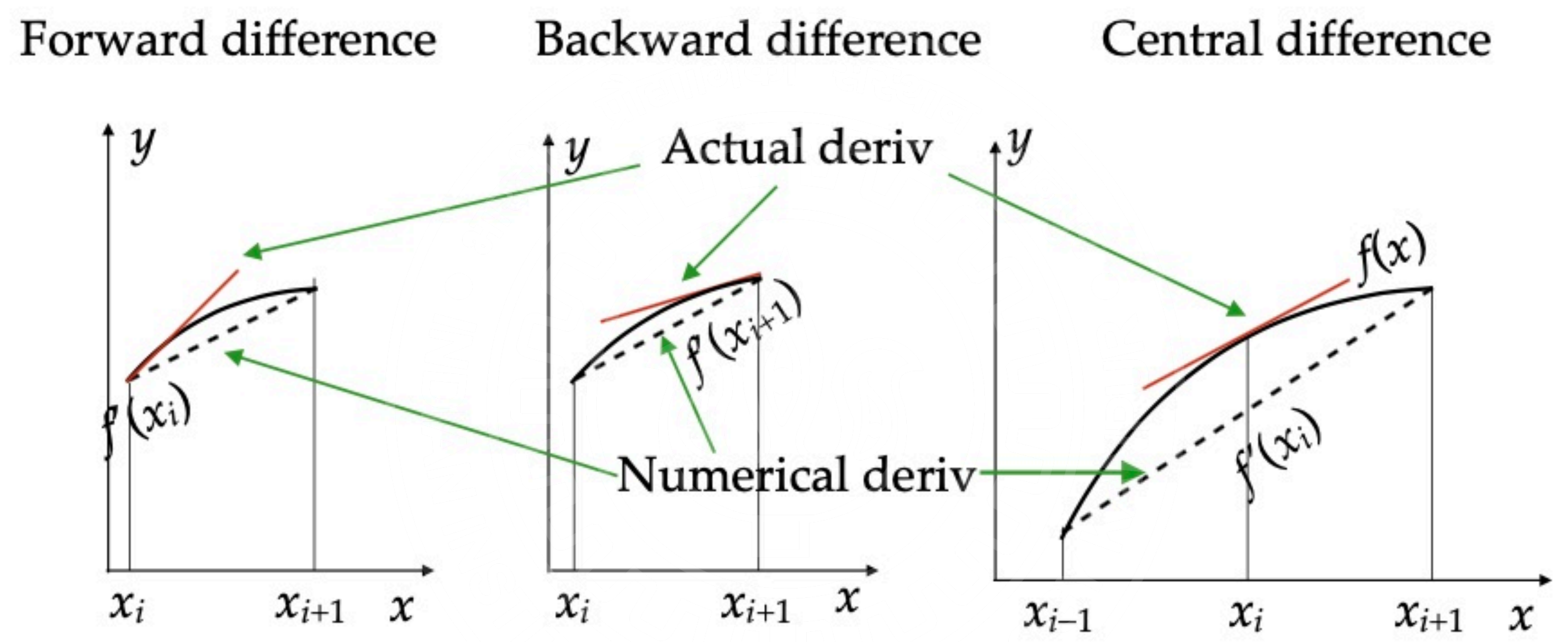
3-Points

$$f(x) \approx P_3(x) = \frac{(x-x_i)(x-x_{i+1})}{(h_{i-1}+h_i)(h_{i-1})} f_{i-1} + \frac{(x-x_{i-1})(x-x_{1+1})}{h_{i-1}(-h_i)} f_i + \frac{(x-x_{i-1})(x-x_i)}{(h_{i-1}+h_i)(h_i)} f_{i+1} \quad \ldots(33)$$

Forward difference: $f'(x_{i-1}) = D_+ f = \dfrac{-3f_{i-1} + 4f_i - f_{i+1}}{2h}$

Backward difference: $f'(x_{i+1}) = D_- f = \dfrac{f_{i-1} - 4f_i + 3f_{i+1}}{2h}$

Central difference: $f'(x_i) = \dfrac{1}{2}(D_+ + D_-) f = \dfrac{f_{i+1} - f_{i-1}}{2h}$



Forward difference
Backward difference
Central difference
Actual deriv
Numerical deriv
y
y
y
$f'(x_i)$
$f'(x_{i+1})$
$f'(x_i)$
$f(x)$
x_i
x_{i+1}
x
x_i
x_{i+1}
x
x_{i-1}
x_i
x_{i+1}
x

Table 23: Coefficients for various formulas for the derivatives. The last column is the error in the derivative.

	f_{i-2}	f_{i-1}	f_i	f_{i+1}	f_{i+2}	Error
Forward						
hf'_i			−1	1		$O(h)$
$2hf'_i$			−3	4	−1	$O(h^2)$
$h^2f''_i$			1	−2	1	$O(h)$
Backward						
hf'_i		−1	1			$O(h)$
$2hf'_i$	1	−4	3			$O(h^2)$
$h^2f''_i$	1	−2	1			$O(h)$
Central						
$2hf'_i$		−1	0	1		$O(h^2)$
$12hf'_i$	1	−8	0	8	1	$O(h^4)$
$h^2f''_i$		1	−2	1		$O(h^2)$
$12h^2f''_i$	−1	16	−30	16	−1	$O(h^4)$

Error Analysis

$$E_n(x) = f(x) - P_n(x) = \frac{f^{(n)}(\zeta)}{n!} \prod_i (x - x_i)$$

$$f'(x_j) - P_n'(x_j) = \frac{d}{dx} E_n(x)\Big|_{x=x_j} = \frac{f^{(n)}(\zeta)}{n!} \frac{d}{dx} \prod_i (x - x_i) = \frac{f^{(n)}(\zeta)}{n!} \prod_{i, i \neq j} (x_j - x_i)$$

$$f'(x_j) - P'_n(x_j) = \frac{d}{dx} E_n(x)\,|_{x=x_j} = \frac{f^{(n)}(\zeta)}{n!} \frac{d}{dx} \prod_i (x - x_i) = \frac{f^{(n)}(\zeta)}{n!} \prod_{i,i\neq j} (x_j - x_i)$$

With *n* points:

Derivative accurate for (*n*-1) order poly

Error : $O(h^{n-1})$

Examples

Table 23: Coefficients for various formulas for the derivatives. The last column is the error in the derivative.

	f_{i-2}	f_{i-1}	f_i	f_{i+1}	f_{i+2}	Error
Forward						
hf'_i			−1	1		$O(h)$
$2hf'_i$			−3	4	−1	$O(h^2)$
$h^2f''_i$			1	−2	1	$O(h)$
Backward						
hf'_i		−1	1			$O(h)$
$2hf'_i$	1	−4	3			$O(h^2)$
$h^2f''_i$	1	−2	1			$O(h)$
Central						
$2hf'_i$		−1	0	1		$O(h^2)$
$12hf'_i$	1	−8	0	8	1	$O(h^4)$
$h^2f''_i$		1	−2	1		$O(h^2)$
$12h^2f''_i$	−1	16	−30	16	−1	$O(h^4)$

$$f(x) = x^2$$

Thank you!

ODE Solvers

Initial Value Problem

Mahendra Verma

ODEs in Physics

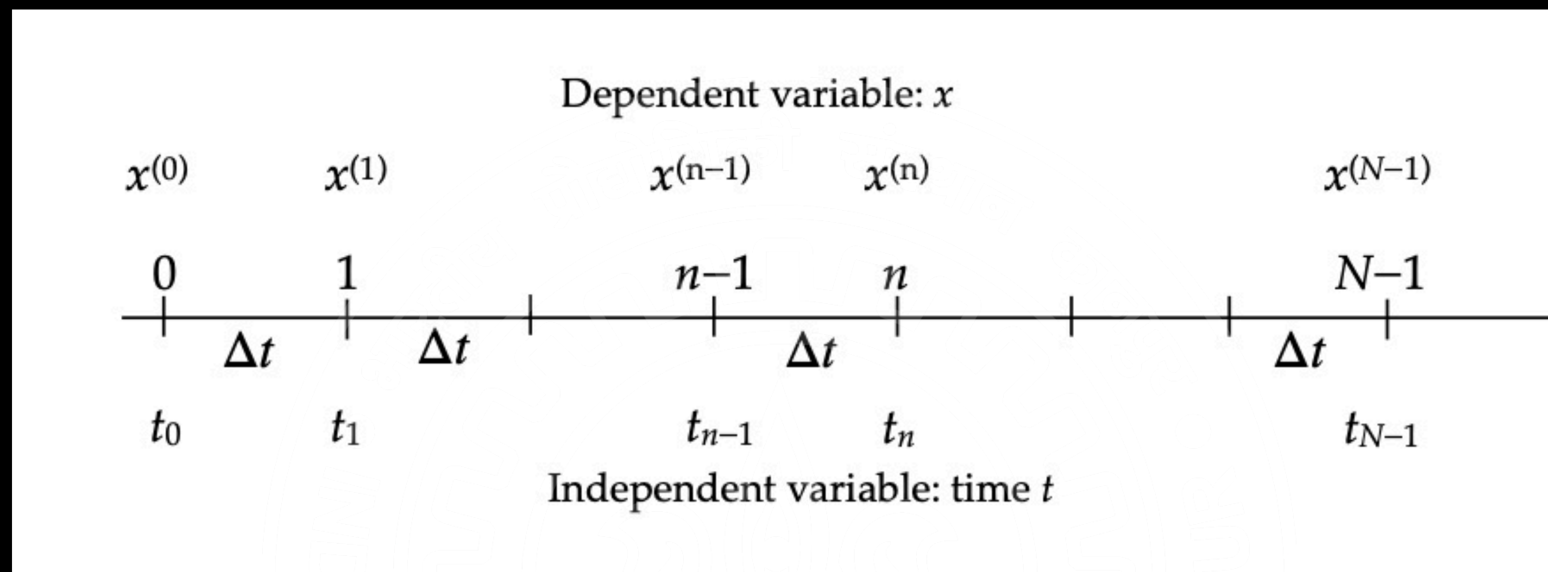
भारतीय प्रौद्योगिकी संस्थान कानपुर

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

ODE

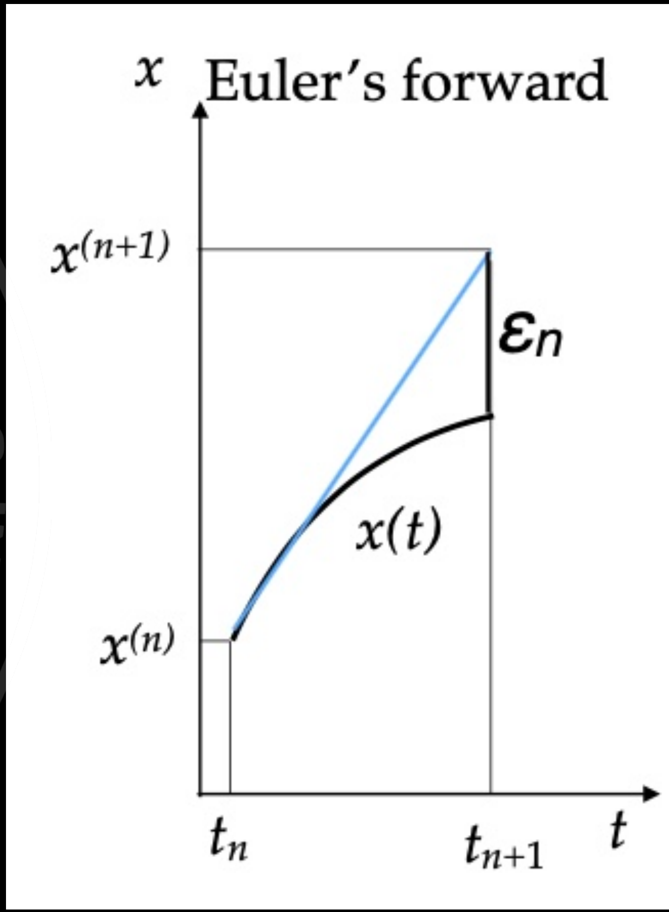
$$\frac{dx}{dt} = \dot{x} = f(x,t)$$

$$\frac{dx}{dt} = \dot{x} = f(x,t)$$

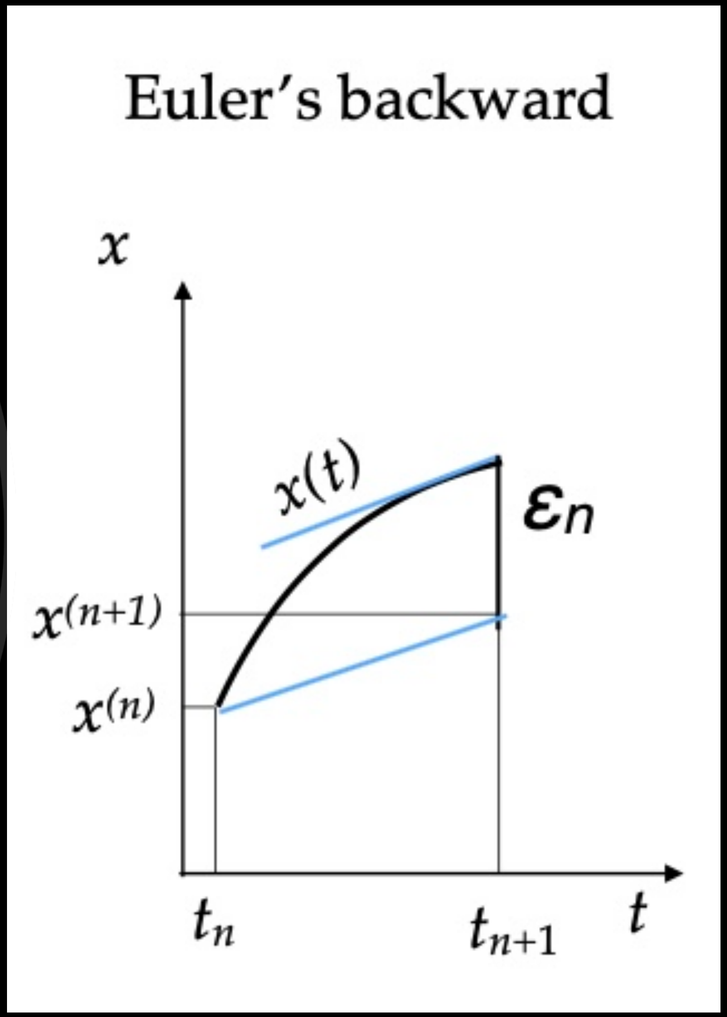


$$x^{(n+1)} - x^{(n)} = \int_{t_n}^{t_{n+1}} f(x,t)\,dt$$

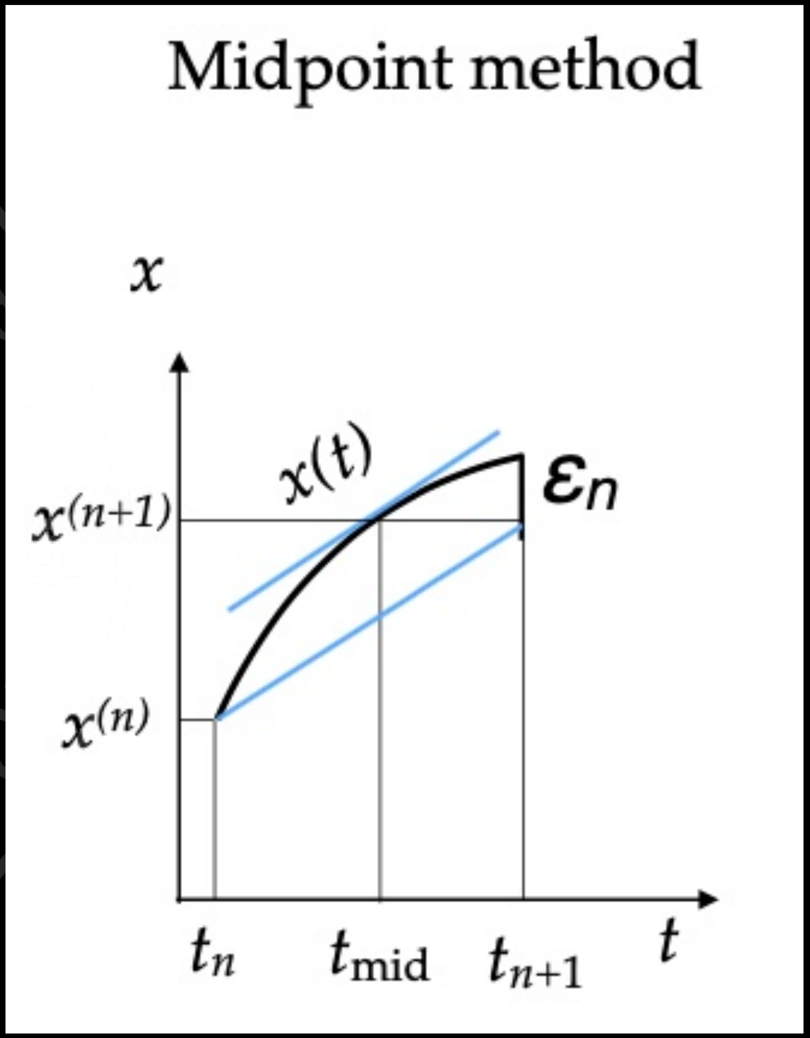
Euler forward method



Euler backward method



Midpoint method



Leapfrog method

Trapezoid method

Euler Forward Method

$$\frac{dx}{dt} = \dot{x} = f(x,t)$$

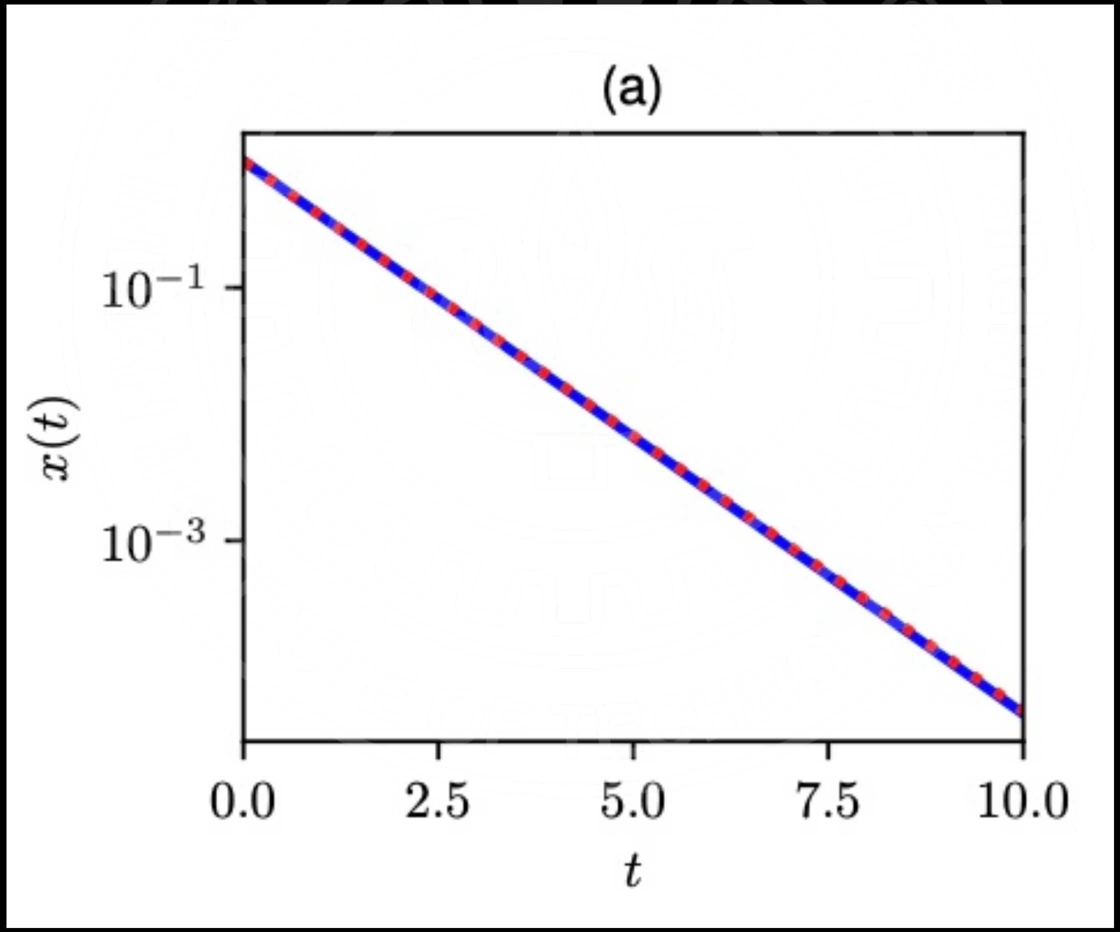
$$x^{(1)} = x^{(0)} + (\Delta t)\, f(x^{(0)}, t_0)$$

$$x^{(n+1)} = x^{(n)} + (\Delta t)\, f(x^{(n)}, t_n)$$

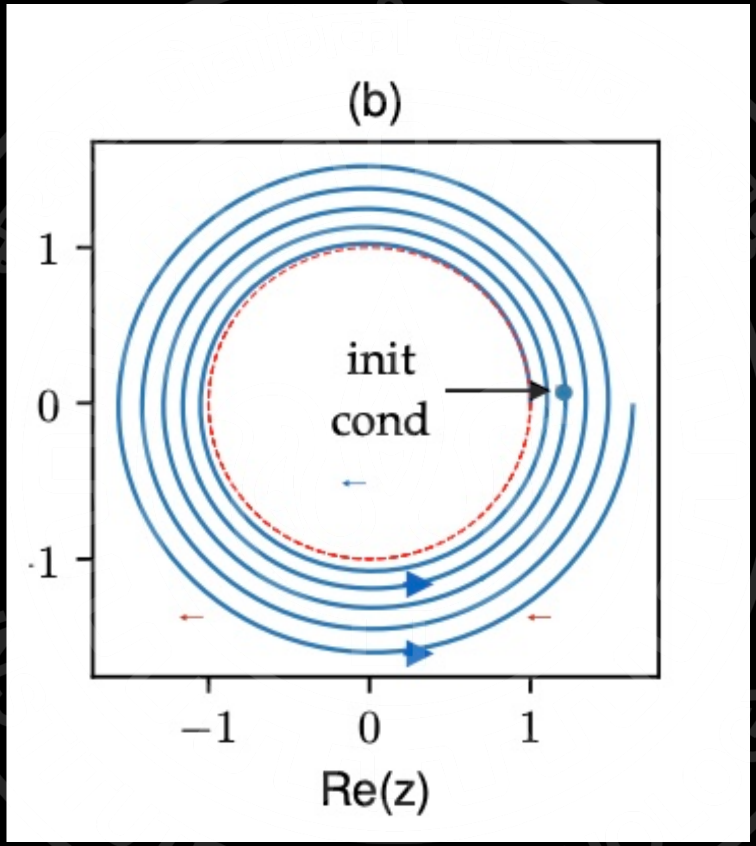
Solving $\dot{x} = -100\,x$

```
def f(x,t):
    return -100*x

def Euler_explicit(f, tinit, tfinal, dt, initcond):
    n = int((tfinal-tinit)/dt)+1   # n-1 divisions
    t = np.linspace(tinit,tfinal,n)
    x = np.zeros(n)
    x[0] = initcond

    for k in range(n-1):
        x[k+1] = x[k] + f(x[k],t[k])*dt
    return t,x
```

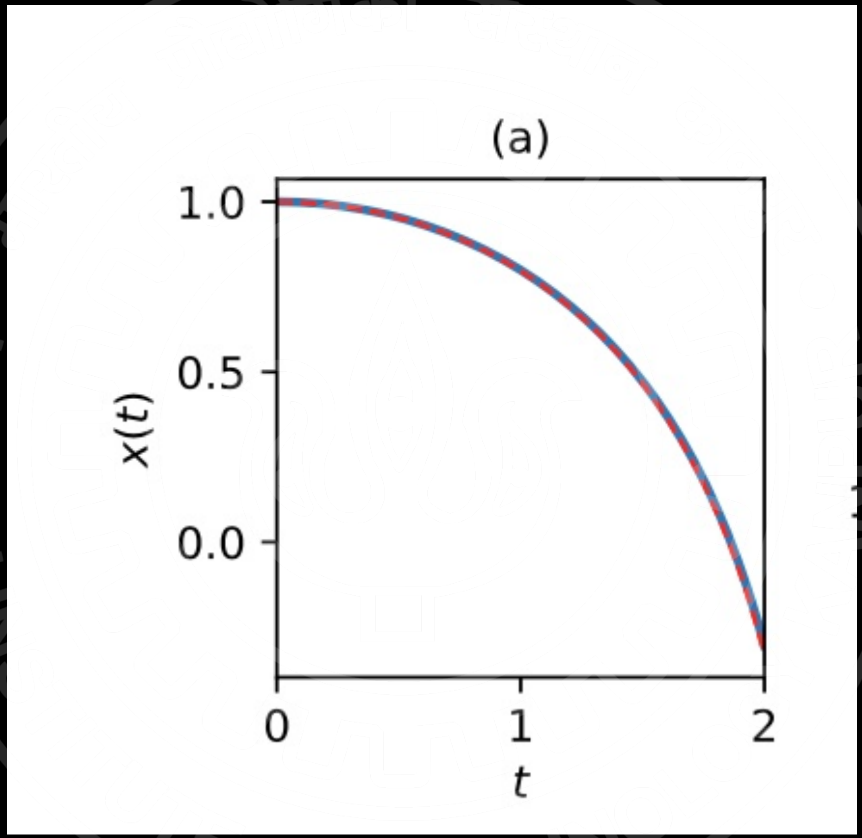
```
tinit = 0
tfinal = 1
dt = 0.01
initcond = 10
t,x = Euler_explicit(f, tinit, tfinal, dt, initcond)
```



$$\dot{z} = -i\pi z$$



$$\dot{x} = -t \exp(-x).$$



Error

$$x^{(n+1)} = x(t_n + \Delta t) = x(t_n) + (\Delta t)\dot{x}(t_n) + \frac{(\Delta t)^2}{2}\ddot{x}(t_n) + H.O.T.$$

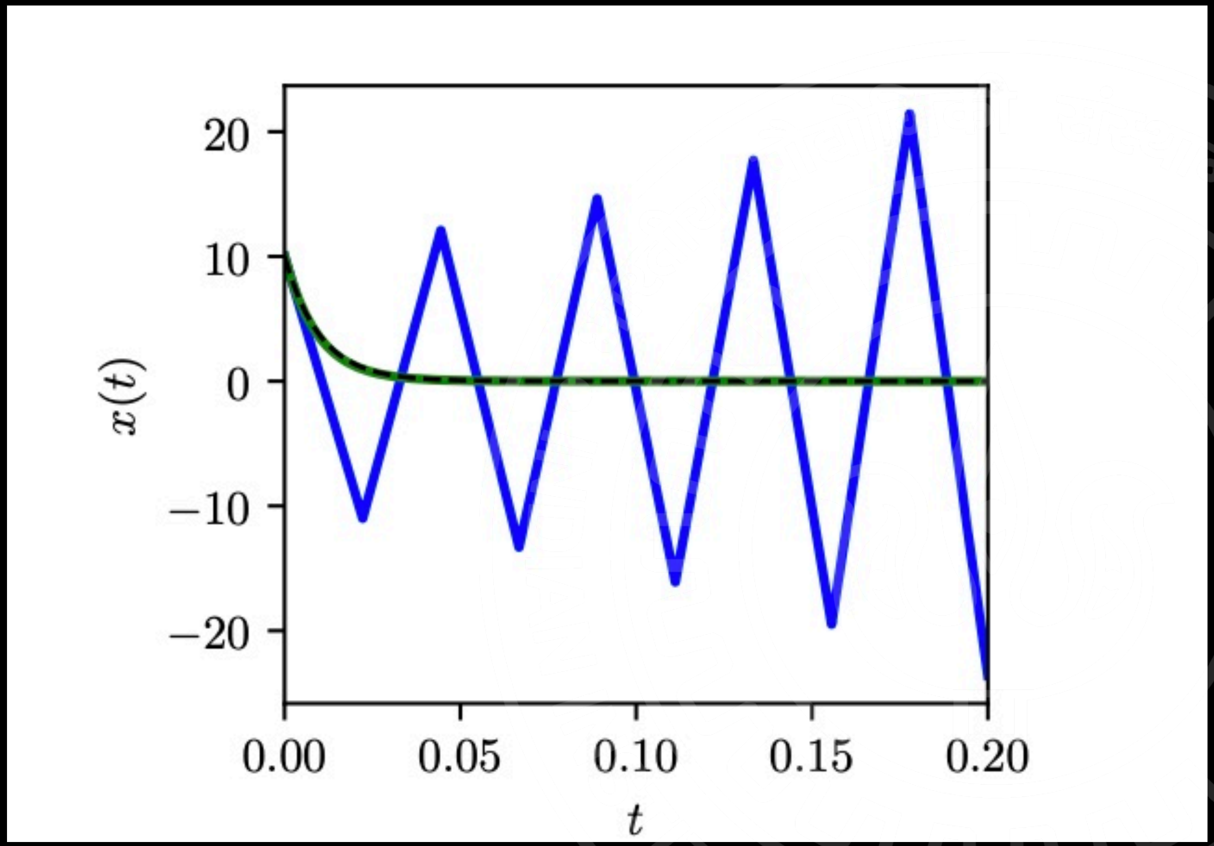
$$= x^{(n)} + (\Delta t) f(x^{(n)}, t_n) + \frac{(\Delta t)^2}{2}\frac{df}{dt}\Big|_{t_n} + H.O.T. \quad \ldots(37)$$

$$\text{Error} = \varepsilon_n = (½)(\Delta t)^2\, \ddot{x}(x(n), t_n)$$

Stability Issues

भारतीय प्रौद्योगिकी संस्थान कानपुर

INDIAN INSTITUTE OF TECHNOLOGY KANPUR



dt = 0.001

0.021

- Stable: A Method is stable if it produces a bounded solution when the solution of the ODE is bounded.

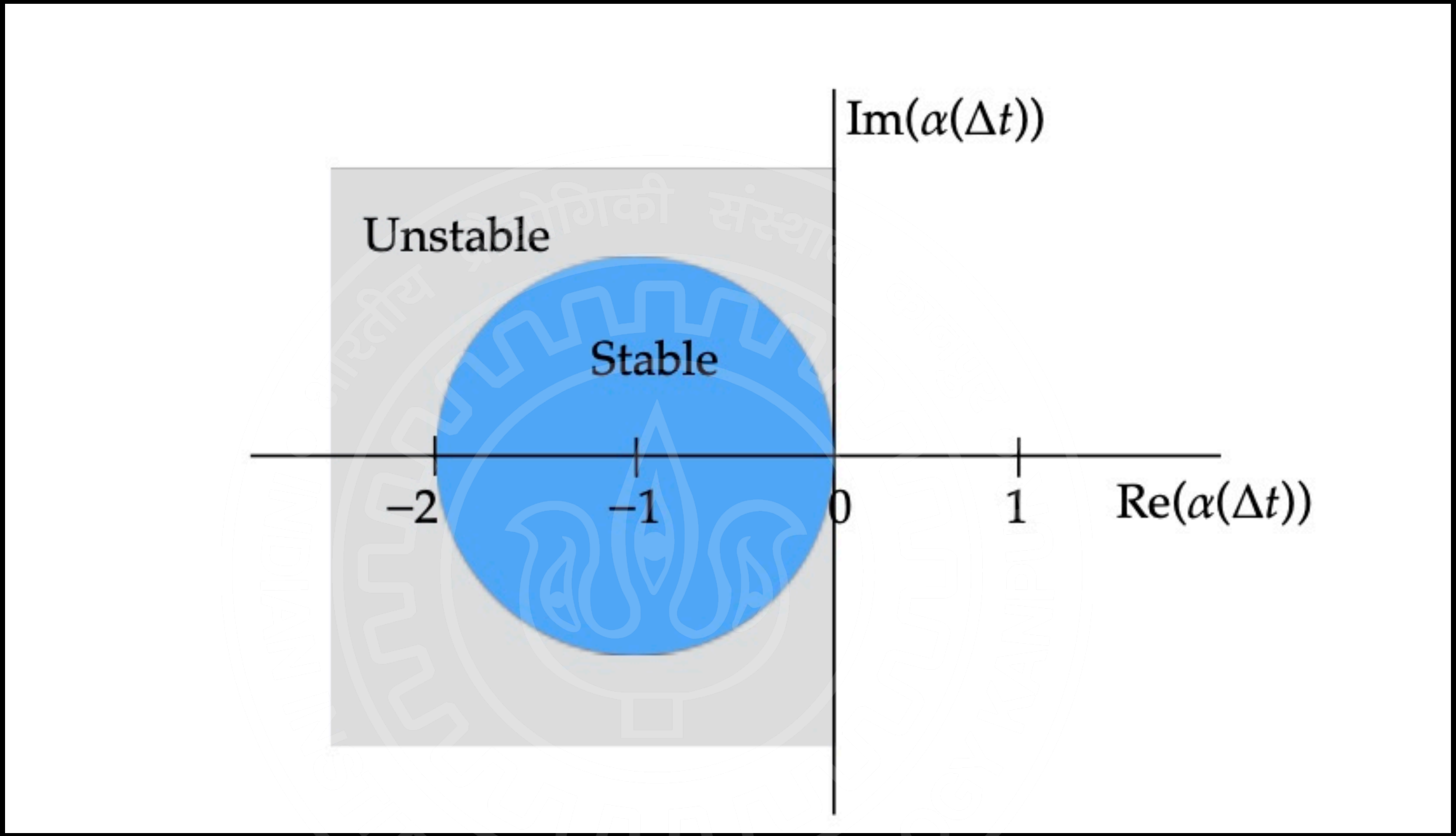
- Unstable: A method which is not stable is said to be unstable.

- Conditionally stable: A method is conditionally stable if it is stable for a set of parameters, and unstable for another set of parameters.

- Unconditionally stable: A method is unconditionally stable if it is stable for all parameter values.

- Unconditionally unstable: A method is unconditionally unstable if it is unstable for all parameter values.

$$\dot{x} = \alpha x$$



Im(α(Δt))
Unstable
Stable
−2
−1
0
1
Re(α(Δt))

Nonlinear equations

$$\dot{x} = f(x,t)$$

$$f(x,t) = f(x^{(n)}, t_n) + (x - x^{(n)})\frac{\partial f}{\partial x}|_{(x^{(n)},t_n)} + (t - t_n)\frac{\partial f}{\partial t}|_{(x^{(n)},t_n)}$$

$$\dot{x}' = \beta + \alpha x' + \gamma t' \ldots$$

Table 24: Regions of stability and instability for Examples 1-4.

ODE	Stablity regime	Instablity regime
$\dot{x} = -x$	$\Delta t < 1$	$\Delta t > 1$
$\dot{z} = -i\pi z$	None	all Δt
$\dot{x} = \alpha x$	$\Delta t < 1/\lvert\alpha\rvert$ (for $\alpha < 0$)	$\Delta t > 1/\lvert\alpha\rvert$ (for $\alpha < 0$)
$\dot{x} = -t\exp(-x)$	all Δt	None
$\dot{x} = x^2 - 100x$	$\Delta t < 1/100$	$\Delta t > 1/100$

Thank you!

ODE Solvers

Implicit Methods

Mahendra Verma

Euler's Implicit Method

$$x^{(n+1)} = x^{(n)} + (\Delta t)\, f(x^{(n+1)}, t_{n+1})$$

$$\dot{x} = \alpha x$$

$$x^{(n+1)} = x^{(n)} + (\Delta t)\, f(x^{(n+1)}, t_{n+1})$$

Difficulties

$$\dot{x} = -t\ \exp(-x).$$

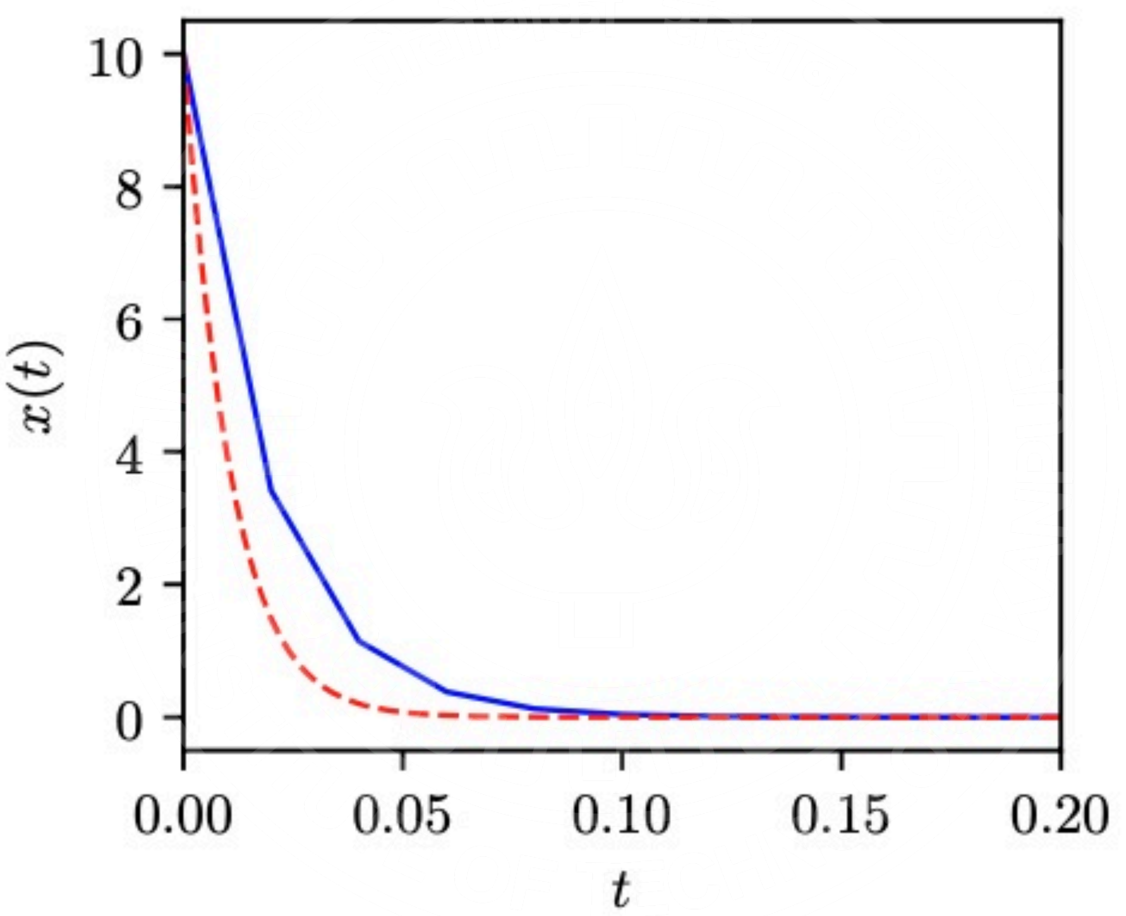
$$x^{(n+1)} = x^{(n)} - \alpha(\Delta t)\ t_{n+1} \exp(-x^{(n+1)})$$

Solving $\dot{x} = x^2 - 100x$

$$x^{(n+1)} = x^{(n)} - \alpha(\Delta t)\,[(x^{(n+1)})^2 - 100\,x^{(n+1)}]$$

$$x^{(n+1)} = \frac{1}{2(\Delta t)}\left[(100(\Delta t) + 1) - \sqrt{(100(\Delta t) + 1)^2 - 4(\Delta t)x^{(n)}}\right]$$

```python
def Euler_implicit(tinit, tfinal, dt, initcond):
    n = int((tfinal-tinit)/dt)+1   # n-1 divisions
    t = np.linspace(tinit,tfinal,n)
    x = np.zeros(n,dtype=complex)
    x[0] = initcond

    for k in range(n-1):
        x[k+1] = ((100*dt+1)-np.sqrt((100*dt+1)**2
                      -4*dt*x[k]))/(2*dt)
    return t,x
```



10
8
6
4
2
0
$x(t)$
0.00
0.05
0.10
0.15
0.20
t

Thank you!

Accuracy

$$x^{(n+1)} = x^{(n)} + (\Delta t) f(x^{(n+1)}, t_{n+1})$$

$$f(x^{(n+1)}, t_{n+1}) = \dot{x}\,|_{n+1} = \dot{x}\,|_n + (\Delta t)\ddot{x}\,|_n + \frac{1}{2}(\Delta t)^2 \dddot{x}\,|_n + \ldots + H.O.T.$$

$$\begin{aligned} x^{(n+1)} &= x^{(n)} + (\Delta t) f(x^{(n+1)}, t_{n+1}) \\ &= x^{(n)} + (\Delta t)\,\dot{x} + (\Delta t)^2\,\ddot{x} + [(\Delta t)^3/2]\,\dddot{x} + H.O.T. \end{aligned}$$

$$x^{(n+1)} = x^{(n)} + (\Delta t)\, f(x^{(n+1)}, t_{n+1})$$

$$= x^{(n)} + (\Delta t)\, \dot{x} + (\Delta t)^2\, \ddot{x} + [(\Delta t)^3/2]\, \dddot{x} + H.O.T.$$

$$\text{Error} = \text{Actual} - \text{Computed} = -(½)(\Delta t)^2\, \ddot{x}(x(n), t_n)\,,$$

Stability Issues

f $\dot{x} = \alpha x$

$$x^{(n+1)} = x^{(n)} + \alpha(\Delta t)x^{(n+1)}$$

भारतीय प्रौद्योगिकी संस्थान कानपुर

INDIAN INSTITUTE OF TECHNOLOGY KANPUR

Thank you!