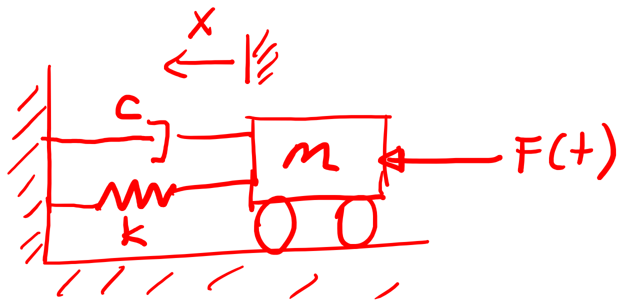
Example:

$$\dot{x} = \frac{dx}{dt} \; ; \quad \ddot{x} = \frac{d^2x}{dt^2}$$



$$\boxed{m\ddot{x} + c\dot{x} + kx = F(t)}$$

Initial conditions: At $t = t_0$ $\;x(t_0) = \alpha_0$, $\dot{x}(t_0) = \alpha_1$

Define: $z_1 = x$; $z_2 = \dot{x} = \frac{dx}{dt} \Rightarrow \dot{z}_2 = \frac{dz_2}{dt} = \ddot{x}$

$$m\dot{z}_2 + cz_2 + kz_1 = F(t) \quad (2)$$

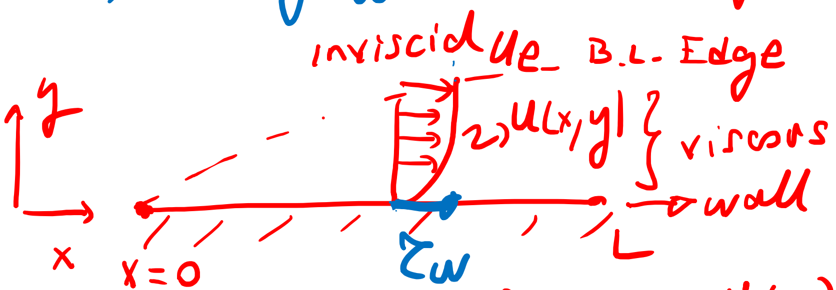
$$\boxed{\frac{dz_1}{dt} = z_2 \quad (1)}$$

$$\boxed{\dot{z}_2 = \frac{dz_2}{dt} = \frac{1}{m}\left(F(t) - cz_2 - kz_1\right) \quad (2)}$$

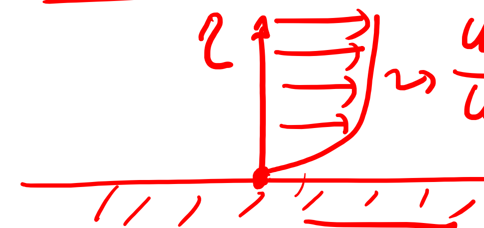
$$\frac{d\vec{z}}{dt} = \vec{f}(t, \vec{z}) \qquad \vec{z} = \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} \; ; \quad \vec{f} = \begin{Bmatrix} z_2 \\ \frac{1}{m}\left(F(t) - cz_2 - kz_1\right) \end{Bmatrix}$$

Example: Solution to 2-D, incompressible, steady, laminar, zero pressure gradient viscous flow over a flat plate $\Rightarrow$ $\boxed{\text{Blasius Solution}}$

$$\tau_w = \mu_w \left(\frac{\partial u}{\partial y}\right)_w$$



$\Rightarrow$



$$\frac{u(\eta)}{u_e} = f'(\eta)$$

$$f = f(\eta) \qquad \frac{df}{d\eta} = f' = \frac{u(\eta)}{u_e} \qquad \eta = y \cdot \sqrt{\frac{u_e}{\nu x}}$$

$$f'' = \frac{d^2 f}{d\eta^2}, \quad f''' = \frac{d^3 f}{d\eta^3}$$

$$\boxed{2f''' + f f'' = 0}$$

$\Rightarrow$ 3rd order non-linear ODE

Define:

$$z_1 = f$$

$$z_2 = \frac{df}{d\eta} = f'$$

$$z_3 = \frac{d^2 f}{d\eta^2} = f''$$

$$2\frac{dz_3}{d\eta} + z_1 \cdot z_3 = 0 \Rightarrow \boxed{\frac{dz_3}{d\eta} = -\frac{z_1 \cdot z_3}{2}} \quad (3)$$

$$(1) \quad \boxed{\frac{dz_1}{d\eta} = z_2} \qquad \boxed{\frac{dz_2}{d\eta} = z_3} \quad (2)$$

$$\frac{d}{d\eta}\underbrace{\begin{Bmatrix} z_1 \\ z_2 \\ z_3 \end{Bmatrix}}_{\vec{z}} = \underbrace{\begin{Bmatrix} z_2 \\ z_3 \\ -\frac{1}{2}(z_1 \cdot z_3) \end{Bmatrix}}_{\vec{f}} \qquad \frac{d\vec{z}}{dt} = \vec{F}(t, \vec{z})$$

→ Need 3 initial conditions defined at $\eta = 0$ (wall)

$$z_1(0) = z_2(0) = 0 \quad , \quad z_3(0) = \left.\frac{d^2 f}{d\eta^2}\right|_{\eta=0} \Rightarrow \text{not known} \quad \propto \tau_w$$

→ Guess $z_3(0)$

→ solve ODE → check if $z_2 = \frac{df}{d\eta} = 1.0$ at $\eta \to \infty$ $(\eta \to 5)$

→ If not, update $z_3(0)$ iterate until converge.

$$g(z_3) = \left.\frac{df}{d\eta}\right|_{\eta=5.0} - 1.0 = 0$$

$$\frac{d\vec{z}}{dt} = \vec{f}(t, \vec{z}) \quad \Rightarrow \quad \frac{d}{dt}\begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} f_1(t, z_1, z_2) \\ f_2(t, z_1, z_2) \end{Bmatrix}$$

$\vec{z} \to \vec{w}$, $z_1 \to w_1$, $z_2 \to w_2$

Example: Apply Modified Euler's Method

$$\vec{w}_{i+1} = \vec{w}_i + \frac{\Delta t}{2}\left[\vec{f}(t_i, \vec{w}_i) + \vec{f}\left(t_i + \Delta t, \vec{w}_i + \Delta t \cdot \vec{f}(t_i, \vec{w}_i)\right)\right]$$

$$(w_{1,2})_{i+1} = (w_{1,2})_i + \frac{\Delta t}{2}\Big\{ f_{1,2}(t_i, w_{1_i}, w_{2_i}) + f_{1,2}\big[t_i + \Delta t,$$
$$w_{1_i} + \Delta t \cdot f_1(t_i, w_{1_i}, w_{2_i}),\ w_{2_i} + \Delta t \cdot f_2(t_i, w_{1_i}, w_{2_i})\big]\Big\}$$

$(w_2)_{i+1} = \cdot \text{ --- --- }$

Numerical Solution to a PDE

1D Time-dependent Heat Equation

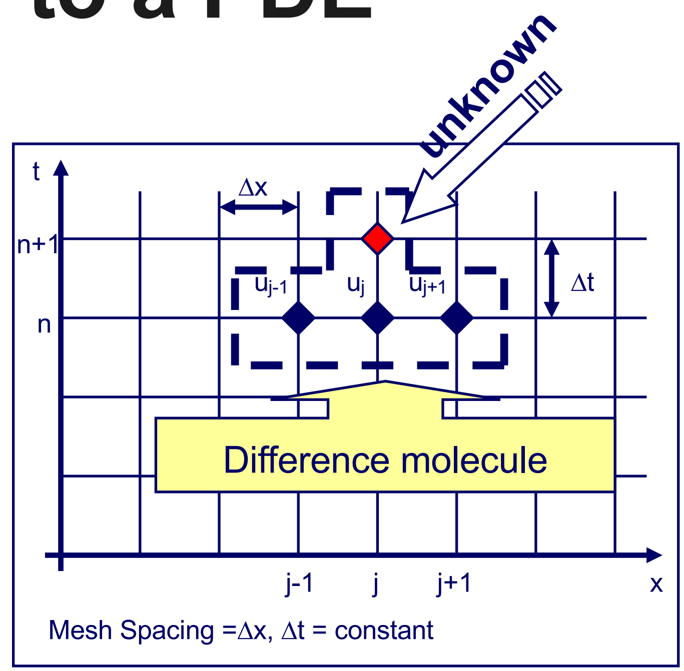
$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

Conditions :

$u(x,0) = u_0(x)$ – Initial Condition

$u(0,t) = u_L(t)$ – Boundary Condition

$u(L,t) = u_R(t)$ – Boundary Condition



Let's use a 2nd order accurate central difference approximation for the spatial derivative and the Euler explicit time integration method.

unknown

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \alpha \frac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2}$$

This is a finite difference equation with a molecule as shown above.

Values at time 'n' are known. We seek values at time 'n+1'.

Operator Notation

Define $\delta u_j \equiv u_{j+1/2} - u_{j-1/2}$ and $\Delta u^n \equiv u^{n+1} - u^n$ and $\lambda \equiv \alpha \dfrac{\Delta t}{\Delta x^2}$

Then

$$\delta^2 u_j = \delta(\delta u_j)$$

$$= \delta(u_{j+1/2} - u_{j-1/2}) = \delta u_{j+1/2} - \delta u_{j-1/2}$$

$$= (u_{j+1} - u_j) - (u_j - u_{j-1}) = u_{j+1} - 2u_j + u_{j-1}$$

Returning to our algorithm: $\dfrac{u_j^{n+1} - u_j^n}{\Delta t} = \alpha \dfrac{u_{j+1}^n - 2u_j^n + u_{j-1}^n}{\Delta x^2}$

In operator notation, this is : $\Delta u_j^n = \lambda \delta^2 u_j^n$

followed by the update step $u_j^{n+1} = u_j^n + \Delta u_j^n$

Euler explicit in operator form

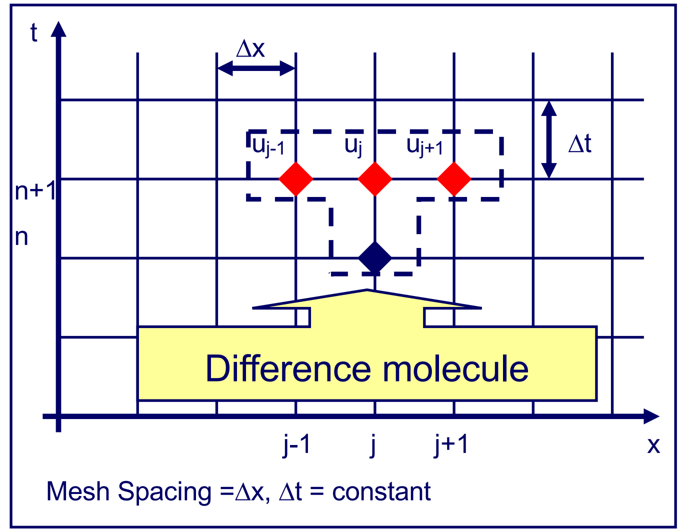
Euler Implicit for 1-D Heat Equation (1)

For our PDE example

$$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

The Euler implicit method is

$$\frac{u_j^{n+1} - u_j^n}{\Delta t} = \alpha \frac{u_{j+1}^{n+1} - 2u_j^{n+1} + u_{j-1}^{n+!}}{\Delta x^2}$$



Note that the only difference is that the RHS is evaluated at 'n+1'.
(Recall that in the Euler explicit method it is evaluate at 'n'.)

$$\frac{\Delta^n u_j}{\Delta t} = \alpha \frac{\delta^2 u_j^{\,n+1}}{\Delta x^2}$$

$$\Delta u_j^n = \lambda \delta^2 \left(u_j^{\,n} + \Delta u_j^n \right)$$

Note the change in the difference molecule above. This algorithm results in a coupled (tri-diagonal) set of equations that we can easily solve.

Euler Implicit for 1-D Heat Equation (2)

From the previous slide, we have

$$\Delta u_j^n = \lambda\delta^2\left(u_j^{\ n} + \Delta u_j^n\right)$$

$$(1-\lambda\delta^2)\Delta u_j^n = \lambda\delta^2 u_j^{\ n}$$

In finite difference form, this is:

$$-\lambda\Delta u_{j-1}^n + (1+2\lambda)\Delta u_j^n - \lambda\Delta u_{j+1}^n = \lambda\delta^2 u_j^{\ n}$$

The linear system in matrix form is thus

$$\begin{pmatrix} b & c & & & \\ a & b & c & & \\ & \ddots & \ddots & \ddots & \\ & & a & b & c \\ & & & a & b \end{pmatrix} \begin{pmatrix} \\ \Delta u_{j-1}^n \\ \Delta u_j^n \\ \Delta u_{j+1}^n \\ \\ \end{pmatrix} = \begin{pmatrix} \\ \lambda\delta^2 u_{j-1}^n \\ \lambda\delta^2 u_j^n \\ \lambda\delta^2 u_{j+1}^n \\ \\ \end{pmatrix}$$

At the interior nodes

$$a = -\lambda$$

$$b = 1+2\lambda$$

$$c = -\lambda$$

Semi-Discrete Form of the Governing Equation

We can discretize our PDE in space only, which leads to:

$$\frac{\partial u_j}{\partial t} = \alpha \frac{\delta^2 u_j}{\Delta x^2}$$

We call this a semi-discrete form since it is discrete in space and continuous in time. (We could discretize in time and leave it continuous in space – this would be another semi-discrete form).

Note that the semi-discrete form shown above is nothing more than a coupled set of ordinary differential equations in the form of an Initial Value Problem. You are free to integrate it with any of the Adams methods, Runge-Kutta, etc..

The choices you make in the spatial discretization and time integration impact the **accuracy** of the solution and **stability** of the algorithm.

PDE's and Root Finding - 1

Euler implicit time integration, 2nd Order Central Difference in Space

Consider our PDE $$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$$

We can write this as: $$\frac{\partial u}{\partial t} + R(u) = 0 \qquad R(u) = -\alpha \frac{\partial^2 u}{\partial x^2}$$

where $\boldsymbol{R(u)}$ is the steady-state residual. If you are only interested in the steady-state solution, then this is clearly a root-finding problem. Recall, that the Euler implicit algorithm is

$$\frac{\Delta u_j^n}{\Delta t} + R(u_j^{n+1}) = 0; \quad \text{or} \quad \frac{\Delta u_j^n}{\Delta t} + R(u_j^n) + \left[\frac{\partial R}{\partial u}\right]_j^n \Delta u_j^n + \cdots = 0$$

The higher order terms are neglected in the linearization.

PDE's and Root Finding - 2

Euler implicit time integration, 2nd Order Central Difference in Space

Thus, in this form, the Euler implicit algorithm is

$$\frac{\Delta u_j^n}{\Delta t} + R(u_j^n) + \left[\frac{\partial R}{\partial u}\right]_j^n \Delta u_j^n = 0$$

Or, in operator form:

$$\left[\frac{1}{\Delta t} + \left(\frac{\partial R}{\partial u}\right)_j^n\right]\Delta u_j^n = -R(u_j^n)$$

The term in brackets "operates" on Δu_j^n

Note that as $\Delta t \to \infty$, this is Newton's root finding method. For the given problem, with the definition of $\boldsymbol{R}$, we have $R(u) = -\alpha \frac{\partial^2 u}{\partial x^2}$

$$\left[\frac{1}{\Delta t} - \alpha \frac{\partial^2}{\partial x^2}\right]\Delta u_j^n = \alpha \frac{\partial^2 u_j^n}{\partial x^2}$$

(Semi-discrete)

PDE's and Root Finding - 3

Euler implicit time integration, 2nd Order Central Difference in Space

Replace the spatial derivative by a 2nd Order central difference operator

$$\left[\frac{1}{\Delta t} - \alpha \frac{\delta^2}{\Delta x^2}\right] \Delta u_j^n = \alpha \frac{\delta^2 u_j^n}{\Delta x^2} \qquad \text{(Fully-discrete)}$$

Multiply by Δt to obtain the algorithm we had earlier

$$\left[1 - \lambda \delta^2\right] \Delta u_j^n = \lambda \delta^2 u_j^n$$

Thus the two approaches are equivalent and we have drawn the connection between the Euler implicit algorithm and root-finding.

Special note: Interestingly, we can also define a residual for the time-dependent problem by adding a pseudo-time variable and iterating on a fixed time level.