Technische Universität Braunschweig

Introduction to PDEs and Numerical Methods

Lecture 6:

Numerical solution of the heat equation with FD method: method of lines, Euler forward, Euler backward, the Theta method, and their stability

Dr. Noemi Friedman, 23.11.2016.

Overview of the course

- Introduction (definition of PDEs, classification, basic math, introductory examples of PDEs)

- Analytical solution of elementary PDEs (Fourier series/transform, separation of variables, Green's function)

- Numerical solutions of PDEs:

  - Finite difference method

  - Finite element method

Overview of this lecture

- Finite difference operators
- The heat equation
  - Analytical solution
  - Semidiscretization- Method of Lines (spatial discretization)
  - Time discretization
    - Euler foward
    - Euler backward
    - The theta-method

  - Consisteny, stability, convergence

Finite Difference Method – derivation of difference operators

Derivation of $u'(x)$

$$u(x+h) = u(x) + u'(x)h + \frac{1}{2}u''(x)h^2 + O(h^3) \qquad (1)$$

$$u(x-h) = u(x) - u'(x)h + \frac{1}{2}u''(x)h^2 + O(h^3) \qquad (2)$$

Subtracting from eq. (1) eq. (2) results in:

$$u(x+h) - u(x-h) = 2u'(x)h + O(h^3)$$

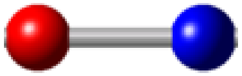
$$u'(x) = \frac{u(x+h) - u(x-h)}{2h} + O(h^2)$$

$$\boxed{u'_k = \frac{u_{k+1} - u_{k-1}}{2h} + O(h^2)}$$

From only (1) (EULER FORWARD)

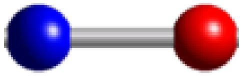
$$u'(x) = \frac{u(x+h) - u(x)}{h} + O(h)$$

$$\boxed{u'_k = \frac{u_{k+1} - u_k}{h} + O(h)}$$



From only (2) EULER BACKWARD

$$u'(x) = \frac{u(x) - u(x-h)}{h} + O(h)$$



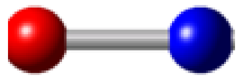
$$\boxed{u'_{k+1} = \frac{u_{k+1} - u_k}{h} + O(h)}$$

Finite Difference Method – derivation of difference operators

Example for using the **two point stencil**

Forward differences – explicit method

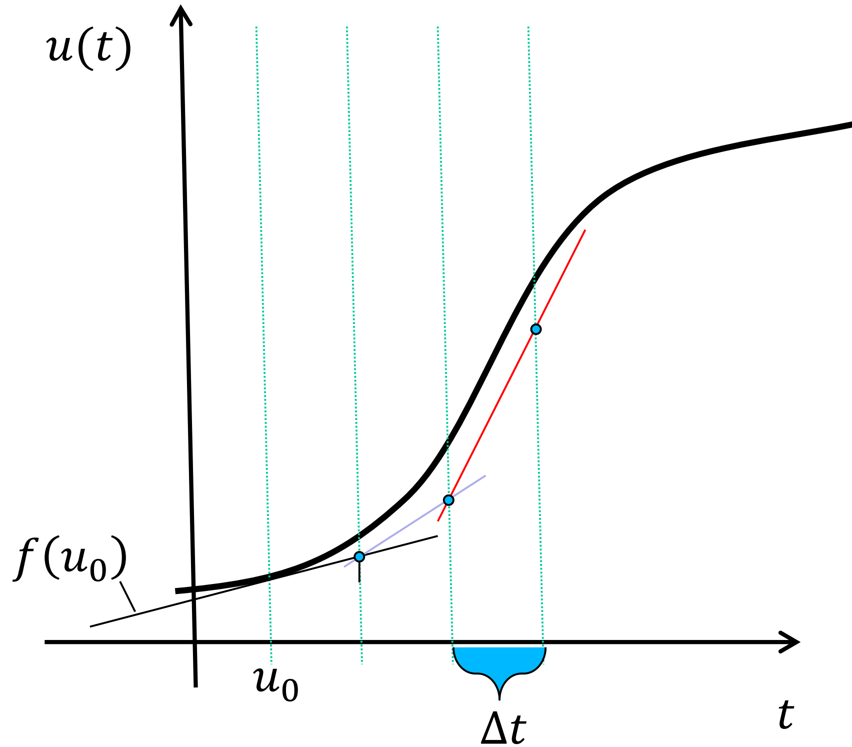
$$\boxed{u'_k = \frac{u_{k+1} - u_k}{\Delta t} + O(\Delta t)}$$



$$u'(t) = f\big(u(t)\big) \qquad B.C.: u(0) = \bar{u} = u_0$$

$$u'_k = \frac{u_{k+1} - u_k}{\Delta t} = f(u_k)$$

$$u_{k+1} = u_k + \Delta t f(u_k)$$

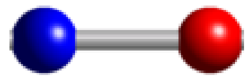


Finite Difference Method – derivation of difference operators

Example for using the **two point stencil**

Backward differences – implicit method

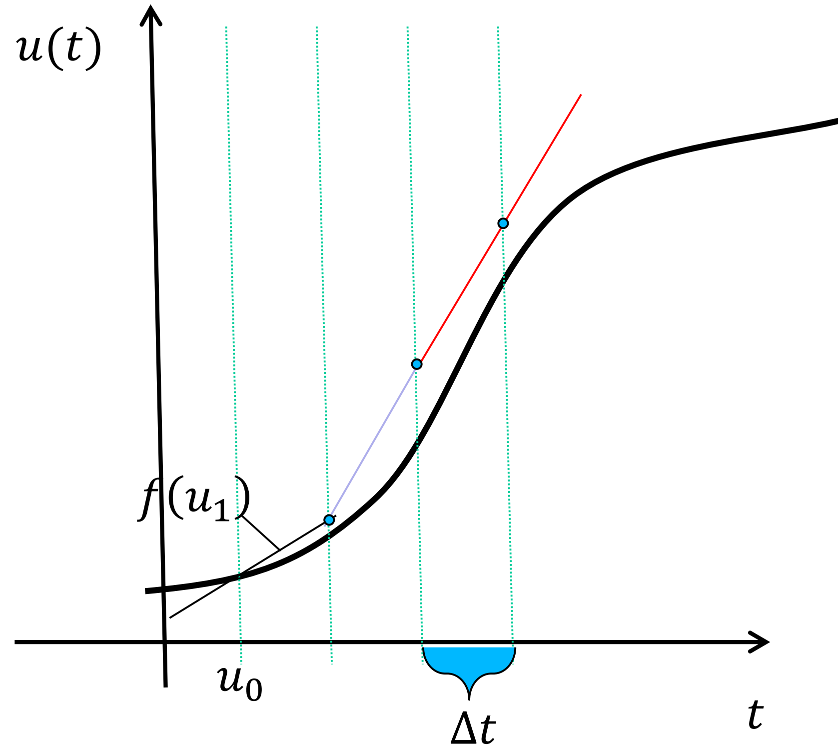
$$\boxed{u'_{k+1} = \frac{u_{k+1} - u_k}{\Delta t} + O(\Delta t)}$$



$$u'(t) = f\big(u(t)\big) \qquad B.C.: u(0) = \bar{u} = u_0$$

$$u'_{k+1} = \frac{u_{k+1} - u_k}{\Delta t} = f(u_{k+1})$$

$$u_{k+1} = \Delta t f(u_{k+1}) + u_k$$



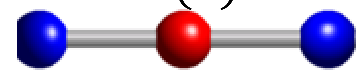
Theta method

$$\left.\begin{array}{l} \boxed{u'_{k} = \dfrac{u_{k+1} - u_k}{\Delta t} + O(\Delta t)} \\ \boxed{u'_{k+1} = \dfrac{u_{k+1} - u_k}{\Delta t} + O(\Delta t)} \end{array}\right\} u'_{k+\theta} = \frac{u_{k+1} - u_k}{\Delta t} = \theta f(u_{k+1}) + (1-\theta) f(u_k)$$

Finite Difference Method – derivation of difference operators

Derivation of $u''(x)$ – **the three point stencil**

$u''(x)$



$$u(x+h) = u(x) + u'(x)h + \frac{1}{2}u''(x)h^2 + \frac{1}{3!}u'''(x)h^3 + \frac{1}{4!}u^{IV}(x)h^4 + \frac{1}{5!}u^{V}(x)h^5 + O(h^6) \quad (3)$$

$$u(x-h) = u(x) - u'(x)h + \frac{1}{2}u''(x)h^2 - \frac{1}{3!}u'''(x)h^3 + \frac{1}{4!}u^{IV}(x)h^4 - \frac{1}{5!}u^{V}(x)h^5 + O(h^6) \quad (4)$$

Adding eq. (3) and eq. (4) results in:

$$u(x+h) + u(x-h) = 2u(x) + 0 + u''(x)h^2 + 0 + \frac{2}{4!}u^{IV}(x)h^4 + 0 + O(h^6)$$

$$u''(x) = \frac{u(x+h) - 2u(x) + u(x-h)}{h^2} + O(h^2)$$

truncation error:

$$\frac{1}{12}\frac{d^4u}{dx^4}h^2 + O(h^4)$$

truncation constant

$$\boxed{u''_j = \frac{u_{j+1} - 2u_j + u_{j-1}}{h^2} + O(h^2)}$$

Finite Difference Method – derivation of difference operators

Derivation of $u''(x)$ – **the five point stencil**

Adding eq. (3) and eq. (4) results in:

$$u(x+h)+u(x-h)=2u(x)+0+u''(x)h^2+0+\frac{2}{4!}u^{IV}(x)h^4+0+O(h^6)$$

$$u(x+2h)=u(x)+2u'(x)h+2u''(x)h^2+\frac{8}{3!}u'''(x)h^3+\frac{16}{4!}u^{IV}(x)h^4+\frac{32}{5!}u^{V}(x)h^5+O(h^6) \quad (5)$$

$$u(x-2h)=u(x)-2u'(x)h+2u''(x)h^2-\frac{8}{3!}u'''(x)h^3+\frac{16}{4!}u^{IV}(x)h^4-\frac{32}{5!}u^{V}(x)h^5+O(h^5) \quad (6)$$

$$u(x+2h)+u(x-2h)=2u(x)+0+4u''(x)h^2+0+\frac{32}{4!}u^{IV}(x)h^4+0+O(h^6)$$

$$u''(x)=\frac{-u(x-2h)+16u(x-h)-30u(x)+16u(x+h)-u(x+2h)}{12h^2}+O(h^4)$$

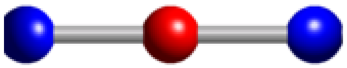
$$\boxed{u''_j=\frac{-u_{j-2}+16u_{j-1}-30u_j+16u_{j+1}-u_{j+2}}{12h^2}+O(h^4)}$$

$u''(x)$

Numerical solution of the heat equation
1.) Method of lines (semi-discretized heat equation)

$$\frac{\partial u(x,t)}{\partial t} - \beta^2 \frac{\partial^2 u}{\partial x^2} = 0 \quad \Longrightarrow \quad \frac{\partial u(x,t)}{\partial t} = \beta^2 \frac{\partial^2 u}{\partial x^2}$$

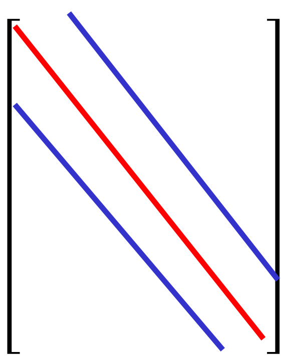
Approximate second derivate with the three point stencil (spatial discretisation of the heat eq.)



$$\frac{\partial^2 u_j}{\partial x^2} = \frac{1}{h^2}(u_{j+1} - 2u_j + u_{j-1}) + O(h^2)$$

$$\frac{\partial u_j(t)}{\partial t} = \frac{\beta^2}{h^2}\left(u_{j+1} - 2u_j + u_{j-1}\right) + O(h^2)$$

$$\frac{d}{dt}\begin{bmatrix} u_1(t) \\ \vdots \\ u_{j-1}(t) \\ u_j(t) \\ u_{j+1}(t) \\ \vdots \\ u_{N-1}(t) \end{bmatrix} = \frac{\beta^2}{h^2}$$



$$\begin{bmatrix} u_1(t)) + \frac{\beta^2}{h^2} u_0(t) \\ \vdots \\ u_{j-1}(t) \\ u_j(t) \\ u_{j+1}(t) \\ \vdots \\ u_{N-1}(t)) - \frac{\beta^2}{h^2} u_0(t) \end{bmatrix}_{-1}.$$

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t)$$

$$\begin{bmatrix} 2 & -1 & 0 & & 0 \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ 0 & & & -1 & 2 \end{bmatrix}$$

Numerical solution of the heat equation
1.) Method of lines (semi-discretized heat equation)

$$\frac{\partial u(x,t)}{\partial t} - \beta^2 \frac{\partial^2 u}{\partial x^2} = f(x,t)$$

truncation error:

$$\frac{\partial}{\partial t}\boldsymbol{u} = \frac{\beta^2}{h^2}\left(u_{j+1} - 2u_j + u_{j-1}\right) + O\left(h^2\right) = f_j \qquad \frac{1}{12}\frac{d^4 u}{dx^4}h^2 + O(h^4)$$

truncation constant

$$\frac{\partial}{\partial t}\boldsymbol{u} = \frac{-\beta^2}{h^2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & & & \\ & & & 2 & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix}\begin{bmatrix} u_1(t) \\ \vdots \\ u_{j-1}(t) \\ u_j(t) \\ u_{j+1}(t) \\ \vdots \\ u_{N-1}(t) \end{bmatrix} + \begin{bmatrix} f_1(t) + \frac{\beta^2}{h^2}u_0(t) \\ \vdots \\ f_{j-1}(t) \\ f_j(t) \\ f_{j+1}(t) \\ \vdots \\ f_{N-1}(t) + \frac{\beta^2}{h^2}u_N(t) \end{bmatrix}$$

$$\frac{\partial}{\partial t}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{u}(t) + \boldsymbol{f}(t)$$

Numerical solution of the heat equation
1.) Method of lines (semi-discretized heat equation)

Solve analyticaly the system of ODEs:

$$\boxed{\frac{d}{dt}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{u}(t)}$$

(see slide: 15)

$$a := \frac{-2\beta^2}{h^2} \qquad b := \frac{\beta^2}{h^2} \qquad \boldsymbol{A} = \begin{bmatrix} a & b & & & & \\ b & a & b & & & \\ & b & a & & & \\ & & & a & b & \\ & & & b & a & b \\ & & & & b & a \end{bmatrix}$$

1.) Find eigenvalues (λ_i) and eigenvectors ($\boldsymbol{v_i}$) of $\boldsymbol{A}$:

$$\lambda_i = a + 2b\cos\left(\frac{i\pi}{N}\right) = \frac{2\beta^2}{h^2}\left(\cos\left(\frac{i\pi}{N}\right) - 1\right) \qquad \boldsymbol{v_i} = \begin{bmatrix} v_i^1 \\ \vdots \\ v_i^{N-1} \end{bmatrix} \qquad v_i^k = \sin\left(\frac{ik\pi}{N}\right)$$

$$i, k = 1..N-1$$

2.) Write initial condition in the basis of the eigenvectors: $\boldsymbol{u}(0) = C_i \boldsymbol{v}_i$

3.) Look for the solution in the form: $\boldsymbol{u}(t) = \sum_i \alpha_i(t)\boldsymbol{v_i} \qquad \alpha_i(0) = C_i$

The solution to the homogenous equations: $\alpha_i(t) = C_i e^{\lambda_i(t-t_0)} \quad \boldsymbol{u_i}(t) = C_i e^{\lambda_i(t-t_0)}\boldsymbol{v_i}$

$$\boxed{\mathbf{u_h}(t) = \sum_i c_i e^{\lambda_i(t-t_0)}\boldsymbol{v_i}}$$

Numerical solution of the heat equation
1.) Method of lines (semi-discretized heat equation)

$$\frac{d}{dt}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{u}(t) + \boldsymbol{f}(t)$$

$$\mathbf{u_h}(t) = \sum_i c_i e^{\lambda_i (t-t_0)} \boldsymbol{v_i}$$

1,2.) See steps 1,2 for the homogenous equation (get eigenvector and eigenvalue of $\boldsymbol{A}$)

3.) Write r.h.t in the basis of the eigenvectors : $\boldsymbol{f}(t) = \sum_i f_i \boldsymbol{v_i}$

4.) Look for the solution in the form: $\boldsymbol{u}(t) = \sum_i u_i(t) \boldsymbol{v_i}$

$$\frac{d}{dt}\boldsymbol{u} - \boldsymbol{A}\boldsymbol{u}(t) = \frac{d}{dt}\sum_i u_i(t)\boldsymbol{v_i} - \boldsymbol{A}\sum_i u_i(t)\boldsymbol{v_i} = \sum_i \frac{d}{dt}u_i(t)\boldsymbol{v_i} - \sum_i u_i(t)\boldsymbol{A}\boldsymbol{v_i}$$

$$= \sum_i \left(\frac{d}{dt}u_i(t) - u_i(t)\lambda_i\right)\boldsymbol{v_i}$$

5.)

$$\sum_i \left(\frac{d}{dt}u_i(t) - u_i(t)\lambda_i\right)\boldsymbol{v_i} = \sum_i f_i \boldsymbol{v_i} \Longrightarrow \left(\frac{d}{dt}u_i(t) - u_i(t)\lambda_i\right) = f_i$$

Numerical solution of the heat equation
1.) Method of lines (semi-discretized heat equation)

$$\boxed{\frac{d}{dt}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{u}(t) + \boldsymbol{f}}$$

Instead of analyzing stability of the inhomogenous case, we discretize the homogenous one. We can do that, because of the following.
Let's take a stationary function $\boldsymbol{u_0}$ for which the equation:

$$\frac{d}{dt}\boldsymbol{u_0} = \boldsymbol{A}\boldsymbol{u_0} + \boldsymbol{f} = 0$$ holds.

And let's suppose, that the solution can be written in the form $\boldsymbol{u}(t) = \boldsymbol{u_0} + \boldsymbol{v}(t)$

r.h.s:

$$\boldsymbol{A}\boldsymbol{u}(t) + \boldsymbol{f} = \underbrace{\boldsymbol{A}\boldsymbol{u_0} + \boldsymbol{f}}_{=0} + \boldsymbol{A}\boldsymbol{v}(t) = \boldsymbol{A}\boldsymbol{v}(t)$$

l.h.s:

$$\frac{d}{dt}\boldsymbol{u}(t) = \frac{d}{dt}\big(\boldsymbol{u_0} + \boldsymbol{v}(t)\big) = \frac{d}{dt}\boldsymbol{v}(t)$$

$$\boxed{\frac{d}{dt}\boldsymbol{v}(t) = \boldsymbol{A}\boldsymbol{v}(t)}$$

Heat equation
Comparism of the solutions: analytical and method of lines

Homogeneous equation (no internal source term): $\boxed{\frac{d}{dt}\boldsymbol{u} = \boldsymbol{A}\boldsymbol{u}(t)}$

Analytical solution with hom. BC:

$$u(x,t) = \sum_j d_j \, e^{-\omega_j^2 \beta t} \sin(\omega_j x) \qquad \omega_j = \frac{j\pi}{l} \qquad j = 0,1,2..$$

$\Rightarrow$ as $t \to \infty \quad u \to 0$

With method of lines:

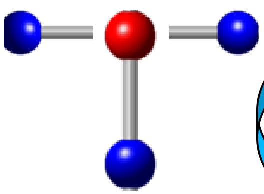
$$\boldsymbol{u}(t) = \sum_i c_i e^{\lambda_i (t - t_0)} \boldsymbol{v_i}$$

$$\lambda_i = \underbrace{\frac{2\beta^2}{h^2}}_{>0} \left(\underbrace{\cos\left(\frac{i\pi}{N}\right)}_{<1} - 1 \right)$$

$\Rightarrow$ as $t \to \infty \quad u \to 0$

Numerical solution of the heat equation
2.) Euler forward - explicit

Euler forward: approximate time derivate with the forward difference



$$\frac{\partial u_j(t)}{\partial t} = \frac{\beta^2}{h^2}\left(u_{j-1} - 2u_j + u_{j+1}\right) + O(h^2) \qquad \frac{d}{dt}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t)$$

$$\boxed{\dot{u}_{j,n} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t)} \qquad \boxed{\dot{\boldsymbol{u}}_n = \frac{\boldsymbol{u}_{n+1} - \boldsymbol{u}_n}{\Delta t} = \mathbf{A}\mathbf{u}_n}$$

$$\frac{u_{j,n+1} - u_{j,n}}{\Delta t} = \frac{\beta^2}{h^2}\left(u_{j-1,n} - 2u_{j,n} + u_{j+1,n}\right) + O(h^2) + O(\Delta t)$$

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t\,\mathbf{A}\mathbf{u}_n$$

$$= \underbrace{(\mathbf{I} + \Delta t\,\mathbf{A})}_{\mathbf{B}}\mathbf{u}_n$$

Stable? (does it give decaying solution?)

$$B = \frac{-\Delta t \cdot \beta^2}{h^2}\begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & & & \\ & & & 2 & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix} + \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

Stable? (does it give decaying solution?)

$\mathbf{u}_{n+1} = \mathbf{B}\mathbf{u}_n$ $\quad \mathbf{u}_{n+2} = \mathbf{B}^2\mathbf{u}_n$ $\quad \mathbf{u}_{n+3} = \mathbf{B}^3\mathbf{u}_n$ $\quad \mathbf{u}_{n+k} = \mathbf{B}^k\mathbf{u}_n$

Some discussion about egienvalues and eigenvectors:

$$\mathbf{A}\mathbf{v}_\mathbf{i} = \lambda_i \mathbf{v}_\mathbf{i}$$

$$\mathbf{A}\mathbf{V} = \mathbf{D}\mathbf{V}$$

$$\mathbf{V}^{-1}\mathbf{A}\mathbf{V} = \mathbf{D}$$

$\Rightarrow$ $\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$ is a diagonal matrix, with the eigenvalues in the diagonal.

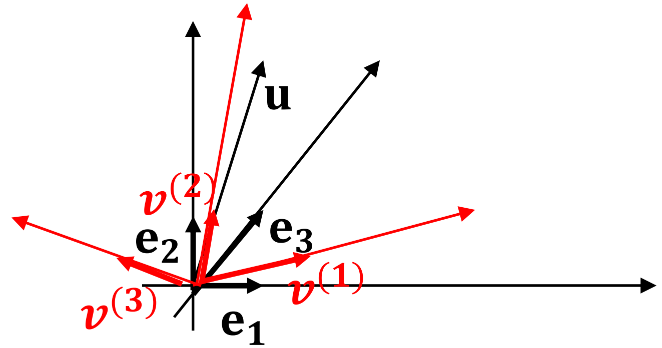
$$\mathbf{D} = \begin{bmatrix} \lambda_1 & & & & & \\ & \lambda_2 & & & & \\ & & \ddots & & & \\ & & & \ddots & & \\ & & & & \lambda_{n-1} & \\ & & & & & \lambda_n \end{bmatrix}$$

$$\mathbf{V} = \begin{array}{c} \begin{matrix} \mathbf{v_1} & \mathbf{v_2} & & & \mathbf{v_{n-1}} & \mathbf{v_n} \end{matrix} \\ \begin{bmatrix} | & | & \dots & \dots & | & | \end{bmatrix} \end{array}$$

Eigenvalues, eigenvectors
change of basis (coordinate system)

How to write an arbitrary vector, $\mathbf{u}$, in a new basis $\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \boldsymbol{v}^{(3)}$

What is $\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$?



$$\mathbf{u} = u_1\mathbf{e_1} + u_2\mathbf{e_2} + u_3\mathbf{e_3}$$

$$\mathbf{u} = \tilde{u}_1\boldsymbol{v}^{(1)} + \tilde{u}_2\boldsymbol{v}^{(2)} + \tilde{u}_3\boldsymbol{v}^{(3)}$$

$$\boldsymbol{v}^{(1)} = v_1^{(1)}\mathbf{e_1} + v_2^{(1)}\mathbf{e_2} + v_3^{(1)}\mathbf{e_3}$$

$$\boldsymbol{v}^{(2)} = v_1^{(2)}\mathbf{e_1} + v_2^{(2)}\mathbf{e_2} + v_3^{(2)}\mathbf{e_3}$$

$$\boldsymbol{v}^{(3)} = v_1^{(3)}\mathbf{e_1} + v_2^{(3)}\mathbf{e_2} + v_3^{(3)}\mathbf{e_3}$$

$$\begin{aligned}\mathbf{u} = \tilde{u}_1&\left(v_1^{(1)}\mathbf{e_1} + v_2^{(1)}\mathbf{e_2} + v_3^{(1)}\mathbf{e_3}\right) + \cdots \\ +\tilde{u}_2&\left(v_1^{(2)}\mathbf{e_1} + v_2^{(2)}\mathbf{e_2} + v_3^{(2)}\mathbf{e_3}\right) + \cdots \\ +\tilde{u}_3&\left(v_1^{(3)}\mathbf{e_1} + v_2^{(3)}\mathbf{e_2} + v_3^{(3)}\mathbf{e_3}\right) + \cdots\end{aligned}$$

$$\Rightarrow$$

$$\begin{aligned}\mathbf{u} = \mathbf{e_1}&\overbrace{\left(\tilde{u}_1 v_1^{(1)} + \tilde{u}_2 v_1^{(2)} + \tilde{u}_3 v_1^{(3)}\right)}^{u_1} + \cdots \\ \mathbf{e_2}&\overbrace{\left(\tilde{u}_1 v_2^{(1)} + \tilde{u}_2 v_2^{(2)} + \tilde{u}_3 v_2^{(3)}\right)}^{u_2} + \cdots \\ \mathbf{e_3}&\underbrace{\left(\tilde{u}_1 v_3^{(1)} + \tilde{u}_2 v_3^{(2)} + \tilde{u}_3 v_3^{(3)}\right)}_{u_3} + \cdots\end{aligned}$$

Eigenvalues, eigenvectors
change of basis (coordinate system)

Some discussion about egienvalues and eigenvectors:

$$\begin{array}{cccc} & \boldsymbol{v}^{(1)} & \boldsymbol{v}^{(2)} & \boldsymbol{v}^{(3)} \\ & \downarrow & \downarrow & \downarrow \end{array}$$

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} v_1^{(1)} & v_1^{(2)} & v_1^{(3)} \\ v_2^{(1)} & v_2^{(2)} & v_2^{(3)} \\ v_3^{(1)} & v_3^{(2)} & v_3^{(3)} \end{bmatrix} \begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \end{bmatrix} \Leftarrow \begin{aligned} \mathbf{u} = {} & \mathbf{e}_1 \overbrace{\left(\tilde{u}_1 v_1^{(1)} + \tilde{u}_2 v_1^{(2)} + \tilde{u}_3 v_1^{(3)}\right)}^{u_1} + \cdots \\ & \mathbf{e}_2 \overbrace{\left(\tilde{u}_1 v_2^{(1)} + \tilde{u}_2 v_2^{(2)} + \tilde{u}_3 v_2^{(3)}\right)}^{u_2} + \cdots \\ & \mathbf{e}_3 \underbrace{\left(\tilde{u}_1 v_3^{(1)} + \tilde{u}_2 v_3^{(2)} + \tilde{u}_3 v_3^{(3)}\right)}_{u_3} + \cdots \end{aligned}$$

$$\Downarrow$$

$$\mathbf{u} = \mathbf{V}\tilde{\mathbf{u}} \Rightarrow \tilde{\mathbf{u}} = \mathbf{V}^{-1}\mathbf{u}$$

Let's consider now the respresentation of an operator in the new basis:

$$\mathbf{A}\mathbf{u} = \mathbf{b}$$

$$\mathbf{A}\mathbf{V}\tilde{\mathbf{u}} = \mathbf{V}\tilde{\mathbf{b}}$$

$$\mathbf{V}^{-1}\mathbf{A}\mathbf{V}\tilde{\mathbf{u}} = \mathbf{V}^{-1}\mathbf{V}\tilde{\mathbf{b}}$$

$\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$ is the representationa of the $\mathbf{A}$ operator in the new basis defined by $\boldsymbol{v}^{(1)}$, $\boldsymbol{v}^{(2)}$, $\boldsymbol{v}^{(3)}$

$$\overbrace{\mathbf{V}^{-1}\mathbf{A}\mathbf{V}}\tilde{\mathbf{u}} = \tilde{\mathbf{b}}$$

Numerical solution of the heat equation
2.) Euler forward – d-tour on eigenvalues and eigenvectors

$\mathbf{V}^{-1}\mathbf{A}\mathbf{V}$ is the representationa of the $\mathbf{A}$ operator in the new basis defined by $\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \boldsymbol{v}^{(3)}$

If $\boldsymbol{v}^{(1)}, \boldsymbol{v}^{(2)}, \boldsymbol{v}^{(3)}$ are the eigenvectors of $\mathbf{A}$, this representation is a diagonal matrix.

Let's go back to the problem of the Euler forward method used for solving the heat equation

Stable? (does it give decaying solution?)

$$\mathbf{u}_{\mathrm{n+1}} = \mathbf{B}\mathbf{u}_{\mathrm{n}} \qquad \mathbf{u}_{\mathrm{n+2}} = \mathbf{B}^2\mathbf{u}_{\mathrm{n}} \qquad \ldots \qquad \mathbf{u}_{\mathrm{n+k}} = \mathbf{B}^k\mathbf{u}_{\mathrm{n}}$$

Let's check instead in the basis defined by the eigenvectors: $\widetilde{\mathbf{u}}_{\mathrm{i}} = \mathbf{V}^{-1}\boldsymbol{u}_{\mathrm{i}}$

$$\widetilde{\mathbf{u}}_{\mathrm{n+1}} = \mathbf{V}^{-1}\mathbf{B}\mathbf{V}\widetilde{\mathbf{u}}_{\mathrm{n}} \qquad \widetilde{\mathbf{u}}_{\mathrm{n+2}} = \mathbf{V}^{-1}\mathbf{B}^2\boldsymbol{V}\widetilde{\mathbf{u}}_{\mathrm{n}} \qquad \widetilde{\mathbf{u}}_{\mathrm{n+k}} = \mathbf{V}^{-1}\mathbf{B}^k\boldsymbol{V}\widetilde{\mathbf{u}}_{\mathrm{n}}$$

$$\mathbf{V}^{-1}\mathbf{B}^k\boldsymbol{V} = \mathbf{V}^{-1}\boldsymbol{B}\boldsymbol{B}\ldots\boldsymbol{B}\boldsymbol{B}\boldsymbol{V} = \mathbf{V}^{-1}\boldsymbol{B}\boldsymbol{V}\mathbf{V}^{-1}\boldsymbol{B}\boldsymbol{V}\mathbf{V}^{-1}\ldots\boldsymbol{V}\underbrace{\mathbf{V}^{-1}\boldsymbol{B}\boldsymbol{V}}_{\boldsymbol{D}}\underbrace{\mathbf{V}^{-1}\boldsymbol{B}\boldsymbol{V}}_{\boldsymbol{D}} = \boldsymbol{D}^k = \begin{bmatrix} \lambda_1^k & & & \\ & \lambda_2^k & & \\ & & \ddots & \\ & & & \lambda_n^k \end{bmatrix}$$

Numerical solution of the heat equation
2.) Euler forward – stability analysis

Stable? (does it give decaying solution?)

$$B = \frac{-\Delta t \cdot \beta^2}{h^2} \begin{bmatrix} 2 & -1 & & & & \\ -1 & 2 & -1 & & & \\ & -1 & 2 & & & \\ & & & 2 & -1 & \\ & & & -1 & 2 & -1 \\ & & & & -1 & 2 \end{bmatrix} + \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & 1 & & \\ & & & & 1 & \\ & & & & & 1 \end{bmatrix}$$

When for all the eigenvalues $|\lambda_i| < 1$ then response is decaying with time.

$$a = 1 - 2\frac{\Delta t \cdot \beta^2}{h^2} \qquad b = \frac{\Delta t \cdot \beta^2}{h^2}$$

$$\lambda_i = a + 2b\cos\left(\frac{i\pi}{N}\right) = 1 - 2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right)$$

$$\left|1 - 2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right)\right| < 1$$

$$i = 1..N-1$$

Numerical solution of the heat equation
2.) Euler forward – stability analysis

Stable? (does it give decaying solution?)

When for all the eigenvalues $|\lambda_i| < 1$ then response is decaying with time.

$$\left|1 - 2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right)\right| < 1$$

$$i = 1..N-1$$

$$2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right) < 1$$

allways satisfied

$$\underbrace{1 - 2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right)}_{<1} < 1$$

$$2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right) > 1$$

0< ≻ 2
Max value: 2

$$\underbrace{1 - 2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right)}_{>1} > -1$$

$$2\frac{\Delta t \beta^2}{h^2}\left(1 - \cos\frac{i\pi}{N}\right) < 2$$

$$\left(1 - \cos\frac{i\pi}{N}\right) < \frac{h^2}{\Delta t \beta^2}$$

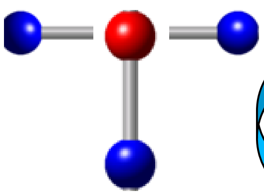
$$2 < \frac{h^2}{\Delta t \beta^2}$$

$$\boxed{\Delta t < \frac{h^2}{2\beta^2}}$$

Method is not unconditionally stable, only stable if criteria is satisfied!

Numerical solution of the heat equation
2.) Euler forward – summary

Euler forward: approximate time derivate with the forward difference



$$\frac{\partial u_j(t)}{\partial t} = \frac{\beta^2}{h^2}\left(u_{j-1} - 2u_j + u_{j+1}\right) + O(h^2) \qquad \frac{d}{dt}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t)$$

$$\boxed{\dot{u}_{j,n} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t)} \qquad \boxed{\dot{\boldsymbol{u}}_n = \frac{\boldsymbol{u}_{n+1} - \boldsymbol{u}_n}{\Delta t} = \mathbf{A}\mathbf{u}_n}$$

$$\frac{u_{j,n+1} - u_{j,n}}{\Delta t} = \frac{\beta^2}{h^2}\left(u_{j-1,n} - 2u_{j,n} + u_{j+1,n}\right) + O(h^2) + O(\Delta t)$$

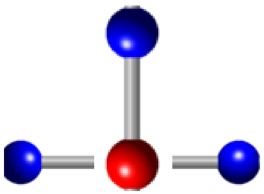
$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t\,\mathbf{A}\mathbf{u}_n = \underbrace{(\mathbf{I} + \Delta t\,\mathbf{A})}_{\mathbf{B}}\mathbf{u}_n$$

Stable? (does it give decaying solution?)
the absolut values of the eigenvalues of matrix **B** can not be greater than one.

$$\lambda_j = 1 - 2\frac{\beta^2\Delta t}{h^2}\left(1 - \cos\frac{(N-1)\pi}{N}\right) \qquad |\lambda_j| \le 1 \Longrightarrow \Delta t < \frac{h^2}{2\beta^2}$$

Numerical solution of the heat equation
3.) Euler backward (implicit) – derivation

Euler backward: approximate time derivate with the backward difference



$$\frac{\partial u_j(t)}{\partial t} = \frac{\beta^2}{h^2}\left(u_{j-1} - 2u_j + u_{j+1}\right) + O(h^2)$$

$$\frac{d}{dt}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t)$$

$$\boxed{\dot{u}_{j,n+1} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t)}$$

$$\boxed{\dot{\mathbf{u}}_{n+1} = \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{\Delta t} = \mathbf{A}\mathbf{u}_{n+1}}$$

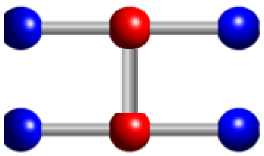
$$\frac{u_{j,n+1} - u_{j,n}}{\Delta t} = \frac{\beta^2}{h^2}\left(u_{j-1,n+1} - 2u_{j,n+1} + u_{j+1,n+1}\right) + O(h^2) + O(\Delta t)$$

$$\mathbf{u}_n = \mathbf{u}_{n+1} - \Delta t\mathbf{A}\mathbf{u}_{n+1} = (\mathrm{I} - \Delta t\mathbf{A})\mathbf{u}_{n+1}$$

(Stability criteria: unconditionaly stable, to be shown later)

Numerical solution of the heat equation
4.) Theta method (implicit) – derivation

Theta method (Crank-Nicolson) $\frac{\partial u_j(t)}{\partial t} = \frac{\beta^2}{h^2}\left(u_{j-1} - 2u_j + u_{j+1}\right) + O(h^2)$ $\quad \frac{d}{dt}\mathbf{u}(t) = \mathbf{A}\mathbf{u}(t)$



$$\boxed{\dot{u}_{j,n+\theta} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t^p)}$$

$$\dot{u}_{j,n} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t)$$

Euler f.

$$\dot{\boldsymbol{u}}_n = \frac{\boldsymbol{u}_{n+1} - \boldsymbol{u}_n}{\Delta t} = \mathbf{A}\mathbf{u}_n$$

$$\dot{u}_{j,n+1} = \frac{u_{j,n+1} - u_{j,n}}{\Delta t} + O(\Delta t)$$

Euler b.

$$\dot{\mathbf{u}}_{n+1} = \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{\Delta t} = \mathbf{A}\mathbf{u}_{n+1}$$

$$\frac{u_{j,n+1} - u_{j,n}}{\Delta t} = \theta \frac{\beta^2}{h^2}\left(u_{j-1,n+1} - 2u_{j,n+1} + u_{j+1,n+1}\right) + $$
$$+(1-\theta)\frac{\beta^2}{h^2}\left(u_{j-1,n} - 2u_{j,n} + u_{j+1,n}\right) + O(h^2) + O(\Delta t^p)$$

$$\boxed{\dot{\mathbf{u}}_{n+\theta} = \frac{\mathbf{u}_{n+1} - \mathbf{u}_n}{\Delta t} = \theta\mathbf{A}\mathbf{u}_{n+1} + (1-\theta)\mathbf{A}\mathbf{u}_n}$$

$$\underbrace{(\mathbf{I} - \theta\Delta t\mathbf{A})}_{\mathbf{B}_1}\mathbf{u}_{n+1} = \underbrace{(\mathbf{I} + (1-\theta)\Delta t\mathbf{A})}_{\mathbf{B}_2}\mathbf{u}_n$$

$\theta = 0$ Euler forward

$\theta = 1$ Euler backward

Numerical solution of the heat equation
4.) Theta method (implicit) – stability analysis

Stability criteria:

$$\mathbf{B_1} = (\mathbf{I} - \theta\Delta t\mathbf{A})$$

$$\mathbf{B_1}\mathbf{u}_{\mathrm{n+1}} = \mathbf{B_2}\mathbf{u}_{\mathrm{n}}$$

$$\mathbf{B_2} = (\mathbf{I} + (1-\theta)\Delta t\mathbf{A})$$

1) $\mathbf{B_1}$ and $\mathbf{B_2}$ are both tridiagonal sym. matrices

➡ $\mathbf{B_1}$ and $\mathbf{B_2}$ have the same eigenvectors

$$\boldsymbol{v_i} = \begin{bmatrix} v_i^1 \\ \vdots \\ v_i^{N-1} \end{bmatrix} \qquad v_i^k = \sin\left(\frac{ik\pi}{N}\right)$$

2) The representation in the basis defined by the eigenvectors

$$\underbrace{\mathbf{V}^{-1}\mathbf{B_1}\boldsymbol{V}}_{\mathbf{D_1}}\,\tilde{\mathbf{u}}_{\mathrm{n+1}} = \underbrace{\mathbf{V}^{-1}\mathbf{B_2}\boldsymbol{V}}_{\mathbf{D_2}}\tilde{\mathbf{u}}_{\mathrm{n}}$$

$$\begin{bmatrix} \lambda_1(\mathbf{B_1}) & & & \\ & \lambda_2(\mathbf{B_1}) & & \\ & & \ddots & \\ & & & \lambda_n(\mathbf{B_1}) \end{bmatrix} \tilde{\mathbf{u}}_{\mathrm{n+1}} = \begin{bmatrix} \lambda_1(\mathbf{B_2}) & & & \\ & \lambda_2(\mathbf{B_2}) & & \\ & & \ddots & \\ & & & \lambda_n(\mathbf{B_2}) \end{bmatrix} \tilde{\mathbf{u}}_{\mathrm{n}}$$

Numerical solution of the heat equation
4.) Theta method (implicit) – stability analysis

Stability criteria:

$$\mathbf{B_1} \mathbf{u}_{\mathrm{n+1}} = \mathbf{B_2} \mathbf{u}_{\mathrm{n}}$$

$$\mathbf{B_1} = (\mathbf{I} - \theta \Delta t \mathbf{A})$$

$$\mathbf{B_1} = \begin{bmatrix} a_1 & b_1 & & \\ b_1 & a_1 & b_1 & \\ & b_1 & a_1 & \\ & & & \ddots \end{bmatrix}$$

$$\tilde{\mathbf{u}}_{\mathrm{n+1}} = \begin{bmatrix} \frac{\lambda_1(\mathbf{B_2})}{\lambda_1(\mathbf{B_1})} & & & \\ & \frac{\lambda_2(\mathbf{B_2})}{\lambda_2(\mathbf{B_1})} & & \\ & & \ddots & \\ & & & \frac{\lambda_n(\mathbf{B_2})}{\lambda_n(\mathbf{B_1})} \end{bmatrix} \tilde{\mathbf{u}}_{\mathrm{n}}$$

$$\mathbf{B_2} = (\mathbf{I} + (1 - \theta) \Delta t \mathbf{A})$$

$$\mathbf{B_2} = \begin{bmatrix} a_2 & b_2 & & \\ b_2 & a_2 & b_2 & \\ & b_2 & a_2 & \\ & & & \ddots \end{bmatrix}$$

$$\lambda_i(\mathbf{B_1}) = a_1 + 2b_1 \cos\left(\frac{i\pi}{N}\right) = 1 + \frac{2\beta^2 \Delta t}{h^2} \theta \left(1 - \cos\left(\frac{i\pi}{N}\right)\right)$$

$$\lambda_i(\mathbf{B_2}) = a_2 + 2b_2 \cos\left(\frac{i\pi}{N}\right) = 1 - \frac{2\beta^2 \Delta t}{h^2} (1 - \theta) \left(1 - \cos\left(\frac{i\pi}{N}\right)\right)$$

Numerical solution of the heat equation
4.) Theta method (implicit) – stability analysis

$$\tilde{\mathbf{u}}_{\mathrm{n+1}} = \begin{bmatrix} \frac{\lambda_1(\mathbf{B_2})}{\lambda_1(\mathbf{B_1})} & & & \\ & \frac{\lambda_2(\mathbf{B_2})}{\lambda_2(\mathbf{B_1})} & & \\ & & \ddots & \\ & & & \frac{\lambda_n(\mathbf{B_2})}{\lambda_n(\mathbf{B_1})} \end{bmatrix} \tilde{\mathbf{u}}_{\mathrm{n}}$$

$$\left|\frac{\lambda_i(\mathbf{B_2})}{\lambda_i(\mathbf{B_1})}\right| = \left|\frac{1 - \overbrace{\frac{2\beta^2\Delta t}{h^2}}^{2r}(1-\theta)\overbrace{\left(1-\cos\left(\frac{i\pi}{N}\right)\right)}^{0< C_i > 2}}{1 + \underbrace{\frac{2\beta^2\Delta t}{h^2}}_{2r}\theta\underbrace{\left(1-\cos\left(\frac{i\pi}{N}\right)\right)}_{0< C_i > 2}}\right| < 1$$

$$r := \frac{\beta^2\Delta t}{h^2}$$

$$C_i := \left(1-\cos\left(\frac{i\pi}{N}\right)\right)$$

$$-1 < \frac{1-2r(1-\theta)C_i}{1+2r\theta C_i} < 1$$

Numerical solution of the heat equation
4.) Theta method (implicit) – stability analysis

$$r := \frac{\beta^2 \Delta t}{h^2}$$

1) Right hand side of the equation:

$$\boxed{-1 < \frac{1-2r(1-\theta)C_i}{\underbrace{1+2r\theta C_i}_{>0}}}$$

$$-1 - 2r\theta C_i < 1 - 2r(1-\theta)C_i \qquad /-1$$

$$-2 - 2r\theta C_i < -2rC_i + 2r\theta C_i \qquad /+2r\theta C_i$$

$$-2 < -2rC_i + 4r\theta C_i \qquad /:(-2)$$

$$1 > r(1-2\theta)C_i$$

$$\boxed{1 - 2\theta \leq 0 \quad (\theta \geq 0.5)}$$

$$1 > r\underbrace{(1-2\theta)C_i}_{<0}$$

unconditionally satisfied

$$\boxed{1 - 2\theta > 0 \quad (\theta < 0.5)}$$

$$\frac{1}{C_i(1-2\theta)} > r \qquad \frac{h^2}{\beta^2 \underbrace{C_i}_{\max 2}\underbrace{(1-2\theta)}_{\max 1}} > \Delta t \qquad \boxed{\frac{h^2}{2\beta^2(1-2\theta)} > \Delta t}$$

2) Left hand side of the equation:

$$\boxed{\frac{1-2r(1-\theta)C_i}{\underbrace{1+2r\theta C_i}_{>0}} < 1}$$

$$1 - 2r(1-\theta)C_i < 1 + 2r\theta C_i$$

$$2rC_i > 0$$

$$\Downarrow$$

unconditionally satisfied

Summary:
Instationary heat equation – what to solve?

Stability checking from eigenvalue analysis:

- Method of lines

$$\mathbf{u}(t) = \sum_{j=1}^{N-1} \beta_j(t)\mathbf{v}_j \quad \beta_j(t) = \beta_j^0 e^{\lambda_j t}$$

$\Rightarrow$ find the eigenvalues (λ_j) and eigenvectors ($\mathbf{v}_j$) of matrix **A**

- Euler forward method

$$\mathbf{u}_{n+1} = \underbrace{(\mathrm{I} + \Delta t\mathbf{A})}_{\mathbf{B}}\,\mathbf{u}_n$$

$\Rightarrow$ $\mathbf{u}_{n+1} = \mathbf{B}\mathbf{u}_n$

- Euler backward method

$$\mathbf{u}_n = \underbrace{(\mathrm{I} - \Delta t\mathbf{A})}_{\mathbf{B}_1}\,\mathbf{u}_{n+1}$$

$\Rightarrow$ $\mathbf{B}_1\mathbf{u}_{n+1} = \mathbf{u}_n$

- Theta method

$$\underbrace{(\mathbf{I} - \theta\Delta t\mathbf{A})}_{\mathbf{B}_{1\theta}}\,\mathbf{u}_{n+1} = \underbrace{(\mathbf{I} + (1-\theta)\Delta t\mathbf{A})}_{\mathbf{B}_{2\theta}}\,\mathbf{u}_n$$

$\Rightarrow$ $\mathbf{B}_{1\theta}\mathbf{u}_{n+1} = \mathbf{B}_{2\theta}\mathbf{u}_n$

Solve system of equations for $\mathbf{u}_{n+1}$