MM04 Computational and Simulation Methods Notes

Based on the 2011 autumn lectures by Dr S N Timoshin and Dr G van der Heijden

MM04

Computational & Simulation Methods

PART 1 Mostly FE (finite element) method

PART 2 Other numerical (integration) methods

PART I

THE FINITE ELEMENT METHOD

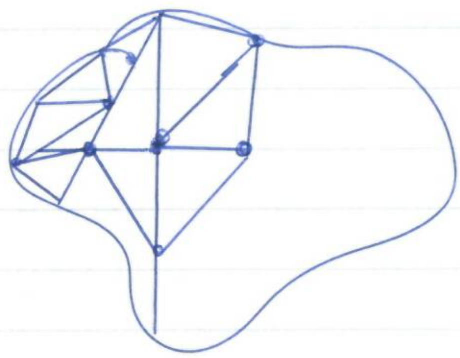
The finite element method is a method for solving differential eq^{n}s (ODEs or PDEs) approximately.

The complication in differential eq^{n}s is not only the eq^n itself but also the boundary conditions.

The strategy of FE is the following 3-step procedure

(1) Divide the domain up into small regular parts ('finite elements') – this also approximates the boundary. Boundary conditions are more easily applied to the elements than on the entire domain.

(2) Choose points (nodes) where the solution will be computed (discretisation step: finite n^o of degrees of freedom).



(3) Use interpolation to construct solution on full domain to be done s.t. continuity of the sol^n is guaranteed across element boundaries.

Thus the FE method has the following desirable properties:

(1) It's universal and robust – it applies to any DE and BCs

(2) Can be successively refined (smaller elements, more nodes) to get any required precision. ($\Rightarrow$ convergence to exact solution of the continuum problem.) ("usually OK").

(3) Can to a very large extent use the Power of a Computer, both for the solution of the final system of eq^{n}s, and for the formulation of the approximation (steps 1 to 3)

1D Approximation Theory

Approximation techniques based on weighted residuals.

E.g.// $$\frac{d^2u}{dx^2} + u + x = 0 \qquad x \in [0,1]$$

b.c. $u(0) = 0 = u(1)$ (Dirichlet b.c. on u)

b.c. on u' called Neumann b.c.

- Take trial sol^n that satisfies the b.c.

$$\tilde{u}(x) = ax(1-x)$$

The error, or residual, we're making is

$$R(x) = \frac{d^2\tilde{u}}{dx^2} + \tilde{u} + x$$

$$= -2a + ax(1-x) + x \neq 0$$

free parameter a

We can adjust a to get "the best" sol^n according to some criterion.

But which criterion to take?

The method of weighted residuals defines a weight f^n w and asks that the weighted average

$$I = \int_0^1 w R \, dx = 0.$$

Three popular choices for w:

(1) Collocation method: $\boxed{w(x) = \underset{\text{Dirac } \delta}{\delta}(x - x_i)}$ $\quad x_i \in (0,1)$

Here, $I = \int_0^1 \delta(x - x_i) R(x) \, dx = R(x_i)$

$$= -2a + a x_i (1 - x_i) + x_i := 0$$

Take $x_i = \frac{1}{2}$. Find $I = -2a + \frac{1}{4}a + \frac{1}{2} := 0$

$$\Rightarrow a = \frac{2}{7} \approx \underline{0.2857}$$

(2) Least squares method:

Minimise $J = \int_0^1 R^2 \, dx$

Here $J = \frac{101}{30} a^2 - \frac{11}{6} a + \frac{1}{3}$

To minimise, $\frac{dJ}{da} = 0 \Rightarrow a = \frac{55}{202} \approx \underline{0.2723}$

Note that $\frac{dJ}{da} = 2 \int_0^1 \frac{dR}{da} R \, dx := 0$

ie this is of the form of a weighted residual where

$$\boxed{w(x) = \frac{dR}{da}}$$

(3) <u>Galerkin method</u>

$$\boxed{w(x) = \tilde{u}(x)}$$

Choose the trial function itself as the weight f^n:

$$w(x) = x(1-x). \qquad \left(\begin{matrix}\text{factor a} \\ \text{irrelevant}\end{matrix}\right)$$

Here $I = \int_0^1 x(1-x)\left[-2a + ax(1-x) + x\right] dx$

$$= -\frac{3}{10}a + \frac{1}{12} = 0$$

$$\Rightarrow a = \frac{5}{18} \approx \underline{0.2778}$$

← To improve the approximation we can take more terms with more free parameters.

For instance, $x^2(1-x)$ also satisfies the BCs so we can take the following trial function

$$\underline{\tilde{u} = a_1 x(1-x) + a_2 x^2(1-x)}$$

Residual $R = \frac{d^2\tilde{u}}{dx^2} + \tilde{u} + x = -2a_1 + 2a_2 - 6a_2 x + a_1 x(1-x) + a_2 x^2(1-x) + x$

Now need two weight f^{n}s.

(1) Collocation: $w_1 = \delta(x - x_1)$ $\quad x_{1,2} \in (0,1)$

$w_2 = \delta(x - x_2)$

(2) Least squares: $w_1 = \frac{dR}{da_1} \Rightarrow a_1 = 0.1875$

$w_2 = \frac{dR}{da_2} \Rightarrow a_2 = 0.1695$

(3) Galerkin: $w_1 = x(1-x)$

$w_2 = x^2(1-x)$

$$\int_0^1 x(1-x)R(x)\,dx := 0 \Rightarrow a_1 = 0.1924$$

and $\int_0^1 x^2(1-x)R(x)\,dx := 0 \Rightarrow a_2 = 0.1707$

We could continue with a whole series of trial functions. For instance we could have

$$\tilde{u}_n = a_n \sin(n\pi x) \quad \text{Fourier basis}$$

The exact sol^n in this case is easily found, of course.

$$u'' + u = -x, \qquad u(0) = 0 = u(1)$$

By inspection, $u(x) = \frac{\sin x}{\sin 1} - x$.

By taking the Fourier basis we'd be computing the Fourier series of this solution

$$u(x) = \sum_{n=1}^{\infty} \tilde{u}_n$$

$$= \sum_{n=1}^{\infty} a_n \sin n\pi x$$

In more general form:

$$Lu = f$$ (L is a linear operator. In previous example $Lu = u'' + u$)

Trial function φ_i, approx. $\tilde{u} = \sum_{i=1}^{n} a_i \varphi_i$

Residual $R = L\tilde{u} - f = \sum_{i=1}^{n} a_i L\varphi_i - f$

Weighted residual: $\int_0^1 w_j R \, dx = 0 \qquad (j = 1, \dots, n)$

Least squares: $w_j = \dfrac{\partial R}{\partial a_j} = L\varphi_j$

$$\Rightarrow 0 = \int_0^1 L\varphi_j R \, dx$$

$$= \int_0^1 L\varphi_j \left(\sum_{i=1}^{n} a_i L\varphi_i - f \right) dx$$

$$\Leftrightarrow \sum_{i=1}^{n} A_{ji} a_i = f_j$$

where $A_{ji} = \int_0^1 L\varphi_j L\varphi_i \, dx$

$$\tilde{\underline{f}} = \begin{pmatrix} f_1 \\ \vdots \\ f_n \end{pmatrix} \qquad f_j = \int_0^1 f L\varphi_j \, dx$$

ie $\mathbf{A}\underline{a} = \tilde{\underline{f}}$

Note: matrix $\mathbf{A}$ is symmetric

Galerkin : $w_j = \varphi_j$

$$0 = \int_0^1 \varphi_j R \, dx = \sum_{i=1}^{n} a_i \int_0^1 \varphi_j L\varphi_i \, dx - \int_0^1 f \varphi_j \, dx = 0$$

So $\mathbf{A}\underline{a} = \underline{\tilde{f}}$

w now $A_{ij} = \int_0^1 \varphi_i L \varphi_j \, dx$ not nec. symmetric

$$f_i = \int_0^1 f \varphi_i \, dx$$

Recall that that the adjoint L^* of the linear operator L is defined by

$$\int_0^1 f L g \, dx = \int_0^1 L^* f g \, dx$$

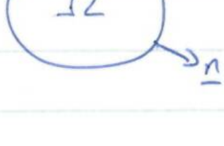
L is called self-adjoint if $L^* = L$

Conclusion: $\mathbf{A}$ is symmetric if L is self-adjoint.

2D Approximation Theory

Gauss' Divergence Thm:

$$\iint_{\Omega} \operatorname{div}(\underline{v}) \, dx\,dy = \oint_{\partial\Omega} \underline{v} \cdot \underline{n} \, dS$$



If $\underline{v} = v_x \underline{\hat{x}} + v_y \underline{\hat{y}}$

$$\nabla \cdot \underline{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}$$

div v > 0

div v < 0

div v = 0

"Sum of sources in Ω = flow through boundary $\partial\Omega$."

Integration by parts in 2D

Now change

~~Take~~ $v \to \varphi v$ (φ scalar $f^{\underline{n}}$)

$$\nabla\cdot(\varphi \underline{v}) \quad \operatorname{div}(\varphi v) = \frac{\partial}{\partial x}(\varphi v_x) + \frac{\partial}{\partial y}(\varphi v_y)$$

$$= \varphi \frac{\partial v_x}{\partial x} + \frac{\partial \varphi}{\partial x} v_x + \varphi \frac{\partial v_y}{\partial y} + \frac{\partial \varphi}{\partial y} v_y$$

$$= \varphi \nabla\cdot\underline{v} + \nabla\varphi\cdot\underline{v}$$

So now taking the divergence thm,

$$\iint_\Omega \varphi \nabla\cdot\underline{v}\, dxdy = -\iint_\Omega \nabla\varphi\cdot\underline{v}\, dxdy + \oint_{\partial\Omega} \varphi\, \underline{v}\cdot\underline{n}\, dS \qquad \text{(Green's thm)}$$

Now take $v_x = \psi$, $v_y = 0$ (holds $\forall v$)

$$\iint_\Omega \varphi \frac{\partial \psi}{\partial x}\, dxdy = -\iint_\Omega \frac{\partial \varphi}{\partial x}\psi\, dxdy + \oint_{\partial\Omega} \varphi\psi n_x\, dx \qquad \text{(Gauss-Green)}$$

"integration by parts"

($\underline{v}\cdot\underline{n} = v_x n_x + v_y n_y = \psi n_x$) above

In 1D:

$n_x(a) = -1$
$n_x(b) = 1$

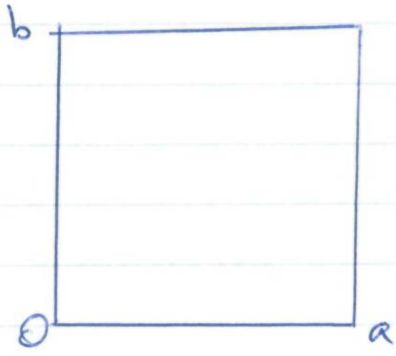
then $$\int_a^b \varphi \frac{d\psi}{dx}\, dx = -\int_a^b \frac{d\varphi}{dx} \psi \, dx + \underbrace{\varphi(a)\psi(a)n_x(a) + \varphi(b)\psi(b)n_x(b)}_{\text{sum over bdy}}$$

$$\parallel$$

$$\varphi(b)\psi(b) - \varphi(a)\psi(a)$$

$$= \left[\psi\varphi\right]_a^b$$

giving the familiar integration by parts formula in 1D.



Example

Poisson's eq^n on a rectangle

$$\nabla^2 u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = -c$$

b.c. $u = 0$ at $x=0$, $x=a$, $y=0$, $y=b$ (ie. on bdy).

Trial sol^n: $\tilde{u}(x,y) = \alpha x y (x-a)(y-b)$

Operator $L = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ $\quad (Lu = f)$

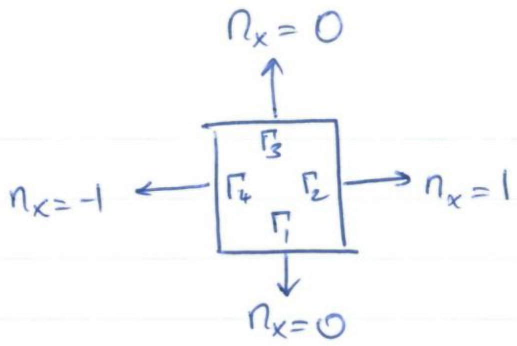
$f = -c$

Proof that L is self-adjoint:

$$\iint_\Omega \varphi \frac{\partial^2 \psi}{\partial x^2}\, dx\, dy = -\iint_\Omega \frac{\partial \varphi}{\partial x}\frac{\partial \psi}{\partial x}\, dx\, dy + \int_{\partial\Omega} \varphi \frac{\partial \psi}{\partial x} n_x \, dS$$

(Gauss-Green)

Boundary term:



$$\oint_{\partial\Omega} \varphi \frac{\partial \psi}{\partial x} n_x \, dx$$

$$= \int_{\Gamma_1} 0 \, ds + \int_{\Gamma_2} \varphi \frac{\partial \psi}{\partial x} \, dS + \int_{\Gamma_3} 0 \, dS - \int_{\Gamma_4} \varphi \frac{\partial \psi}{\partial x} \, dS$$

Actually $= 0$ because $\varphi = 0$ on bdy

$$\Rightarrow \iint_{\Omega} \varphi \frac{\partial^2 \psi}{\partial x^2} \, dx dy = - \iint_{\Omega} \frac{\partial \varphi}{\partial x} \frac{\partial \psi}{\partial x} \, dx dy$$

$$\overset{\text{Gauss-Green}}{=} \iint_{\Omega} \frac{\partial^2 \varphi}{\partial x^2} \psi \, dx dy - \oint_{\partial\Omega} \underbrace{\frac{\partial \varphi}{\partial x} \psi n_x \, dS}_{= 0}$$

<u>Conclusion:</u> $\iint \varphi \frac{d^2 \psi}{dx^2} \, dx dy = \int \frac{\partial^2 \varphi}{\partial x^2} \psi \, dx dy$.

Same for y-derivatives:

$$\iint \varphi \frac{\partial^2 \psi}{\partial y^2} \, dx dy = \iint \frac{\partial^2 \varphi}{\partial y^2} \psi \, dx dy$$

Hence: $\iint \varphi L \psi \, dx dy = \iint \psi L \varphi \, dx dy$.

$\Rightarrow L$ is self-adjoint.

$Aa = \bar{f}$, A is symmetric.

Take $n<1$, $Aa = \bar{f}$, $A = \int_\Omega \varphi L \varphi \, dx\,dy$

where $\varphi = xy(x-a)(y-b)$

Exercise: verify that $A = -\frac{a^3b^3}{90}(a^2+b^2)$

$$\bar{f} = c\int_0^b\int_0^a (x-a)(y-b)\,dx\,dy$$

$$= -\frac{c}{36}a^3b^3$$

Thus $\tilde{u}(x,y) = \frac{5c}{2(a^2+b^2)}xy(x-a)(y-b)$

L2 New Lect.

This formulation in terms of the diff^{l} eq^{n} $Lu=f$ is called the strong formulation. To reduce the differentiability requirements, multiply by a test f^{n} and integrate

$$\int_0^1 (u'' + u + x)v\,dx = 0$$

Int. by parts: $\int_0^1 (-u'v' + uv + xv)\,dx + \underbrace{[vu']_0^1}_{=0 \text{ (bcs)}} = 0 \quad \forall v$

on first term $(u''v)$

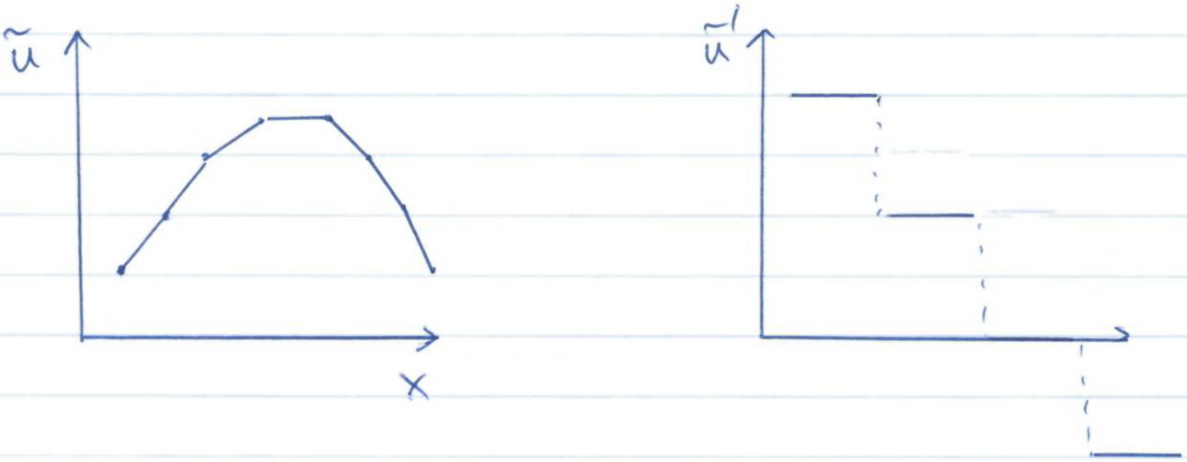
$$\begin{cases} \int_0^1 (-u'v' + uv + xv)\,dx = 0 & (\forall v) \\ u(0) = u(1) = 0 \end{cases}$$

This integral form of the problem is called the <u>Weak formulation</u>

(second derivative not req^{d}!)

For sufficiently smooth f^{n}s, the strong and weak formulations are equivalent, but the weak formulation allows for a much wider class of trial f^{n}s to approximate the real sol^n.

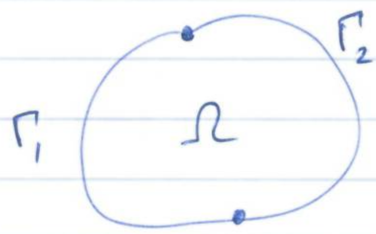
For instance, it will be convenient to use piecewise linear approximations.



u' fine but u'' does not exist

Weak formulation in 2D

Poisson eq^n: $$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = f$$



$$\text{BC} \begin{cases} u = 0 & \text{on } \Gamma_1 \\ \nabla u \cdot \underline{n} = g & \text{on } \Gamma_2 \quad (\text{given 'flux'}) \end{cases}$$

$\Gamma_1 \cup \Gamma_2 = \Gamma$

g, a f^n.

Weak formulation: $$\int_\Omega \underbrace{\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} - f \right)}\, v \, dx\,dy = 0$$

bring everything over to the LHS, then $\times v$ and integrate, set to zero.

Gauss-Green:
$$-\int_\Omega \left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\frac{\partial v}{\partial y}\right)dx\,dy$$
$$-\int_\Omega f v\,dx\,dy + \oint_\Gamma v\underbrace{\left(\frac{\partial u}{\partial x}n_x+\frac{\partial u}{\partial y}n_y\right)}_{\nabla u\cdot \underline{n}} dS = 0$$

$$\Leftrightarrow\ -\int_\Omega \left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\frac{\partial v}{\partial y}\right)dx\,dy$$
$$+\underbrace{\int_{\Gamma_1} \underset{=0}{v}\,\nabla u\cdot\underline{n}\,dS}_{=0} + \int_{\Gamma_2} v g\,dS - \int_\Omega f v\,dx\,dy = 0$$

$g = \nabla u\cdot\underline{n}$ on Γ_2

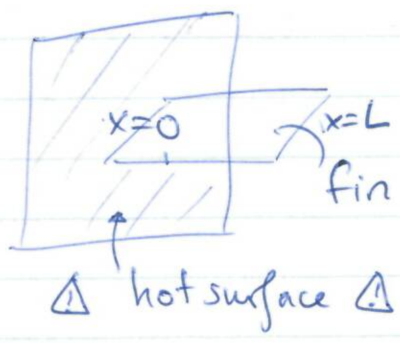
Weak formulation:

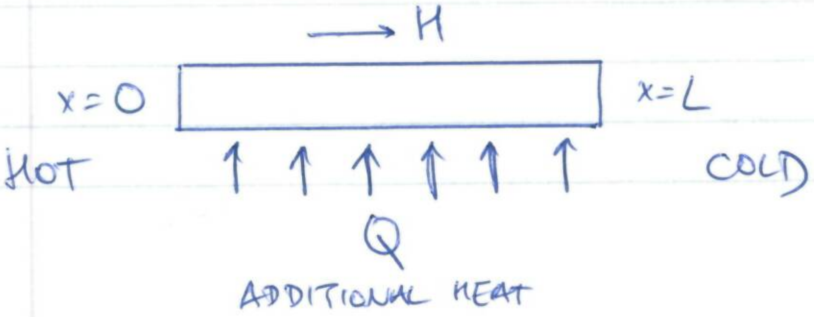
$$\begin{cases} \displaystyle\int_\Omega \left(\frac{\partial u}{\partial x}\frac{\partial v}{\partial x}+\frac{\partial u}{\partial y}\frac{\partial v}{\partial y}\right)dx\,dy + \int_\Omega f v\,dx\,dy - \int_{\Gamma_2} v g\,ds = 0 \\ u = 0 \text{ on } \Gamma_1 \qquad \text{(second b.c., } \nabla u\cdot n = g \text{ on } \Gamma_2 \text{ has been absorbed)} \end{cases}$$

for all v satisfying BCs.

Heat flow modelling in 1D

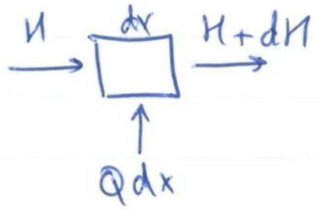
Consider heat circulation in a thin fin:





Q = heat input / unit time / unit length
H = heat flow per unit time along fin

Energy balance on infinitesimal element



$$H + Q\,dx = H + dH \quad \Rightarrow \quad \frac{dH}{dx} = Q$$

The heat flux ~~is~~ q is defined by $H(x) = A(x)\,q(x)$ where $A(x)$ is the X-sectional area.

Flux is related to temperature gradient

$$q = -k\frac{dT}{dx} \qquad k \sim \text{thermal conductivity}$$

(Fourier's law). Example of a constitutive (depends on the material) relation. Minus sign ∵ heat flows from hot to cold. Put together:

$$\underline{\frac{d}{dx}\left(Ak\frac{dT}{dx}\right) + Q = 0} \qquad \text{(1D heat eq}^{n}\text{.)}$$

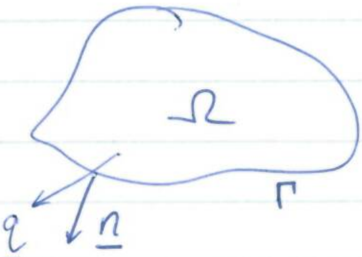
Possible b.c.s:

$$T(0) = T_0$$

$$q(L) = \left(-k\frac{dT}{dx}\right)\Big|_{x=L} = q_L \qquad (T_0,\ q_L \text{ given})$$

(e.g. if end $x = L$ insulated then no flux at $x = L$, so $q_L = 0$.)

Heat flow in 2D



Q = amount of heat supplied to the body / unit area / unit time

$\underline{q}$ = heat flux vector

$q_n = \underline{q} \cdot \underline{n}$ = flux

Energy conservation: $\int_\Omega Q \, dx\,dy = \oint_\Gamma q_n \, dS$

Gauss integration formula: $\int_\Omega Q \, dx\,dy = \int_\Omega \nabla \cdot \underline{q} \, dx\,dy$

$\Rightarrow$ each part of the region $\nabla \cdot \underline{q} = Q$

q is related to temperature gradient ∇T

$$q = -D \nabla T \qquad \text{(D conductivity matrix)}$$

Since heat flows from hot to cold,

$$\nabla T \cdot \underline{q} < 0$$

$$\Rightarrow \quad (\nabla T)^T D \nabla T > 0 \quad \forall \; \nabla T \neq 0$$

$\Rightarrow$ D is positive definite

$\Rightarrow$ $\det D \neq 0$

In practice often $D = \begin{pmatrix} k_{xx} & 0 \\ 0 & k_{yy} \end{pmatrix}$ (orthotropic material)

or $D = kI = k\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ (isotropic material)

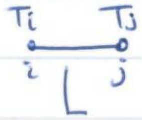
Put together (for an isotropic material)

$$\frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{Q}{k} = 0 \qquad (\text{Poisson's eq}^{\underline{n}})$$

If no source ($Q=0$), $\underbrace{\nabla^2 T = 0}$ (Laplace's eq$^{\underline{n}}$)

Possible BCs: $T = g$ on Γ_g
$q_n = n$ on Γ_n

L3 FE mesh in 1D

Consider a 2-node element 

with an unknown field variable T, e.g. temperature

Approximate: $T(x) = \alpha_1 + \alpha_2 x$ (2 parameters).

For consistency, $T(x_i) = T_i$
$T(x_j) = T_j$

$\Rightarrow$ 2 eq$^{\underline{n}}$s for 2 unknowns: $\alpha_1 + \alpha_2 x_i = T_i$
$\alpha_1 + \alpha_2 x_j = T_j$

$\Rightarrow \alpha_2 = \dfrac{T_j - T_i}{L} \qquad (L = x_j - x_i)$

$\Rightarrow \alpha_1 = T_i - \dfrac{T_j - T_i}{L} x_i$

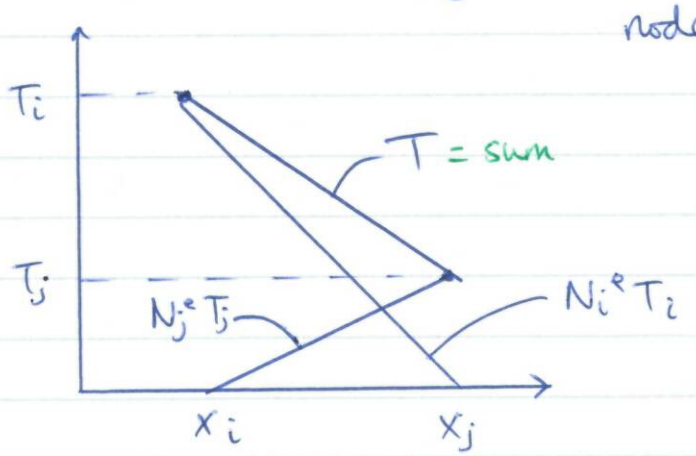
Rearrange: $T(x) = N_i^e(x) T_i + N_j^e(x) T_j$

where $N_i^e(x) = \dfrac{x - x_j}{L}$ (element-shaped f^ns.)

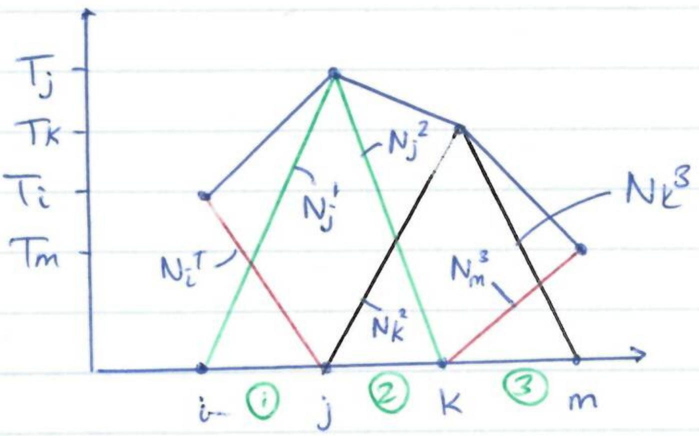
$$N_j^e(x) = \frac{x - x_i}{L}$$

Note: $N_i^e(x_i) = 1, \quad N_i^e(x_j) = 0$
$N_j^e(x_i) = 0, \quad N_j^e(x_j) = 1$

Graphically, this is simply linear interpolation between nodal values T_i, T_j

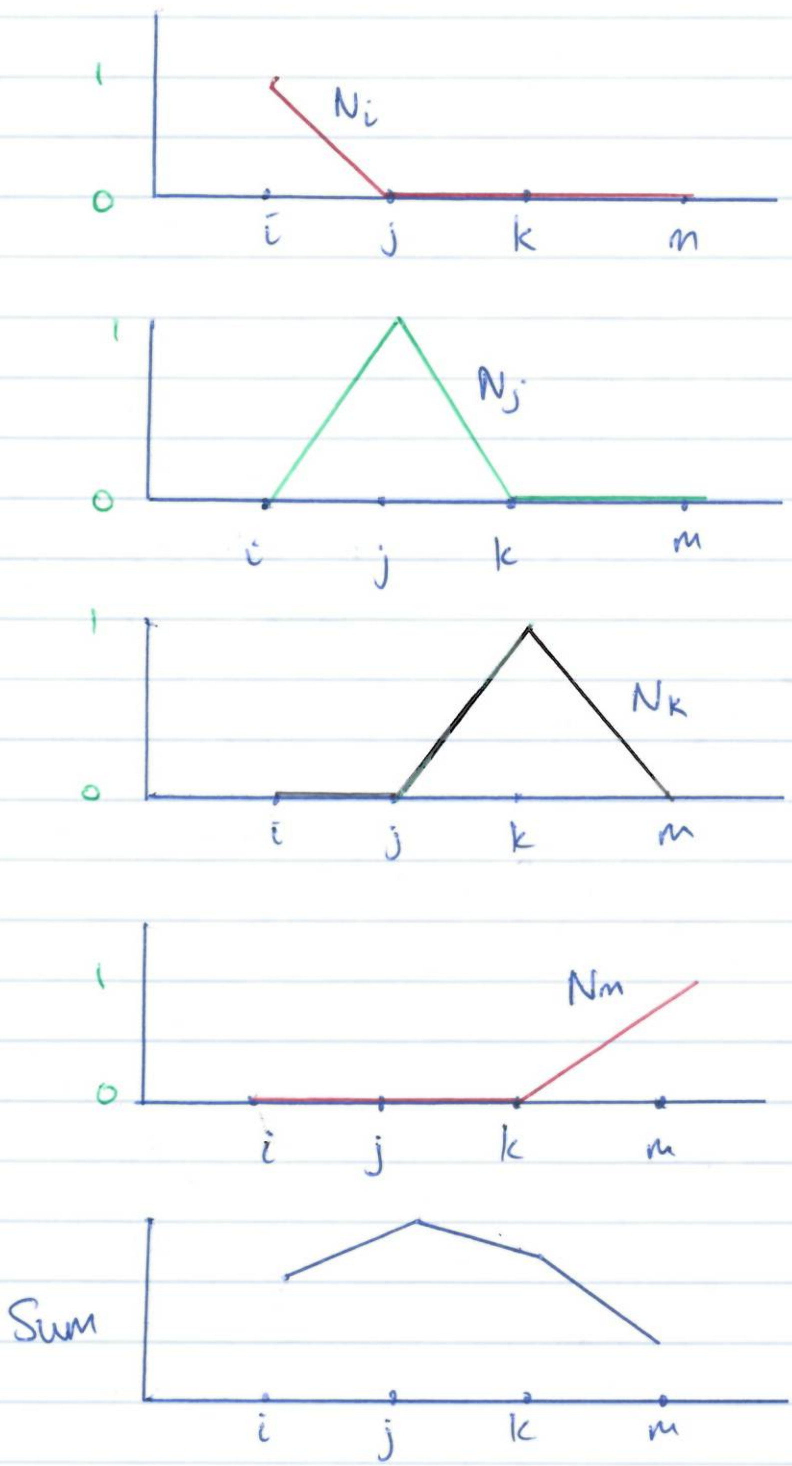


Consider 3 elements:



Introduce global shape functions defined over the the whole domain and associated to the nodes

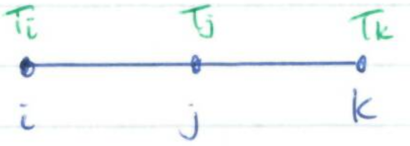
TENT F$^{\underline{n}}$S



We have $T(x) = N_i(x)T_i + N_j(x)T_j + N_k(x)T_k + N_m(x)T_m$

(in terms of global shape f$^{\underline{n}}$s)

Quadratic (3-node) element



Approximation : $T = \alpha_1 + \alpha_2 x + \alpha_3 x^2$

We require : $\begin{cases} T_i = \alpha_1 + \alpha_2 x_i + \alpha_3 x_i^2 \\ T_j = \alpha_1 + \alpha_2 x_j + \alpha_3 x_j^2 \\ T_k = \alpha_1 + \alpha_2 x_k + \alpha_3 x_k^2 \end{cases}$

ie $$\begin{pmatrix} 1 & x_i & x_i^2 \\ 1 & x_j & x_j^2 \\ 1 & x_k & x_k^2 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} T_i \\ T_j \\ T_k \end{pmatrix}$$

Solve by Cramer's rule

$$\alpha_1 = \frac{\begin{vmatrix} T_i & x_i & x_i^2 \\ T_j & x_j & x_j^2 \\ T_k & x_k & x_k^2 \end{vmatrix}}{\begin{vmatrix} 1 & x_i & x_i^2 \\ 1 & x_j & x_j^2 \\ 1 & x_k & x_k^2 \end{vmatrix}}$$

$$\alpha_2 = \frac{\begin{vmatrix} 1 & T_i & x_i^2 \\ 1 & T_j & x_j^2 \\ 1 & T_k & x_k^2 \end{vmatrix}}{|\cdot|}$$

$$\alpha_3 = \frac{\begin{vmatrix} 1 & x_i & T_i \\ 1 & x_j & T_j \\ 1 & x_k & T_k \end{vmatrix}}{|\cdot|}$$

Rearrange to write: $T(x) = N_i^e(x) T_i + N_j^e(x) T_j + N_k^e(x) T_k$

$$N_i^e(x) = \frac{2}{L^2}(x - x_j)(x - x_k)$$

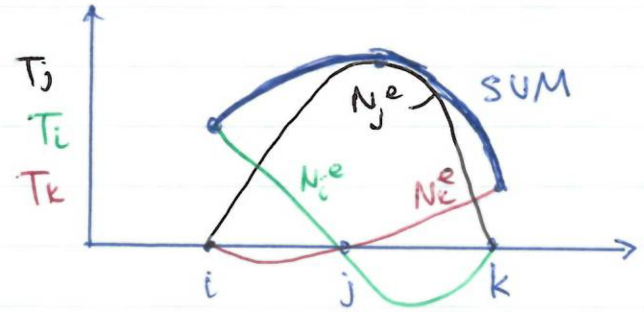
$L = x_k - x_i$

$$N_j^e(x) = -\frac{4}{L^2}(x - x_i)(x - x_k)$$

$$N_k^e(x) = \frac{2}{L^2}(x - x_k)(x - x_j)$$

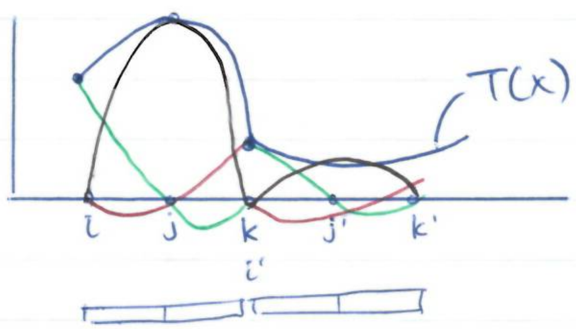
note $N_i^e(x_j) = \delta_{ij}$

Graphically,



Quadratic interpolation.

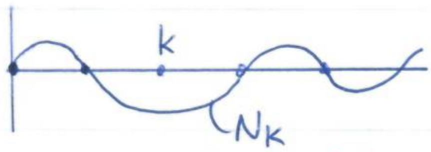
Two elements:



Higher order elements

$$T(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \cdots$$

Looking for polynomials that are zero at all nodes except one where it is 1.



note missing x_k term

$$P_k^{n-1}(x) = \frac{(x-x_1)(x-x_2)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_1)(x_k-x_2)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)}$$

Lagrange interpolation formula

Clearly $P_k^{n-1}(x_i) = \begin{cases} 0 & i \neq k \\ 1 & i = k \end{cases}$.

Take for shape f^{n}s of n-node element

$$N_k^e(x) = P_k^{n-1}(x) \qquad k=1,\dots,n$$

<u>for $n=2$</u>: $P_1^1(x) = \dfrac{x-x_2}{x_1-x_2} = \dfrac{-(x-x_2)}{L}$

$$P_2^1(x) = \frac{x-x_1}{x_2-x_1} = \frac{x-x_1}{L} \qquad L = x_2-x_1 .$$

<u>Properties of shape f^{n}s</u>

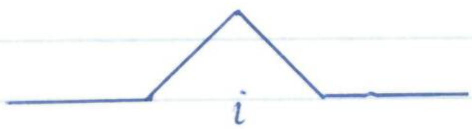
(1) $T(x) = \alpha_1 + \alpha_2 x + \dots + \alpha_n x^{n-1} = \sum_{i=1}^{n} N_i(x) T_i$

Interpolation for any f^n $T(x)$,
take special case $T(x) = T_0$ (const.)

$$\Rightarrow \quad T_0 = \sum_{i=1}^{n} N_i(x) T_0$$

$$\Longleftrightarrow \quad \boxed{\sum_{i=1}^{n} N_i(x) = 1} \quad \left(\begin{array}{c}\text{partition}\\ \text{of unity}\end{array}\right)$$

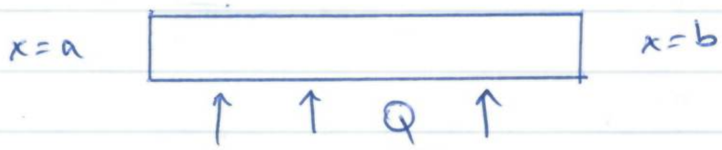
(2) $N_i(x) \neq 0$ only on those elements that contain node i



(3) $N_i(x_j) = \delta_{ij}$ ($N_i = 1$ at its own node i and 0 at all other nodes)

First FE eq^n in 1D :

Heat : $\frac{d}{dx}\left(Ak\frac{dT}{dx}\right) + Q = 0$



$$T(x) = \sum_{i=1}^{n} N_i(x)\, T_i \qquad \text{(interpolation)}$$

(will play the role of a trial f^n in a Galerkin weighted residual method)

Pass from strong formulation to weak formulation.

$$\int_a^b \left(\frac{d}{dx}\left(Ak\frac{dT}{dx}\right) + Q \right) v \, dx = 0$$

$$\Longleftrightarrow \int_a^b \left(-\frac{dv}{dx} Ak \frac{dT}{dx} + Qv \right) dx - \left[vAq \right]_a^b = 0$$

(using the relation $q = -k\frac{dT}{dx}$)

$$\Longleftrightarrow \int_a^b \frac{dv}{dx} Ak \frac{dT}{dx}\, dx = \int_a^b Qv\, dx - \left[vAq \right]_a^b$$

Galerkin: for test functions v take $N_i(x)$

and $T = \sum_j N_j T_j$

$$\Rightarrow \sum_{j=1}^{n} \int_a^b \frac{dN_i}{dx} Ak \frac{dN_j}{dx}\, dx \; T_j$$

$$= -\left[N_i A q\right]_a^b + \int_a^b Q N_i \, dx$$

or $\boxed{\sum_{j=1}^{n} K_{ij} T_j = f_{b_i} + f_{l_i}}$ F.E. EQ$^{\underline{N}}$

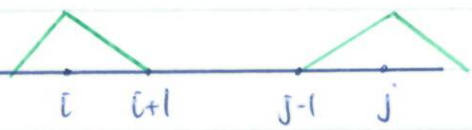
where $\boxed{K_{ij} = \int_a^b \frac{dN_i}{dx} Ak \frac{dN_j}{dx}\, dx}$ STIFFNESS MATRIX

$$f_{l_i} = \int_a^b Q N_i \, dx \qquad \text{LOAD/ FORCE VECTOR}$$

$$f_{b_i} = -\left[N_i A q\right]_a^b \qquad \text{BOUNDARY VECTOR}$$

<u>Properties of the stiffness matrix</u>

(1) <u>Bounded</u> $\because$ $N_i N_j$ will be nonzero only if i and j belong to the same element

2node element: 

$$N_i N_j = 0 \quad \text{if} \quad |i-j| > 1$$

2-node element: $\begin{pmatrix} \times\times\times & & & \\ \times\times\times & & 0 & \\ & \times\times\times & & \\ 0 & & \times\times\times & \\ & & & \times\times\times \end{pmatrix}$ tridiagonal

for higher-order elements, the 'bandwidth' will be larger.

(2) <u>Symmetric</u> : $K_{ij} = K_{ji}$

(3) <u>Singular</u> : $\det K = 0$. K does not depend on Q, so consider the homogeneous eq^n

$$\frac{d}{dx}\left(Ak\frac{dT}{dx}\right) = 0 \implies \text{F.E. Eq}^n \quad Ku = 0, \quad u = \begin{pmatrix} T_1 \\ \vdots \\ T_n \end{pmatrix}$$

Any constant T is a sol^n so K must be singular

(4) <u>Rows and columns sum up to zero</u> : follows from partition of unity :

$$\sum_i N_i(x) = 1 \implies$$

$$\sum_i K_{ij} = \int_a^b \frac{d\overset{=1}{\sum N_i(x)}}{dx} Ak \frac{dN_j}{dx}\, dx$$

$$= 0$$

Same for $\sum_j K_{ij} = 0$.

<u>Property of the boundary vector</u>

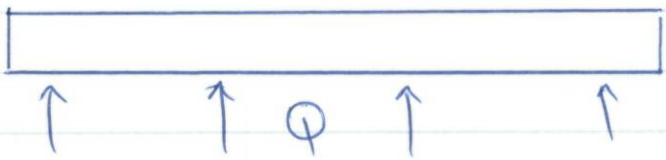
$$f_{bi} = -\left[N_i A q\right]_a^b$$

$x = a$ corresponds to node 1

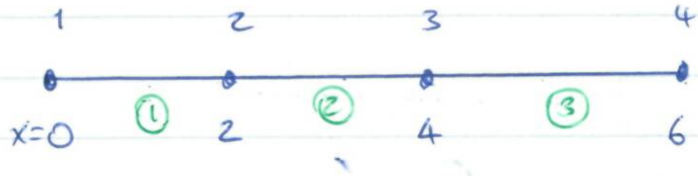
$x = b$ . . . . n

$$\implies f_b = \begin{pmatrix} (Aq)|_{x=a} \\ 0 \\ \vdots \\ 0 \\ -(Aq)|_{x=b} \end{pmatrix}$$

Example (heat flow in a uniform fin)



model with 3 2-node elements

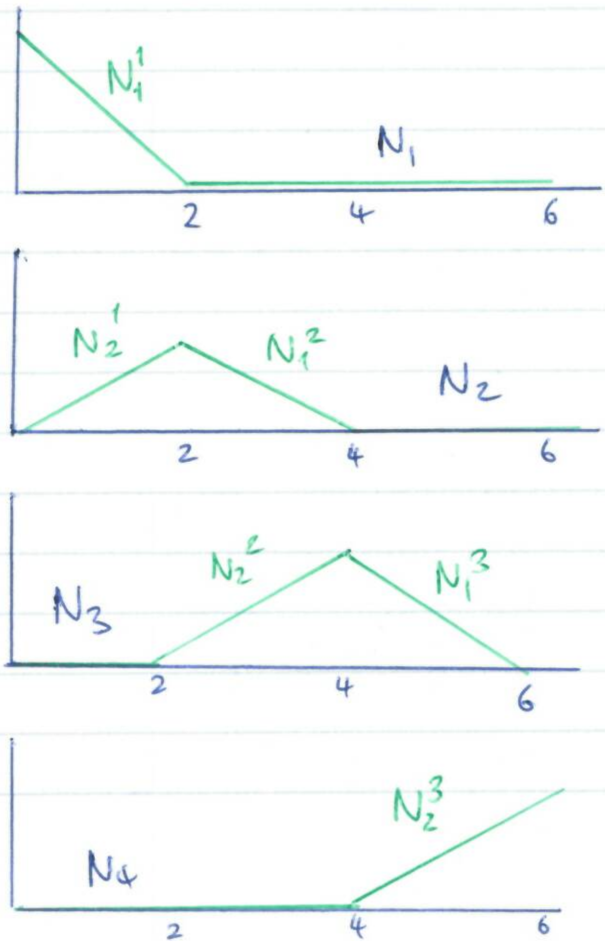


$$\frac{d}{dx}\left(Ak\,\frac{dT}{dx}\right) + Q = 0$$

$$T(0) = T_0$$

$$q(6) = \bar{q}$$

Global shape f^{n}s



Element shape f^{n}s :

$$N_1^1(x) = -\frac{(x-x_2)}{L} = \frac{2-x}{2} \qquad (x_2 = 2,\ L=2)$$

$$N_2^1(x) = \frac{(x-x_1)}{L} = \frac{x}{2}$$

$$N_1^2(x) = \frac{-(x-x_3)}{L} = \frac{4-x}{2}$$

$$N_2^2(x) = \frac{(x-x_2)}{L} = \frac{x-2}{2}$$

$$N_1^3(x) = \frac{-(x-x_4)}{L} = \frac{6-x}{2}$$

$$N_2^3(x) = \frac{x-x_3}{L} = \frac{x-4}{2}$$

Stiffness matrix $\quad K_{ij} = \int_0^6 \frac{dN_i}{dx} Ak \frac{dN_j}{dx}\, dx$

$$K_{11} = \int_0^6 \frac{dN_1}{dx} Ak \frac{dN_1}{dx}\, dx$$

$$= \int_0^2 \frac{dN_1^1}{dx} Ak \frac{dN_1^1}{dx}\, dx =$$

$$= \int_0^2 \left(-\tfrac{1}{2}\right)^2 Ak\, dx$$

$$= \tfrac{1}{2} Ak$$

$$K_{12} = \int_0^6 \frac{dN_1}{dx} Ak \frac{dN_2}{dx}\, dx$$

$$= \int_0^2 \frac{dN_1^1}{dx} Ak \frac{dN_2^1}{dx}\, dx$$

$$= \int_0^2 \left(-\tfrac{1}{2}\right) Ak \left(\tfrac{1}{2}\right) dx$$

$$= -\tfrac{1}{2} Ak$$

~~K~~ $$K_{13} = \int_0^6 \frac{dN_1}{dx} Ak \frac{dN_3}{dx}\, dx$$

$$= 0 \quad \text{(no overlap of } N_1 \text{ and } N_3\text{)}$$

$$K_{14} = 0 \quad \text{(same reason)}.$$

$$K_{22} = \int_0^6 \frac{dN_2}{dx} Ak \frac{dN_2}{dx}\, dx$$

$$= \int_0^2 \frac{dN_2^1}{dx} Ak \frac{dN_2^1}{dx}\, dx + \int_2^4 \frac{dN_1^2}{dx} Ak \frac{dN_1^2}{dx}\, dx$$

$$= \int_0^2 \left(\frac{1}{2}\right)^2 Ak\, dx + \int_2^4 \left(-\frac{1}{2}\right)^2 Ak\, dx$$

$$= Ak$$

etc.

Get
$$K = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \frac{1}{2} Ak$$

Load vector: $$f_{e_1} = \int_0^6 Q N_1\, dx$$

$$= Q \int_0^2 N_1^1\, dx$$

$$= Q \int_0^2 \frac{2-x}{2}\, dx$$

$$= Q$$

$$f_{e_2} = \int_0^6 Q N_2 \, dx$$

$$= Q \int_0^2 N_2^1 \, dx + Q \int_2^4 N_1^2 \, dx$$

$$= 2Q \qquad \rightarrow \underline{f_e = Q \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix}}$$

Boundary vector: $f_{b_i} = -\left[N_i A q \right]_0^6$

$$= \underbrace{-(N_i A q)\big|_{x=6}}_{A q(6)\,\delta_{i4}} + \underbrace{(N_i A q)\big|_{x=0}}_{A q(0)\,\delta_{i1}}$$

Kron.

or $f_b = \begin{pmatrix} A q(0) \\ 0 \\ 0 \\ -A q(6) \end{pmatrix}$

FE eq^n: $\frac{1}{2} A k \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{pmatrix}$

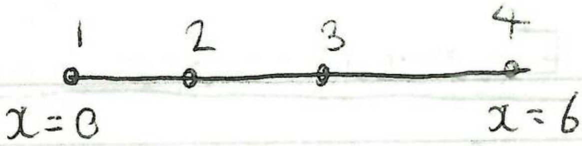
$$= Q \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix} + \begin{pmatrix} A q(0) \\ 0 \\ 0 \\ -A q(6) \end{pmatrix} \qquad (*)$$

Apply b.c.s : $T_1 = T_0$

$q(6) = \bar{q}$

Exercise: show that the FE sol$^{\underline{n}}$ is exact at the nodes.

$$\begin{cases} \dfrac{d}{dx}\left(Ak\dfrac{dT}{dx}\right) + Q = 0 \\ T(0) = T_0 \\ q(6) = \bar{q} \end{cases}$$



FE eq: $$\frac{1}{2}Ak\begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} \overset{T_0}{\overset{\shortparallel}{T_1}} \\ T_2 \\ T_3 \\ T_4 \end{pmatrix}$$

$$= Q\begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix} + \begin{pmatrix} Aq(0) \\ 0 \\ 0 \\ -A\underset{\bar{q}}{\underset{\shortparallel}{q(6)}} \end{pmatrix} \qquad (*)$$

New stuff starts here

<u>2-step soln</u>

1. Solve reduced problem for (T_2, T_3, T_4)

$$\begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} T_2 \\ T_3 \\ T_4 \end{pmatrix} = \begin{pmatrix} 2Q + \frac{1}{2}AkT_0 \\ 2Q \\ Q - A\bar{q} \end{pmatrix}$$

$\Rightarrow T_2 = \ldots, T_3 = \ldots, T_4 = \ldots$

2. $\frac{1}{2}Ak(T_0 - T_2) = Q + Aq(0)$ [first row of (*)]

$\Rightarrow q(0) = \bar{q} - \dfrac{6Q}{A}$ (using T_2 solution)

Exact solution in this case

$$\frac{d^2T}{dx^2} = \frac{-Q}{Ak} \Rightarrow \frac{dT}{dx} = -\frac{Qx}{Ak} + c$$

$$BC: q(6) = -k\left.\frac{dT}{dx}\right|_{x=6} = \bar{q}$$

$$\Rightarrow c = \frac{-\bar{q}}{k} + \frac{6Q}{Ak}$$

$$\Rightarrow \frac{dT}{dx} = \frac{-\bar{q}}{k} - \frac{Q}{Ak}(x-6)$$

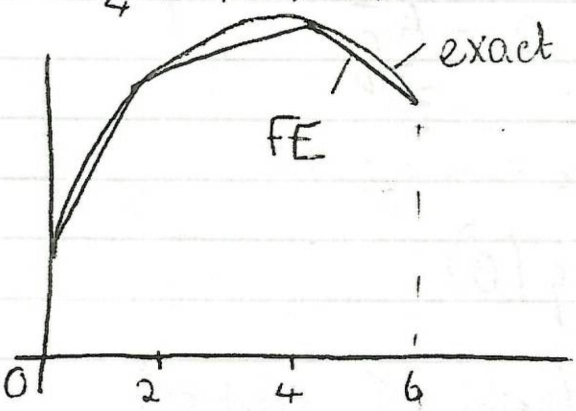
$$\Rightarrow T(x) = \frac{-\bar{q}}{k}x - \frac{Q}{Ak}\left(\tfrac{1}{2}x^2 - 6x\right) + d$$

$$BC: T(0) = T_0 = d$$

exact sol^n

$$\Rightarrow \text{Solution. } T(x) = T_0 - \frac{\bar{q}x}{k} - \frac{Q}{2Ak}x(x-12)$$

$$\Rightarrow T_2 = T(2) = \ldots\ldots$$
$$T_3 = T(4) = \ldots\ldots$$
$$T_4 = T(6) = \ldots\ldots$$



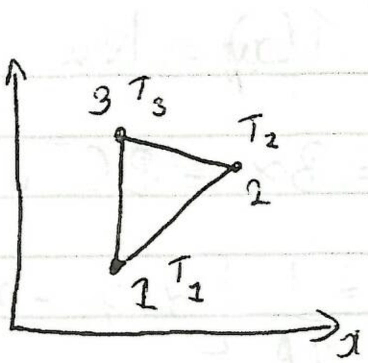
Conclusion

FE exact at nodes but linear interpolation in between.

(this is **not** a general rule)

Two-dimensional elements

3-node triangular element



$$T(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y$$
$$= \underline{B}\,\underline{\alpha}$$

where $B = \begin{pmatrix} 1 & x & y \end{pmatrix}$ $\qquad \alpha = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix}$

want to express $T(x,y) = N_1^e(x,y)\,T_1 + N_2^e(x,y)\,T_2 + N_3^e(x,y)\,T_3$

for consistency $T(x_1, y_1) = T_1$, $T(x_2, y_2) = T_2$, $T(x_3, y_3) = T_3$

$$T_1 = \alpha_1 + \alpha_2 x_1 + \alpha_3 y_1$$
$$T_2 = \alpha_1 + \alpha_2 x_2 + \alpha_3 y_2$$
$$T_3 = \alpha_1 + \alpha_2 x_3 + \alpha_3 y_3$$

$$\Longleftrightarrow \begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{pmatrix} = \begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix}$$

Write $C\alpha = u$, $u = \begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix}$

Also $T(x,y) = Nu$ where $N = (N_1^e, N_2^e, N_3^e)$

$\Rightarrow T = B\alpha = BC^{-1}u$ Hence $N = BC^{-1}$

$$N_1^e(x,y) = \frac{1}{2A}\left[x_2y_3 - x_3y_2 + (y_2 - y_3)x + (x_3 - x_2)y\right]$$

$$N_2^e(x,y) = \frac{1}{2A}\left[x_3y_1 - x_1y_3 + (y_3 - y_1)x + (x_1 - x_3)y\right]$$

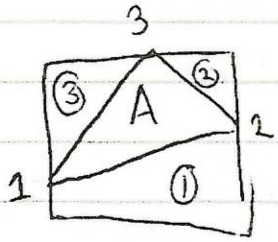
$$N_3^e(x,y) = \frac{1}{2A}\left[x_1y_2 - x_2y_1 + (y_1 - y_2)x + (x_2 - x_1)y\right]$$

This is a
Cyclic permutation!

(Here, $2A = \det C$)

Claim: A is the area of the triangle

Proof



$$A = (x_2 - x_1)(y_3 - y_1) - ① - ② - ③$$

$$① = \frac{1}{2}(x_2 - x_1)(y_2 - y_1)$$

$$② = \frac{1}{2}(x_2 - x_3)(y_3 - y_2)$$

$$③ = \frac{1}{2}(x_3 - x_1)(y_3 - y_1)$$

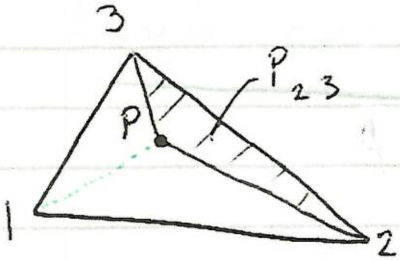
$$\Rightarrow A = \frac{1}{2}\left[(x_2y_3 - x_3y_2) + (x_1y_2 - x_2y_1) + (x_3y_1 - x_1y_3)\right]$$

$$= \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}$$

$$= \frac{1}{2} \det C$$

$$\Leftrightarrow \quad 2A = \det C$$

<u>Area coordinates</u>



$$\eta_1 = \frac{\text{area}(P_{23})}{\text{area}(123)}$$ total area

$$\eta_2 = \frac{\text{area}(P_{31})}{\text{area}(123)}$$

$$\eta_3 = \frac{\text{area}(P_{12})}{\text{area}(123)}$$

Clearly $\eta_1 + \eta_2 + \eta_3 = 1$

If P coincides with node 1 then $\eta_1 = 1$, $\eta_2 = 0$, $\eta_3 = 0$

In fact $\eta_i \overbrace{(x_j, y_j)}^{P} = \delta_{ij}$

<u>Fact</u> $\eta_i(x,y) = N_i(x,y)$ (shape functions are the area coordinates)

Proof

$$\eta_1 = \frac{\text{area }(P_{23})}{\text{area }(123)}$$

$$= \frac{\frac{1}{2}\det\begin{vmatrix} 1 & x & y \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}}{A}$$

$$= \ldots$$

$$= N_1(x,y)$$

Inverse relationship: $x = x_1\eta_1 + x_2\eta_2 + x_3\eta_3$

$$y = y_1\eta_1 + y_2\eta_2 + y_3\eta_3$$

$$(\eta_3 = 1 - \eta_1 - \eta_2)$$

The area coordinates are convenient when computing integrals of type $I_{nm} = \int_{\triangle} x^n y^m \, dx\, dy$ that appear in stiffness matrices and load vectors

(Recall $\cancel{K_{ij} = \iint_{\triangle} \partial N_i}$ $\quad f_{Li} = \iint_{\triangle} Q N_i(x,y)\, dx\, dy$)

Idea: transform $(x, y) \Rightarrow (\eta_1, \eta_2, \eta_3)$

$$\Rightarrow \int_{\triangle} \eta_1^m \eta_2^n \eta_3^p \, dx\, dy$$

$$= \frac{m!\, n!\, p!}{(m+n+p+2)!} \quad (2A)$$

Example

1. $$I_{00} = \int_{\triangle} dx\, dy = A$$

2. $$I_{10} = \int_{\triangle} x \, dx\, dy = x_1 \int_{\triangle} \eta_1 \, dx\, dy + x_2 \int_{\triangle} \eta_2 \, dx\, dy$$

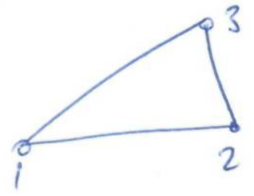
$$+ x_3 \int_{\triangle} \eta_3 \, dx\, dy$$

$$= \frac{A}{3}(x_1 + x_2 + x_3)$$

3. $$I_{11} = \int_{\triangle} xy \, dx\, dy$$

$$= \ldots$$

Triangular elements



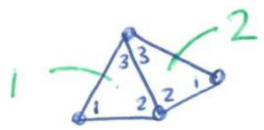
$\left(T = \alpha_1 + \alpha_2 x + \alpha_3 y = N_1^e T_1 + N_2^e T_2 + N_3^e T_3 \right)$

Shape f$^{\underline{n}}$s N_1^e, N_2^e, N_3^e

Property: $N_i(x_j, y_j) = \delta_{ij}$

$\Rightarrow N_i(x, y) = 0$ along entire edge 2-3.

Conformity (or compatibility)



A mesh of elements is conforming if the shape functions, and hence the sol$^{\underline{n}}$ T, is guaranteed to be continuous across element boundaries.

Need to show $N_2^1 = N_2^2$ along boundary 2-3

We can write for the boundary $\quad y = c_1 + c_2 x$

$$\Rightarrow N_2^1(x, y) = \beta_1 + \beta_2 x + \beta_3 y$$
$$= a_1 + b_1 x \qquad (\because y = f(x))$$

and $N_2^2(x, y) = a_2 + b_2 x$

But we have 4 conditions:

$$N_2^1(x_2, y_2) = 1 = N_2^2(x_2, y_2)$$
$$N_2^1(x_3, y_3) = 0 = N_2^2(x_3, y_3)$$

$\Rightarrow$ can uniquely determine coefficients a_1, a_2, b_1, b_2

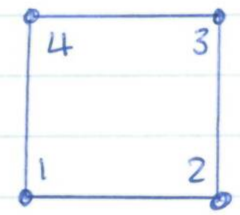
$\Rightarrow$ we find $a_1 = a_2$ and $b_1 = b_2$

$\Rightarrow N_2^1 = N_2^2$ along common boundary

and $N_3^1 = N_3^2$ along common boundary similarly.

$\Rightarrow$ mesh of arbitrary triangular elements is conforming.

4-node rectangular element



to keep symmetry in x and y

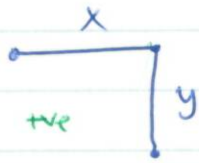
$$T = \alpha_1 + \alpha_2 x + \alpha_3 y + \underbrace{\alpha_4 xy}$$

We want again $\quad T = \sum_{i=1}^{4} N_i T_i$.

Can use direct method to construct shape functions. But we can also obtain them by taking products of 1D Lagrange polynomials.

$$N_1(x,y) = P_1^1(x)\, P_1^1(y)$$

$$= \frac{x - x_2}{x_1 - x_2} \cdot \frac{y - y_4}{y_1 - y_4}$$



$$= \frac{1}{4ab}(x - x_2)(y - y_4)$$

area

$$N_2(x,y) = P_2^1(x)\, P_1^1(y) = -\frac{1}{4ab}(x - x_1)(y - y_3)$$ -ve

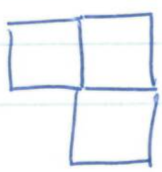
$$N_3(x,y) = P_2^1(x)\, P_2^1(y) = \frac{1}{4ab}(x - x_4)(y - y_2)$$ +ve

$$N_4(x,y) = P_1^1(x)\, P_2^1(y) = -\frac{1}{4ab}(x - x_3)(y - y_1)$$ -ve

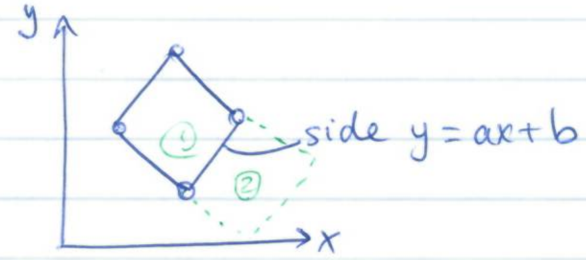
(note: $N_i(x_j, y_j) = \delta_{ij}$)

A mesh of these elements is conforming (proof similar to triangular case: N uniquely determined by nodal values).

However, it does require that the sides are parallel to the x- and y-axes.



Otherwise, consider



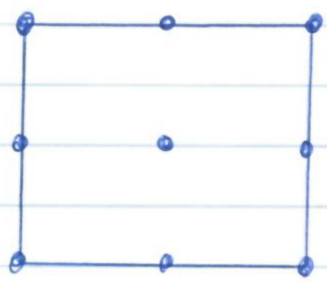
$$\Rightarrow \quad T = \alpha_1 + \alpha_3 b + (\alpha_2 + \alpha_3 a + \alpha_4 b)\,x + \alpha_4 a x^2$$
$$= \beta_1 + \beta_2 x + \beta_3 x^2$$

Quadratic
3 coefficients to find $\beta_1, \beta_2, \beta_3$
2 nodes of info T_1, T_2
$\Downarrow$

- not uniquely defined.
- no guarantee that ① and ② give same values.

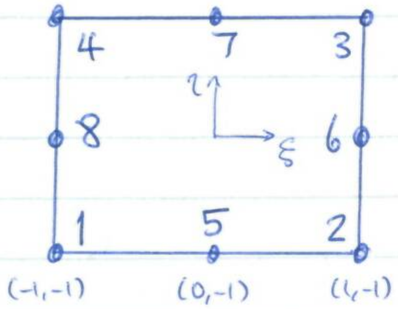
We can solve this by modifying the mesh (at the cost of more nodes).

9-node rectangular element



Shape f^ns are products of quadratic Lagrange elements along x and y.

8-node rectangular element



Not Lagrange!

We can construct shape f^ns for this element by using the

$$N_i(x_j, y_j) = \delta_{ij}$$

property:

Introduce local coordinates (ξ, η)

Consider mid-side node:

$$N_5(x,y) = \overbrace{\alpha}^{\text{norm}}\overbrace{(\xi+1)(\xi-1)(\eta-1)}^{\text{these kill all the other nodes.}}$$

needs to be 1 at own node and 0 elsewhere.

where $\alpha = \frac{1}{2}$ $\because$ $N_5(0,-1) := 1$

Consider corner node:

$$N_1(x,y) = \alpha(\xi-1)(\eta-1)\overbrace{(1+\xi+\eta)}^{\text{diagonal line connecting 8+5.}}$$

(Turns out you can require N_1 to be 0 along entire line connecting 5 + 8).

where $\alpha = -\frac{1}{4}$.

This element is also known as the "<u>Serendipity element</u>"

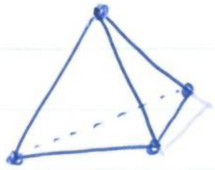
Approximation in this case:

$$T = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 y^2 + \alpha_7 x^2 y + \alpha_8 xy^2$$

A mesh of such elements is conforming provided the sides are parallel to the x- and y-axes.

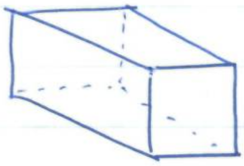
3D elements

4-node tetrahedral element:



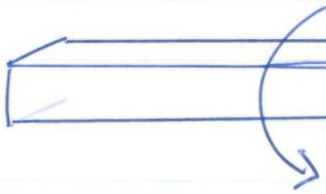
$T_1 = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 z$

8-node prism/brick element

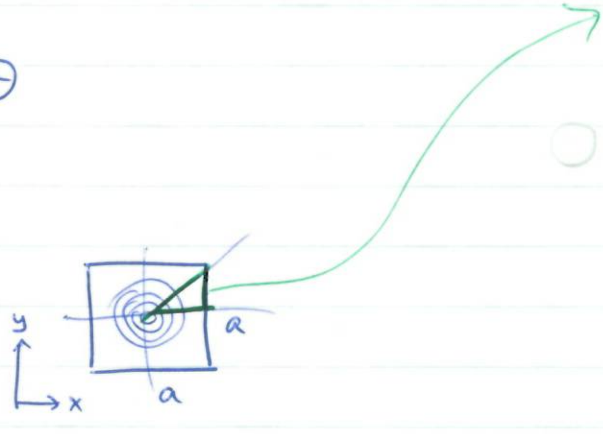


2D example (the big one!)

Torsion of square cross-section shaft



Consider square cross-section



Governing eq^n for stress in shaft:

$$\begin{cases} \dfrac{\partial^2\phi}{\partial x^2} + \dfrac{\partial^2\phi}{\partial y^2} + 2G\theta = 0 \\ \phi = 0 \text{ on boundary} \end{cases}$$

(shaft only loaded at the end)

ϕ stress f^n

G shear

θ twist rate

by symmetry about $(x, y, y=x)$ we need only consider an eighth of the section

On this reduced domain we have

$$\begin{cases} \dfrac{\partial^2\phi}{\partial x^2} + \dfrac{\partial^2\phi}{\partial y^2} + 2G\theta = 0 \\ \phi = 0 \quad \text{on } \Gamma_g \quad \text{(original outer boundary} \\ \dfrac{\partial\phi}{\partial n} := \underline{\nabla}\phi \cdot \underline{n} = 0 \quad \text{on } \Gamma_h \end{cases}$$

by symmetry

$(\underline{\nabla}\phi$ parallel to boundary$)$

Weak formulation:

$$\iint_\Delta \left(\frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} + 2G\theta \right) v \, dx\, dy = 0$$

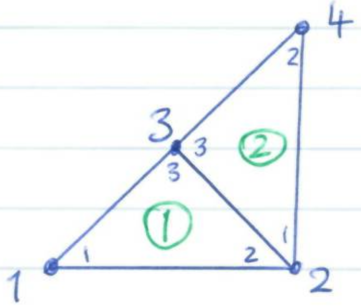
$$\overset{G\text{-}G}{\Longleftrightarrow} \iint_\Delta \left(-\frac{\partial\phi}{\partial x}\frac{\partial v}{\partial x} - \frac{\partial\phi}{\partial y}\frac{\partial v}{\partial y} + 2G\theta v \right) dx\, dy$$

$$+ \int_\Gamma \left[v\frac{\partial\phi}{\partial x} n_x + v\frac{\partial\phi}{\partial y} n_y \right] ds = 0$$

$$\overset{G-G}{\Longleftrightarrow} \iint_\Delta \left(\frac{\partial \phi}{\partial x}\frac{\partial v}{\partial x} + \frac{\partial \phi}{\partial y}\frac{\partial v}{\partial y} \right) dx\,dy$$

$$= \int_{\Gamma_g} v \frac{d\phi}{dn}\, ds + 2G\theta \iint_\Delta v\, dx\,dy$$

Choose mesh of two 3-node triangular elements.



For single element we introduce the interpolation

$$\phi = \sum_{j=1}^{3} N_j^e \phi_j$$

Galerkin: $v = N_i^e \quad (i = 1, 2, 3)$

$$\Rightarrow \sum_{j=1}^{3} \iint_\Delta \left(\frac{\partial N_j^e}{\partial x}\frac{\partial N_i^e}{\partial x} + \frac{\partial N_j^e}{\partial y}\frac{\partial N_i^e}{\partial y} \right) \phi_j \, dx\,dy$$

$$= \int_{\Gamma_g} N_i^e \frac{d\phi}{dn}\, ds + 2G\theta \iint_\Delta N_i \, dx\,dy$$

FE eq^n: $\sum_{j=1}^{3} K_{ij}\phi_j = f_{e_i} + f_{b_i}$ where

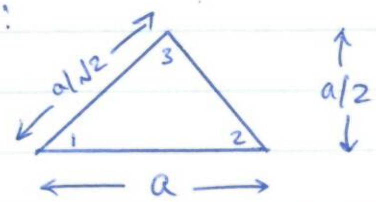
where $K_{ij} = \iint_\Delta \left(\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} \right) dx\,dy$

(stiffness matrix)

$$f_{e_i} = 2G\theta \iint_{\Delta} N_i \, dx\,dy \qquad \text{(load vector)}$$

$$f_{b_i} = \int_{\Gamma_g} N_i \frac{d\phi}{dn} \, ds$$

Element ① :



$A = \text{area} = \frac{a^2}{4}$

Shape f^ns: $N_1^e = \frac{1}{2A}\left[x_2y_3 - x_3y_2 + (y_2-y_3)x + (x_3-x_2)y\right]$

N_2^e, N_3^e cyclic.

$$\Rightarrow \quad K_{11} = \iint_{\Delta} \left[(y_2-y_3)^2 + (x_3-x_2)^2\right] \frac{1}{4A^2} \, dx\,dy$$

$$K_{12} = 0$$

$$\vdots$$

$$\Rightarrow \qquad K = \frac{1}{2}\begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{pmatrix}$$

note rols + cols add to 0, as they should.
So you only need to compute 2 and you know the 3rd!

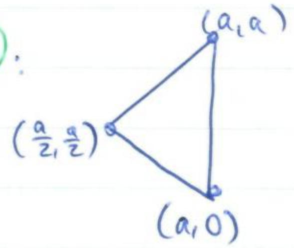
Load vector: remember that $N_i = \eta_i$ (area coords)

Integration formula: $\iint_{\Delta} \eta_1^n \eta_2^m \eta_3^p \, dx\,dy$

$$= \frac{m!\,n!\,p!\,(2A)}{(m+n+p+2)!}$$

$$\Rightarrow \quad f_e^1 = \frac{2}{3} G\theta A \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

__Element 2:__



Element congruent to element 1 with equivalent local node numbering $\Rightarrow$ $K^2 = K^1$.

Also $f_e^2 = f_e^1$

__Q:__ How to 'add up' these local element data into global data?

__Assembly:__ Element 1:

global degrees of freedom

$$u^1 = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix}$$

local degrees of freedom

$$=: A^1 u$$

u two

Element 2: $$u^2 = \begin{pmatrix} \phi_2 \\ \phi_4 \\ \phi_3 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix}$$

$$=: A^2 u$$

$$K^1 u^1 = f_\ell^1 \Rightarrow K^1 A^1 u = f_\ell^1$$

$$\Rightarrow A^{1T} K^1 A^1 u = A^{1T} f_\ell^1$$

Also, for element ②, $A^{2T} K^2 A^2 u = A^{2T} f_\ell^2$

Now sum over elements,

$$\sum_e A^{eT} K^e A^e u = \sum_e A^{eT} f_\ell^e$$

or

$$Ku = f_\ell$$

where $K = \sum_e A^{eT} K^e A^e$ (global stiffness matrix)

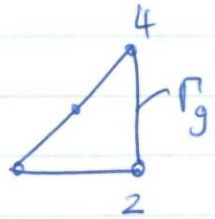
$f_\ell = \sum_e A^{eT} f_\ell^e$ (global load vector)

We find

$$K = \frac{1}{2}\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 2 & -2 & 0 \\ -1 & -2 & 4 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

$$f_\ell = \frac{2}{3} G A \theta \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix}$$

Boundary vector: $f_{b_i} = \int_{\Gamma_g} N_i \frac{d\phi}{dn} ds$



$\because$ Γ_g along 2-4 edge only, only 2nd + 4th component will be nonzero

Along Γ_g the only nonzero global shape f^{n}s are N_2 and N_4.

Thus $f_b = \begin{pmatrix} 0 \\ R_2 \\ 0 \\ R_4 \end{pmatrix}$ where $R_2 = \int_{\Gamma_g} N_2 \frac{d\phi}{dn} ds$

$\Rightarrow$ FE eq^n: $\frac{1}{2}\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 2 & -2 & 0 \\ -1 & -2 & 4 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{pmatrix}$

$$= \begin{pmatrix} 0 \\ R_2 \\ 0 \\ R_4 \end{pmatrix} + \frac{2}{3} G\theta A \begin{pmatrix} 1 \\ 2 \\ 2 \\ 1 \end{pmatrix}$$

Boundary conditions: $\phi = 0$ on Γ_g

$$\Rightarrow \phi_2 = 0 = \phi_4$$

Reduced system for (ϕ_1, ϕ_3):

$$\frac{1}{2}\begin{pmatrix} 1 & -1 \\ -1 & 4 \end{pmatrix}\begin{pmatrix} \phi_1 \\ \phi_3 \end{pmatrix} = \frac{2}{3} G\theta A \begin{pmatrix} 1 \\ 2 \end{pmatrix}$$

$$\Rightarrow \phi_1 = \frac{8}{3} G\theta A \qquad \Rightarrow \begin{cases} R_2 = -\frac{8}{3} GA\theta \\ R_4 = -\frac{4}{3} GA\theta \end{cases}$$

$$\phi_3 = \frac{4}{3} G\theta A$$

$\boxed{L}$

General approach

1. Formulate boundary-valued problem
2. Weak formulation
3. Mesh (discretisation)
4. Interpolation $[\phi = \sum \phi_i N_i]$
5. Compute element data
6. Assembly into global data
7. Apply BCs
8. Solution for ϕ_i's.

Let's see if this result makes any sense.

(i) Torsional stiffness $C = \frac{M}{\theta}$,

where the twisting moment M is given by

$$M = 2 \iint_{\square} \phi \, dx dy$$

we have $\square$ $= 8 \left[2 \int_{①} \phi \, dx dy + 2 \int_{②} \phi \, dx dy \right]$

$$= 8 \left[2 \int_{①} (N_1^e \phi_1 + N_2^e \phi_2 + N_3^e \phi_3) \, dx \, dy + 2 \int_{②} (N_1^e \phi_2 + N_2^e \phi_4 + N_3^e \phi_3) \, dx \, dy \right]$$

$$= 16 \left[\frac{A}{3} (1 \; 1 \; 1) \begin{pmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{pmatrix} + \frac{A}{3} (1 \; 1 \; 1) \begin{pmatrix} \phi_2 \\ \phi_4 \\ \phi_3 \end{pmatrix} \right]$$

because $\iint_{\Delta} N_i(x,y) \, dx \, dy = \frac{A}{3}$.

Plugging in our values for ϕ_i:

$$\frac{16}{3} A(\phi_1 + 2\phi_3) = \frac{256}{9} G\theta A^2$$

split our element into 2   we had $\frac{1}{8}$

2×8

$$\Rightarrow \quad C = \frac{M}{\theta} = \frac{256}{9} GA^2$$

$$= \frac{256}{9} G\left(\frac{\bar{A}}{16}\right)^2$$

where $\bar{A} = 16A$, total cross-sect. area of square rod.

$$= \frac{1}{9} G\bar{A}^2$$

$$= 0.1111 \, G\bar{A}^2$$

Exact result: $C = k_1 G\bar{A}^2$ with $k_1 = 0.1406$, so we have a

21% error.

(2) Compute the stressf^n over the entire domain (equipotential lines)

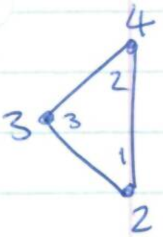
Element-wise:

①: $\phi = N_1\phi_1 + N_2\phi_2 + N_3\phi_3$ (in terms of global shape f^{n}s)
$= N_1^e\phi_1 + N_2^e\phi_2 + N_3^e\phi_3$

$$N_1^e = \frac{1}{2A}\left[x_2y_3 - x_3y_2 + (y_2 - y_3)x + (x_3 - x_2)y\right]$$

$N_3^e = \ldots$

$$\Rightarrow \phi = \frac{8}{3} G\theta A\left(1 - \frac{x}{a}\right)$$



② : $\phi = N_2\phi_2 + N_3\phi_3 + N_4\phi_4$ (globally)

$= N_1^e\phi_2 + N_3^e\phi_3 + N_2^e\phi_4$ (locally)

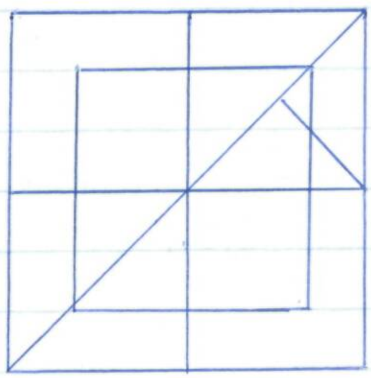
$ \overset{\shortparallel}{0} \qquad\qquad\qquad\qquad \overset{\shortparallel}{0}$

$$= \frac{8}{3} G\theta A \left(1 - \frac{x}{a}\right)$$

<u>Note</u>: $\phi^1 = \phi^2$ (in particular, cts across element bdy

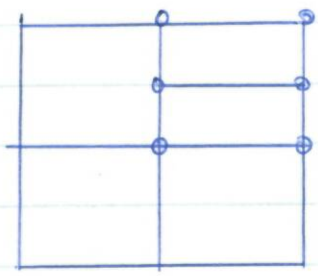
ϕ^1 and ϕ^2 satisfy the b.c. at $x = a$ (zero)

Constant ϕ curves: $\phi = \text{const} \Rightarrow x = \text{const}$



Makes sense to this <u>linear</u> order of approx.

[<u>Exercise</u>:



Consider a mesh of 2 rectangular elements for a quarter of the shaft's X-section.]

<u>Assembly</u>: $K = A^{1T}K^1A^1 + A^{2T}K^2A^2$

$$= \frac{1}{2}\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ -1 & -1 & 2 \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$+ \frac{1}{2}\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} K^2 \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

$$= \frac{1}{2}\begin{pmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 \\ -1 & -1 & 2 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} + \frac{1}{2}\begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$$

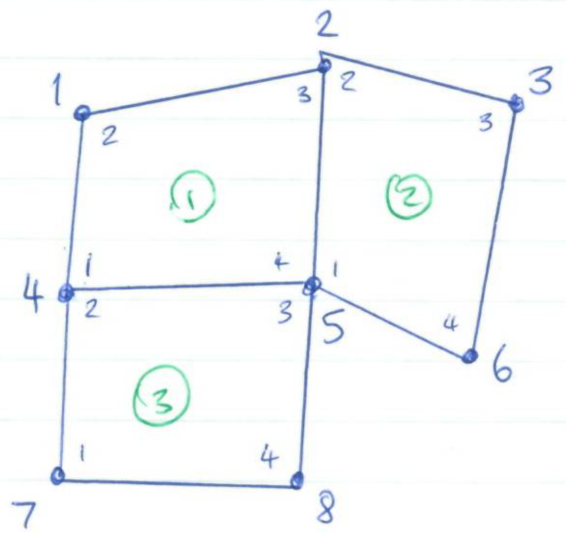
More convenient way to encode the network topology is to use the connectivity matrix / table.

$$C = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 4 & 3 \end{pmatrix}$$

$\xrightarrow{\text{horizontally}}$ global nodes in order of local nodes

vertically $\downarrow$ elements

<u>Example</u>



(as on sheet)

An Example of the Assembly Process

In Figure 2.12 is shown a small mesh containing three elements all of which have properties defined by (2.51). The problem is to assemble the element matrices into the complete system matrix. Assuming that there is only one unknown at each node the global node numbers will correspond to the global freedom numbers and the element node numbers will correspond to the local freedom numbers.

Each element possesses local freedom numbers (shown in parentheses) which follow the standard scheme, namely a consistent clockwise numbering.

Boolean matrices:

$$A_1 = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

$$A_3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

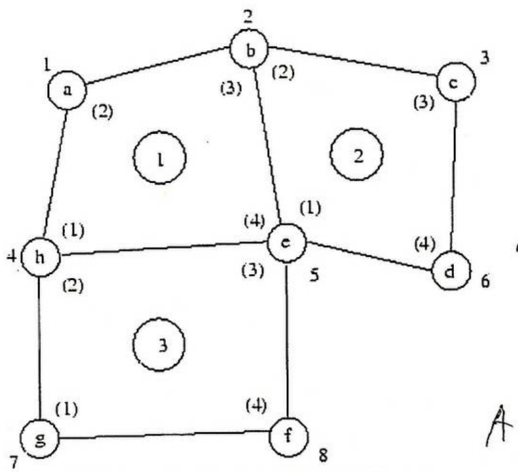


Figure 2.12: Mesh of quadrilateral elements

Each individual element equation can then be written as

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} \\ K_{21} & K_{22} & K_{23} & K_{24} \\ K_{31} & K_{32} & K_{33} & K_{34} \\ K_{41} & K_{42} & K_{43} & K_{44} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} \qquad (2.52)$$

Connectivity matrix:

$$C = \begin{pmatrix} 4 & 1 & 2 & 5 \\ 5 & 2 & 3 & 6 \\ 7 & 4 & 5 & 8 \end{pmatrix}$$

However, in the global freedom numbering system (not in parentheses) freedom 4 at node (h) corresponds to local freedom (1) in element 1 and local freedom (2) in element 3. In the assembled system matrix, term K_{11} from element 1 and K_{22} from element 3 would be added together and would appear in location (4,4) of the system matrix and so on. Similarly the contributions to global freedom 5 at node (e) will come from three elements. The assembly process is essentially additive. The only complication is in ensuring that the correct element contributions are added together. The total system matrix for Figure 2.12 is given in Table 2.3 where the superscripts refer to element numbers.

$$K = A_1^T K^1 A_1 + A_2^T K^2 A_2 + A_3^T K^3 A_3$$

Table 2.3: System stiffness matrix for mesh Figure 2.12

K^1_{22}	K^1_{23}	0	K^1_{21}	K^1_{24}	0	0	0
K^1_{32}	$K^1_{33}+K^2_{22}$	K^2_{23}	K^1_{31}	$K^1_{34}+K^2_{21}$	K^2_{24}	0	0
0	K^2_{32}	K^2_{33}	0	K^2_{31}	K^2_{34}	0	0
K^1_{12}	K^1_{13}	0	$K^1_{11}+K^3_{22}$	$K^1_{14}+K^3_{23}$	0	K^3_{21}	K^3_{24}
K^1_{42}	$K^1_{43}+K^2_{12}$	K^2_{13}	$K^1_{41}+K^3_{32}$	$K^1_{44}+K^2_{11}+K^3_{33}$	K^2_{14}	K^3_{31}	K^3_{34}
0	K^2_{42}	K^2_{43}	0	K^2_{41}	K^2_{44}	0	0
0	0	0	K^3_{12}	K^3_{13}	0	K^3_{11}	K^3_{14}
0	0	0	K^3_{42}	K^3_{43}	0	K^3_{41}	K^3_{44}

This matrix will be symmetric if its constituent matrices are symmetric, and also possesses the useful property of *bandedness*. That is, the terms making up Table 2.3 are concentrated around the *leading diagonal* which stretches from t upper left to the lower right of the table. In fact no term in any row can be more than 4 locations removed from the leading diagonal so the system is said to have a *semi-bandwidth* of 5. The semi-bandwidth is usually denoted by HBAND.

The semi-bandwidth can be obtained by inspection of Figure 2.10 by subtracting lowest freedom number from the highes each element and adding one. Complex meshes have variable bandwidths and computer programs make use of bandedn when storing the system matrices.

Chris Greenough (c.greenough@rl.ac.uk): September 2001

Connectivity matrix:

$$C = \begin{pmatrix} 4 & 1 & 2 & 5 \\ 5 & 2 & 3 & 6 \\ 7 & 4 & 5 & 8 \end{pmatrix}$$

Suppose $$K^1 = \begin{pmatrix} K^1_{11} & \cdots & K^1_{14} \\ \vdots & \ddots & \vdots \\ K^1_{41} & \cdots & K^1_{44} \end{pmatrix}$$

The rows in C tell us where the entries K^1_{ij} go in the global 8×8 stiffness matrix K.

$K^1_{11} \to K_{44}$ slot of K
$K^1_{12} \to K_{41}$
$K^1_{13} \to K_{42}$
$K^1_{14} \to K_{45}$
$K^1_{21} \to K_{14}$ (just relabels local to global node n°s)
(see diagram opposite)

Algorithm for assembly

Let $K = \underline{0}$.
for $e = 1:N$ (N = # elements)
for $i = 1:n$ (n = # nodes per element) assumed const.
for $j = i:n$ (by symmetry of K)
find global nodes p, q
corresponding to local nodes i, j
using C.

compute/retrieve K_{ij} and add
this to K_{pq} (and K_{qp} if $p \neq q$)
end
end
end

[L] Note on accuracy and efficiency of the FE method

1D: $T = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3 + \cdots + \alpha_n x^{n-1}$

for an n-node element

Taylor expansion of real T on element $[x_i, x_{i+1}]$ about x_i:

$$T = T(x_i) + \left.\frac{\partial T}{\partial x}\right|_{x=x_i}(x-x_i) + \frac{1}{2}\left.\frac{\partial^2 T}{\partial x^2}\right|_{x=x_i}(x-x_i)^2 + \cdots + \frac{1}{(n+1)!}\left.\frac{\partial^{n+1} T}{\partial x^{n+1}}\right|_{x=\xi}(x-x_i)^{n+1}$$

$\xi \in [x_i, x_{i+1}]$

Conclude: The error we're making is $O(h^{p+1})$ if we include all terms up to order p, and h is the size of the element. ($h = x_i - x_j$)

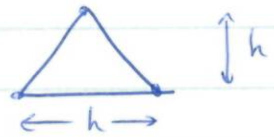
2D: $T = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \cdots$

Taylor expansion of real T on element about (x_i, y_i):

$$T = T(x_i, y_i) + \left.\frac{\partial T}{\partial x}\right|_{(x_i,y_i)}(x-x_i) + \left.\frac{\partial T}{\partial y}\right|_{(x_i,y_i)}(y-y_i) + \frac{1}{2}\left.\frac{\partial^2 T}{\partial x^2}\right|_{(x_i,y_i)}(x-x_i)^2 + \frac{1}{2}\left.\frac{\partial^2 T}{\partial y \partial x}\right|_{(x_i,y_i)}(x-x_i)(y-y_i) + \cdots$$

<u>Conclude:</u> The error we're making is $O(h^{p+1})$ if all terms of order p are included, & h is a measure for the size of the element

(h is really about <u>order</u>, not too bothered about exact value)



<u>Triangular elements</u>

3-node  $T = \alpha_1 + \alpha_2 x + \alpha_3 y$ error $= O(h^2)$

6-node  $T = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 y^2$ error $= O(h^3)$

These elements are called <u>order-complete</u> (all terms up to certain order are included)

Pascal's triangle:

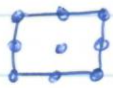
$$
\begin{array}{ccccccccc}
 & & & & 1 & & & & \\
 & & & x & & y & & & \\
 & & x^2 & & xy & & y^2 & & \\
 & x^3 & & x^2y & & xy^2 & & y^3 & \\
x^4 & & x^3y & & x^2y^2 & & x^1y^3 & & y^4
\end{array}
$$

you need a 10-node element to be order complete up to x^3.

<u>Rectangular elements</u>

4-node  $T = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 xy$ error $= O(h^2)$

$\because$ not all quadratic terms are included

9-node  cannot contain all terms up
(Lagrange) to cubic order $\Rightarrow$ error $= O(h^3)$

note central node is useless
(1) It doesn't increase accuracy (error still $O(h^3)$)
(2) doesn't help conformity (i.e. for guaranteeing continuity across element boundaries) because it's not on the boundary

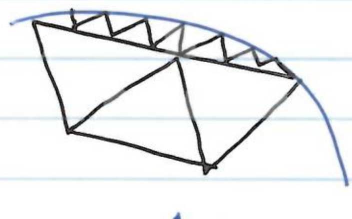
it only increases the dimension of the problem. Such a node is called <u>parasitic</u> and hence it is often left out and the 8-node serendipity element is considered instead

<u>Conclude</u>: There are two ways of increasing accuracy:

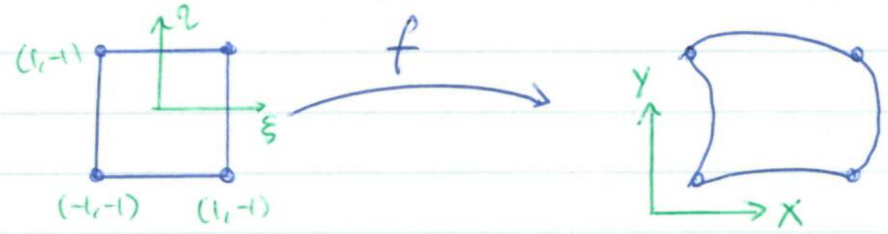
(1) <u>h-method</u>: reduce h, ie finer mesh, taking smaller elements

(2) <u>p-method</u>: introduce more nodes per element and construct higher-order shape functions (increasing p)

<u>Isoparametric elements</u>



To deal with curved boundaries of the domain we could
(1) refine the mesh (at computational cost)
(2) construct elements with curved edges (Isoparametric elements)



parent or natural element

distorted or physical element

We require that f is:

(1) continuous

(2) locally one-to-one, i.e. Jacobian must be non-singular

$$X = X(\xi, \eta)$$
$$Y = Y(\xi, \eta)$$

$$|J| = \begin{vmatrix} \partial X/\partial\xi & \partial X/\partial\eta \\ \partial Y/\partial\xi & \partial Y/\partial\eta \end{vmatrix} \neq 0$$

(3) globally one-to-one
(usually OK if element not too distorted, and we'll assume it to be OK)

How to choose this mapping??

$\Downarrow$

use SHAPE FUNCTIONS

(hence the name <u>iso</u>parametric: same functions used for interpolation and transformation).

4-node isoparametric quadrilateral element

$$N_1^e = \frac{1}{4}(\xi - 1)(\eta - 1)$$
$$N_2^e = -\frac{1}{4}(\xi + 1)(\eta - 1)$$
$$N_3^e = \frac{1}{4}(\xi + 1)(\eta + 1)$$
$$N_4^e = -\frac{1}{4}(\xi - 1)(\eta + 1)$$

these are special cases of the 4-node rectangular element

$$N_i(x_j, y_j) = \delta_{ij}$$

Transformation f:

$$X = X(\xi, \eta) = \sum_{i=1}^{4} N_i^e(\xi, \eta) X_i$$

$$Y = Y(\xi, \eta) = \sum_{i=1}^{4} N_i^e(\xi, \eta) Y_i$$

Consider node 1: $\left.\begin{array}{l} X(-1,1) = X_1 \\ Y(-1,-1) = Y_1 \end{array}\right\}$ and same for other nodes

$\Downarrow$

maps nodes onto nodes

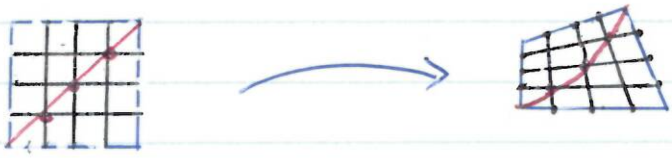
Consider edge $\xi = -1$. $X = X(-1, \eta) = -\frac{1}{2}(\eta - 1) X_1 + \frac{1}{2}(\eta + 1) X_4$

$Y = Y(-1, \eta) = -\frac{1}{2}(\eta - 1) Y_1 + \frac{1}{2}(\eta + 1) Y_4$

Eliminating η gives a linear relationship between X and Y $(Y = aX + b)$.

and same for other edges $\Longrightarrow$ edges map onto edges

Warning: horizontal + vertical lines are mapped onto straight lines but diagonal lines are <u>not</u>.



this is because shape f^n's are <u>bilinear</u> but not linear.

The isoparametric element is conforming because the square parent element is. (It has its edges parallel to the ξ and η axes). The mapping preserves this by continuity.

In particular this solves the problem that nonaligned rectangular elements are not conforming

Now (with these isoparametric shape functions) a mesh of such elements <u>is</u> conforming

<u>Question</u>: What about the uniqueness of this mapping?

$$X = N_1^e X_1 + N_2^e X_2 + N_3^e X_3 + N_4^e X_4$$
$$Y = N_1^e Y_1 + N_2^e Y_2 + N_3^e Y_3 + N_4^e(\xi,\eta) Y_4$$

$$\rightarrow X = \tfrac{1}{4}\left[(-\xi-\eta+1+\xi\eta)X_1 + (\xi-\eta+1-\xi\eta)X_2\right]$$
$$Y = \tfrac{1}{4}\left[\cdots\right]$$

$$J = \begin{pmatrix} \partial X/\partial\xi & \partial X/\partial\eta \\ \partial Y/\partial\xi & \partial Y/\partial\eta \end{pmatrix} \Rightarrow \det J = c_1 + c_2\xi + c_3\eta$$

note quadratic terms cancel

<u>Consequence</u>: If det J has the same sign at all 4 nodes then det J cannot vanish on the element $\Rightarrow$ <u>uniqueness</u>!

At node 1 we have $\xi = -1, \; \eta = -1$

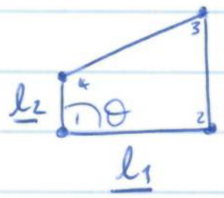
$$\frac{\partial X}{\partial \xi} = \frac{1}{4}(-X_1 + X_2 + X_3 + X_4) + \frac{1}{4}(X_1 - X_2 + X_3 + X_4)\eta$$

$$\left.\begin{matrix}\eta=-1\\ \xi=-1\end{matrix}\right\} \Rightarrow \quad = \frac{1}{2}(X_2 - X_1)$$

Similarly $\dfrac{\partial Y}{\partial \eta} = \dfrac{1}{2}(Y_4 - Y_1)$

$$\det J = \frac{1}{4}\left[(X_2-X_1)(Y_4-Y_1) - (Y_2-Y_1)(X_4-X_1)\right]$$

Geometric interpretation:



$$\underline{l}_1 = \begin{pmatrix} X_2 - X_1 \\ Y_2 - Y_1 \end{pmatrix}$$

$$\underline{l}_2 = \begin{pmatrix} X_4 - X_1 \\ Y_4 - Y_1 \end{pmatrix}$$

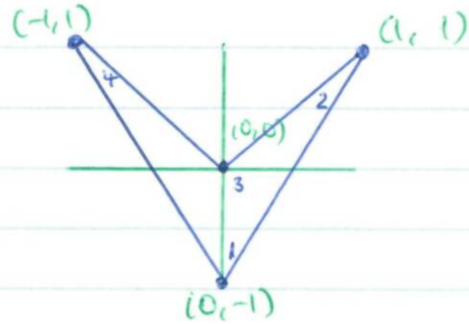
Then $\det J = \frac{1}{4}\, \underline{l}_1 \times \underline{l}_2$

$$= \frac{1}{4}|\underline{l}_1||\underline{l}_2| \sin\theta \quad > 0 \quad \text{if } \theta < \pi.$$

<u>Conclude</u>:

det J is nonvanishing on the element $\Updownarrow$ the quadrilateral is convex, i.e. if all the angles are less than π.

Example of a singular Jacobian



Isoparametric transformation:

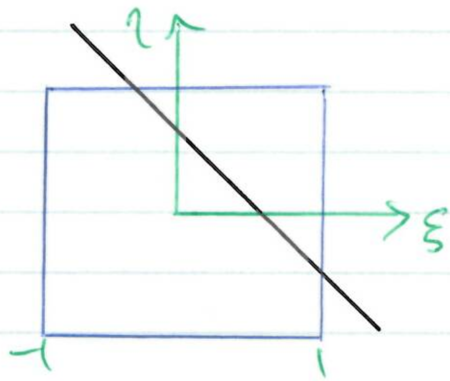
$$X = \sum_{i=1}^{N} N_i^e(\xi, \eta) X_i$$

$$Y = \sum_{i=1}^{N} N_i^e(\xi, \eta) Y_i$$

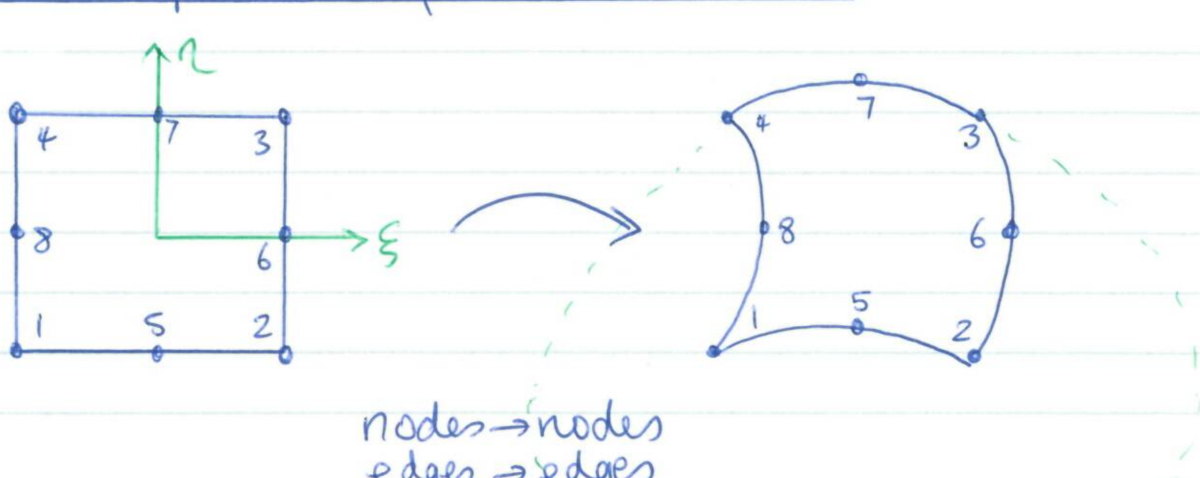
$$J = \begin{pmatrix} \partial X/\partial \xi & \partial X/\partial \eta \\ \partial Y/\partial \xi & \partial Y/\partial \eta \end{pmatrix} = \begin{pmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{4}(1-3\eta) & \frac{1}{4}(1-3\xi) \end{pmatrix}$$

$\Rightarrow \det J = \frac{1}{8}(2 - 3\xi - 3\eta)$ ⟵ this changes sign on $-1 \leq \xi, \eta \leq 1$

namely on $\eta = -\xi + \frac{2}{3}$



8-node isoparametric quadrilateral element



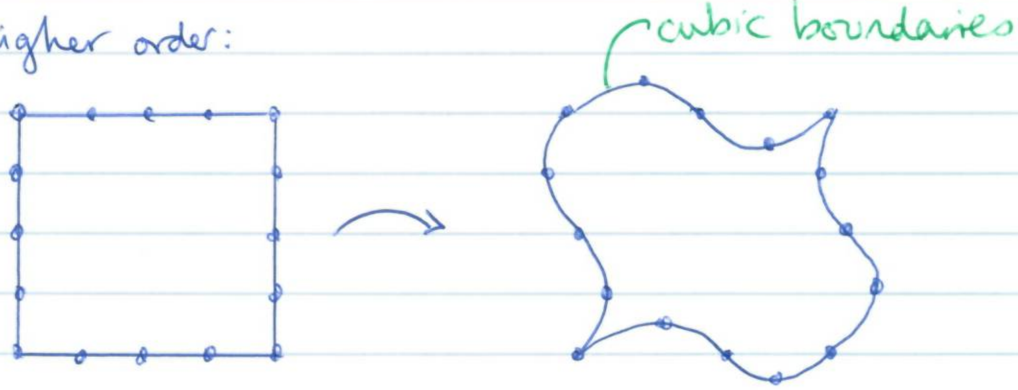
nodes → nodes
edges → edges
(straight) (quadratic)

Shape functions:

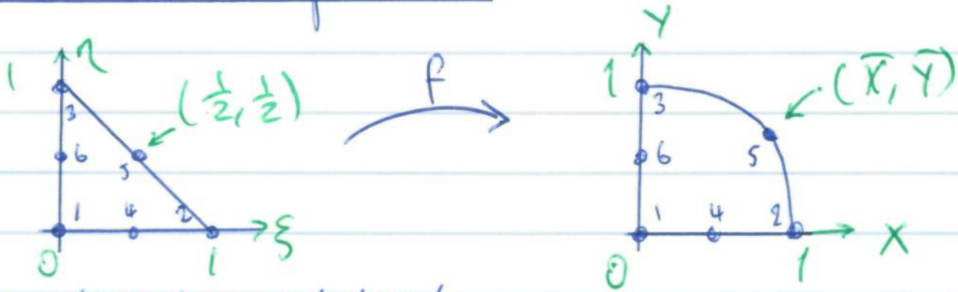
$$N_1^e = -\tfrac{1}{4}(1-\xi)(1-\eta)(1+\xi+\eta) \qquad \text{biquadratic.}$$

We can use this 8-node element to map quadratic boundary exactly:

higher order:



Restriction on node placement



6-node triangular parent element

$$f: \quad \begin{aligned} X &= \xi + (4\bar{X} - 2)\xi\eta \qquad \left(= \sum N_i^e X_i\right) \\ Y &= \eta + (4\bar{Y} - 2)\xi\eta \qquad \left(= \sum N_i^e Y_i\right) \end{aligned}$$

note: $X(\tfrac{1}{2}, \tfrac{1}{2}) = \bar{X}$
$Y(\tfrac{1}{2}, \tfrac{1}{2}) = \bar{Y}$

$$\det J = \begin{vmatrix} 1 + (4\bar{X}-2)\eta & (4\bar{X}-2)\xi \\ (4\bar{Y}-2)\eta & 1+(4\bar{Y}-2)\xi \end{vmatrix}$$

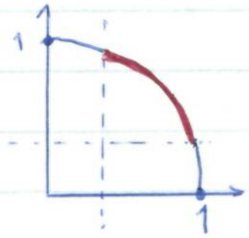
$$= 1 + (4\bar{X}-2)\eta + (4\bar{Y}-2)\xi \qquad \text{linear!}$$

at node 1: $\xi = 0,\ \eta = 0 \Rightarrow \det J = 1$

at node 2: $(1,0)$: $4\bar{Y}-2 > -1 \Rightarrow \bar{Y} > 1/4$

at node 3: $(0,1)$: $4\bar{X}-2 > -1 \Rightarrow \bar{X} > 1/4$

node needs to be placed in sector for well-behaved map



$\boxed{L}$ Transformation of the stiffness matrix

$$K_{ij} = \iint \left[\frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y}\frac{\partial N_j}{\partial y} \right] dx\,dy$$

Now, $N_i = N_i(\xi, \eta)$ so we need to transform $(x,y) \longrightarrow (\xi, \eta)$

[comes from $\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} = c$ (Poisson's eq$^{\underline{n}}$)]

More generally,
$$a_{11}\frac{\partial^2 \phi}{\partial x^2} + 2a_{12}\frac{\partial^2 \phi}{\partial x \partial y} + a_{22}\frac{\partial^2 \phi}{\partial y^2}$$

weak formulation
$$\Longrightarrow \qquad K_{ij} = \iint \left(\frac{\partial N_i}{\partial x}, \frac{\partial N_i}{\partial y} \right) D \begin{pmatrix} \frac{\partial N_j}{\partial x} \\ \frac{\partial N_j}{\partial y} \end{pmatrix} dx\,dy$$

where $D = \begin{pmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{pmatrix}$ $\qquad D = D(x,y)$

Jacobian of isoparametric transformation

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix}$$

$$\frac{\partial N_i}{\partial \xi} = \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial N_i}{\partial y}\frac{\partial y}{\partial \xi}$$

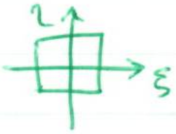
$$\frac{\partial N_i}{\partial \eta} = \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial N_i}{\partial y}\frac{\partial y}{\partial \eta}$$

$$\begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix} = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{pmatrix} \begin{pmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{pmatrix} = J^T \begin{pmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{pmatrix}$$

Invert: $$\begin{pmatrix} \frac{\partial N_i}{\partial x} \\ \frac{\partial N_i}{\partial y} \end{pmatrix} = (J^T)^{-1} \begin{pmatrix} \frac{\partial N_i}{\partial \xi} \\ \frac{\partial N_i}{\partial \eta} \end{pmatrix}$$

$$\Rightarrow K_{ij} = \int_{-1}^{1}\int_{-1}^{1} \left(\frac{\partial N_i}{\partial \xi}, \frac{\partial N_i}{\partial \eta}\right) J^{-1} D (J^T)^{-1} \begin{pmatrix} \frac{\partial N_j}{\partial \xi} \\ \frac{\partial N_j}{\partial \eta} \end{pmatrix} |\det J| \, d\xi \, d\eta$$

assuming a square



Transformation of the load vector

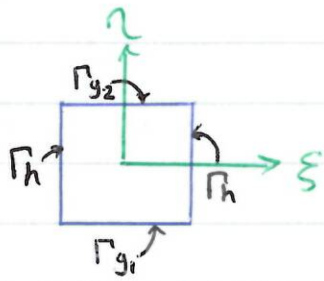
$$f_{e_i} = \iint_e N_i(\xi,\eta)\, Q\left(x(\xi,\eta),\, y(\xi,\eta)\right) |\det J| \; d\xi\, d\eta$$

Transformation of the boundary vector

[for Poisson: $f_{b_i} = \int_\Gamma N_i \frac{d\phi}{dn}\, ds$]

$$dx = \frac{\partial x}{\partial \xi} d\xi + \frac{\partial x}{\partial \eta} d\eta$$

Consider square element



along Γ_g : $d\eta = 0 \Rightarrow dx = \frac{\partial x}{\partial \xi} d\xi$

$$dy = \frac{\partial y}{\partial \xi} d\xi$$

$$\Rightarrow \int_{\Gamma_{g_1}} f(x,y)\, ds = \int_{-1}^{1} f(x(\xi,\eta), y(\xi,\eta)) \left[\left(\frac{\partial x}{\partial \xi}\right)^2 + \left(\frac{\partial y}{\partial \xi}\right)^2 \right]^{1/2} d\xi$$

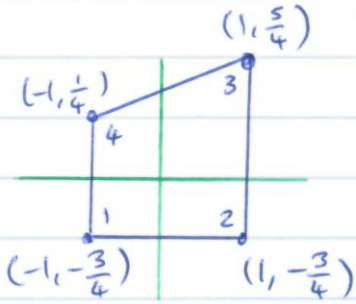
along Γ_h :

$$\int_{\Gamma_h} f(x,y)\, ds = \int_{-1}^{1} f(x(\xi,\eta), y(\xi,\eta)) \left[\left(\frac{\partial x}{\partial \eta}\right)^2 + \left(\frac{\partial y}{\partial \eta}\right)^2 \right]^{1/2} d\eta$$

"The calculations are straightforward but you need to know what you're doing"

Example:

Q: Compute stiffness matrix for Laplacian $\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)$ on this element



$$N_1 = \frac{1}{4}(\xi-1)(\eta-1)$$

Transformation:

$$x = \sum_{i=1}^{4} N_i x_i \qquad y = \sum_{i=1}^{4} N_i(\xi,\eta)\, y_i$$

$$J = \begin{pmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{1}{4}(\eta+1) & \frac{1}{4}(\xi+3) \end{pmatrix}$$

$$\Rightarrow \det J = \frac{1}{4}(\xi+3)$$

$$J^{-1} = \frac{4}{\xi+3}\begin{pmatrix} \frac{1}{4}(\xi+3) & 0 \\ -\frac{1}{4}(\eta+1) & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ \frac{-\eta+1}{\xi+3} & \frac{4}{\xi+3} \end{pmatrix}$$

$$\Rightarrow K_{11} = \iint_{\square} \left(\frac{\partial N_1}{\partial x}, \frac{\partial N}{\partial y}\right) \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}}_{D} \begin{pmatrix} \frac{\partial N_1}{\partial x} \\ \frac{\partial N_1}{\partial y} \end{pmatrix} dx\,dy$$

D ← from Laplacian

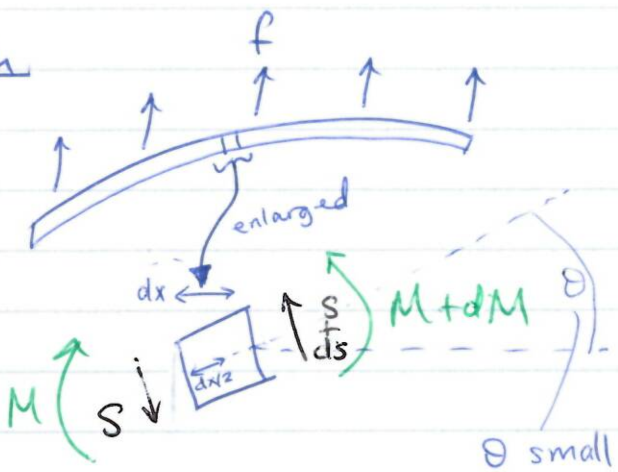
$$= \int_{-1}^{1}\int_{-1}^{1} \left(\frac{\partial N_1}{\partial \xi}, \frac{\partial N_1}{\partial \eta}\right) J^{-1} D (J^T)^{-1} \begin{pmatrix} \frac{\partial N_1}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} \end{pmatrix} |\det J|\, d\xi\, d\eta$$

$$= \int_{-1}^{1}\int_{-1}^{1} \left[\left[\frac{1}{4}(\eta-1) - \frac{(\eta+1)(\xi-1)}{4(\xi+3)} \right]^2 + \left[\frac{\xi-1}{\xi+3} \right]^2 \right] \frac{\xi+3}{4} \, d\xi \, d\eta$$

polynomial in η, then rational in ξ.

FOURTH ORDER EQUATIONS

Euler-Bernoulli beam equation



M moment

S shear

assume no shear deformation

force balance in vertical direction:

$$(S+dS)\cos\theta - S\cos\theta + f\,dx = 0$$

force per unit length

$$\Rightarrow \boxed{\frac{dS}{dx} + f = 0} \qquad \text{if } \theta \text{ is small}$$

Moment balance (about axis through centre of mass)

$$(M+dM) - M + \underbrace{S\frac{dx}{2}}_{\text{Force} \times \text{dist}} + (S+dS)\frac{dx}{2} = 0$$

$$\Rightarrow \boxed{\frac{dM}{dx} + S = 0}$$

$$\Rightarrow \boxed{\frac{d^2M}{dx^2} = f}$$

Constitutive relation:

$$M = EI\frac{d^2y}{dx^2}$$

curvature

bending

Young's modulus

area moment of inertia

Conclusion: $$\frac{d^2}{dx^2}\left(EI\,\frac{d^2y}{dx^2}\right) = f$$

Euler-Bernoulli Beam Equation
statics case

Possible b.c.s:

(1) 

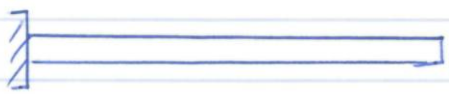
"Simply $\left\{\begin{array}{l}\text{supported}\\ \text{hinged}\\ \text{pinned}\end{array}\right.$"

$y(0) = 0$
$y(L) = 0$

$\frac{d^2y}{dx^2}(0) = 0$

(no bending moment)

$\frac{d^2y}{dx^2}(L) = 0$

(2) 

"cantilever beam"

fixed/clamped/welded at $x=0$

free at $x=L$

$y(0) = 0$
$\frac{dy}{dx}(0) = 0$

$\frac{d^2y}{dx^2}(L) = 0$ (no bending moment at end)

[Shear force $S = -\frac{dM}{dx}$]

$$\frac{d^3y}{dx^3}(L) = 0 \qquad \left(\text{no force} \cdot \frac{d^3}{dx^3} \because S = -\frac{dM}{dx}\right).$$

Weak formulation

$$\int_0^L \left(\frac{d^2M}{dx^2} v - fv \right) dx = 0 \qquad (v \text{ test } f^n)$$

by parts

$$-\int_0^L \left[\frac{dM}{dx}\frac{dv}{dx} + fv \right] dx + \left[v \frac{dM}{dx} \right]_0^L = 0$$

by parts

$$\int_0^L \left[M \frac{d^2v}{dx^2} - fv \right] dx + \left[v \underbrace{\frac{dM}{dx}}_{-S} - \frac{dv}{dx} M \right]_0^L = 0$$

Interpolation: $y = \sum_j N_j(x) u_j$

Galerkin: $v = N_i$

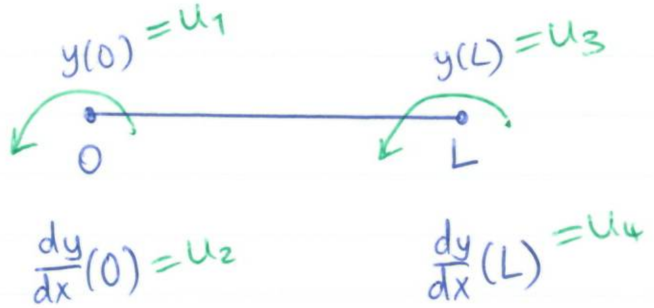
$\Rightarrow$ FE equation becomes

$$\sum_j \int_0^L \frac{d^2N_i}{dx^2} EI \frac{d^2N_j}{dx^2} \, dx \, u_j$$

$$= \int_0^L fN_i \, dx + \left[\frac{dN_i}{dx} M \right]_0^L + \left[N_i S \right]_0^L$$

This makes sense if we require $\frac{dN_i}{dx}$ to be continuous and piecewise differentiable.

Idea: interpolate derivatives as well as the function itself



Two degrees of freedom per node
- displacement y
- slope $\frac{dy}{dx}$

Approximation: $y(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3$ (4 data points)

For consistency: $y(0) = u_1$ $\quad y(L) = u_3$

$\frac{dy}{dx}(0) = u_2$ $\quad \frac{dy}{dx}(L) = u_4$

4 equations for α_i, $i = 1, \dots, 4$.

$$y(x) = \sum_{i=1}^{4} N_i(x) u_i.$$

Shape f^ns:

$$N_1^e = 1 - \frac{3x^2}{L^2} + \frac{2x^3}{L^3}$$

$$N_2^e = x - \frac{2x^2}{L} + \frac{x^3}{L^2}$$

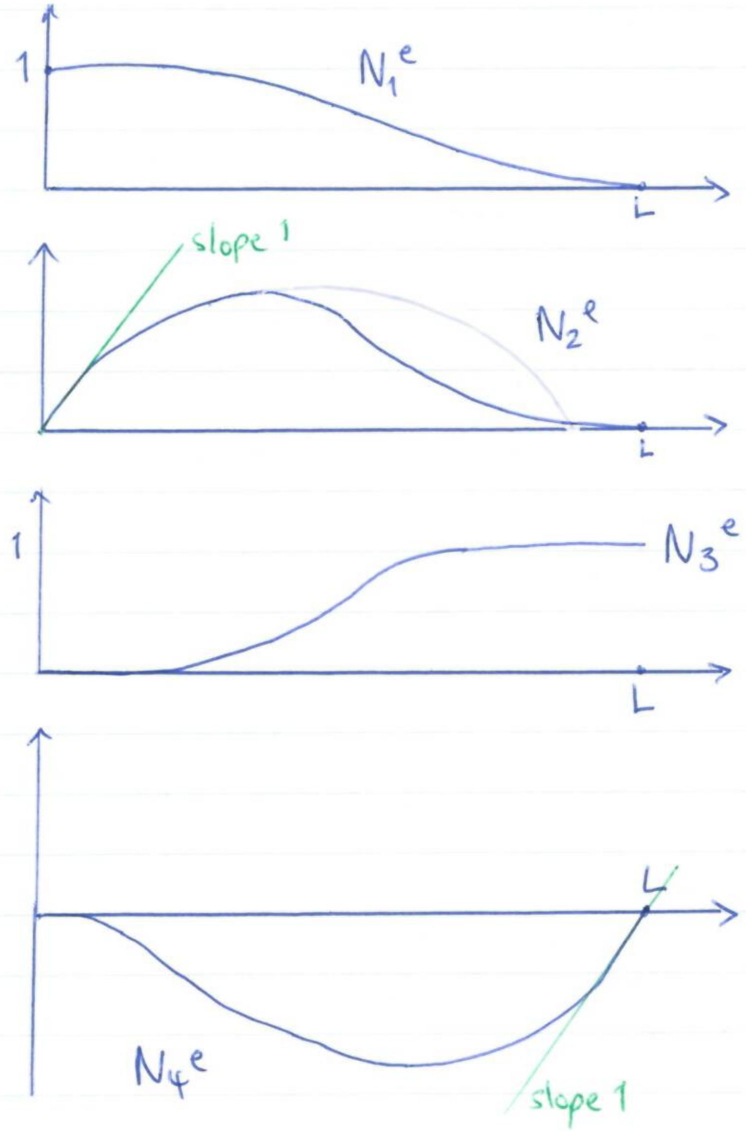
$$N_3^e = \frac{3x^2}{L^2} - \frac{2x^3}{L^3}$$

$$N_4^e = -\frac{x^2}{L} + \frac{x^3}{L^2}$$

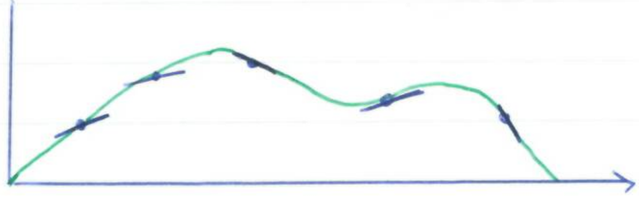
Note that $N_{odd}^e(x_j) = \delta_{ij}$ where $N_{odd}^e = N_{2i-1}^e$

$\frac{dN_{even}^e}{dx}(x_j) = \delta_{ij}$ where $N_{even}^e = N_{2i}^e$

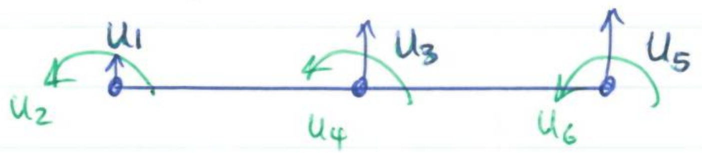
Graphically,



Polynomial approximations of f^ns that fulfil not only the f^n values at given pts but also the slopes at these pts are called <u>Hermite polynomials</u>.



Higher-order shape f^{n}s



$$y(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \alpha_4 x^3 + \alpha_5 x^4 + \alpha_6 x^5$$

Evaluation of the element data (for uniform beam, ie $EI = \text{const}$, $f = \text{const}$)

Stiffness matrix

$$K_{ij} = \int_0^L \frac{d^2 N_i}{dx^2} EI \frac{d^2 N_j}{dx^2} \, dx$$

$$K_{12} = EI \int_0^L \left(-\frac{6}{L^2} + \frac{12x}{L^3} \right) \left(-\frac{4}{L} + \frac{6x}{L^2} \right) dx$$

$$= \frac{6EI}{L^2}$$

etc

to find
$$K = \frac{EI}{L^3} \begin{pmatrix} 12 & 6L & -12 & 6L \\ 6L & 4L^2 & -6L & 2L^2 \\ -12 & -6L & 12 & -6L \\ 6L & 2L^2 & -6L & 4L^2 \end{pmatrix}$$

Load vector

$$f_{\ell_i} = \int_0^L f N_i \, dx$$

to find
$$f_\ell = \frac{fL}{12} \begin{pmatrix} 6 \\ L \\ 6 \\ -L \end{pmatrix}$$

boundary vector

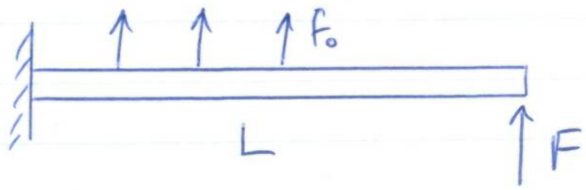
$$f_{b_i} = \left[N_i S \right]_0^L + \left[\frac{dN_i}{dx} M \right]_0^L$$

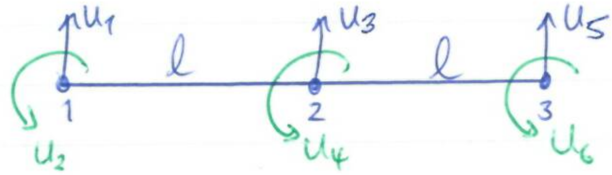
to find

$$f_b = \begin{pmatrix} -S(0) \\ -M(0) \\ S(L) \\ M(L) \end{pmatrix}$$

Lots of these shape f^ns are 0 or have 0 derivative at x= 0 or L.

Example (2 element model for a cantilever beam with an end force)





$2l = L$

BCs: $y(0) = 0$
$y'(0) = 0$
$M(L) = 0$
$S(L) = F$

$$K^1 = K^2 = \frac{EI}{l^3} \begin{pmatrix} 12 & 6l & \cdots \\ 6l & 4l^2 & \\ \vdots & & \ddots \end{pmatrix}$$

Assembly

Connectivity matrix

$$C = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 5 & 6 \end{pmatrix} \begin{matrix} \leftarrow e=1 \\ \leftarrow e=2 \end{matrix}$$

Note: we count the degrees of freedom rather than the nodes.

$$K = \frac{EI}{l^3}\begin{pmatrix} 12 & 6l & -12 & 6l & 0 & 0 \\ 6l & 4l^2 & -6l & 2l^2 & 0 & 0 \\ -12 & -6l^2 & 12+12 & -6l+6l & -12 & 6l \\ 6l & 2l^2 & -6l+6l & 4l^2+4l^2 & -6l & 2l^2 \\ 0 & 0 & -12 & -6l & 12 & -6l \\ 0 & 0 & 6l & 2l^2 & -6l & 4l^2 \end{pmatrix}$$

global stiffness matrix

load vector: $f_l^1 = f_l^2 = \frac{fl}{12}\begin{pmatrix} 6 \\ l \\ 6 \\ -l \end{pmatrix}$

$$\rightarrow f_l = \begin{pmatrix} 6 \\ l \\ 6+6 \\ -l+l \\ 6 \\ l \end{pmatrix}$$

boundary vector:

$$f_b = \begin{pmatrix} -S(0) \\ -M(0) \\ 0+0 \\ 0+0 \\ S(L) \\ M(L) \end{pmatrix}$$

makes sense

node 2 is not on the bdry.

Solving the FE eq^{n}

$$Ku = f_e + f_b \quad \text{ie}$$

$$\frac{EI}{l^3}\begin{pmatrix} 12 & 6l & -12 & 6l & 0 & 0 \\ 6l & 4l^2 & -6l & 2l^2 & 0 & 0 \\ -12 & -6l & 24 & 0 & -12 & 6l \\ 6l & 2l^2 & 0 & 8l^2 & -6l & 2l^2 \\ 0 & 0 & -12 & -6l & 12 & -6l \\ 0 & 0 & 6l & 2l^2 & -6l & 4l^2 \end{pmatrix}\begin{pmatrix} u_1 = 0 \\ u_2 = 0 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \end{pmatrix} = \frac{fl}{12}\begin{pmatrix} 6 \\ l \\ 12 \\ 0 \\ 6 \\ -l \end{pmatrix} + \begin{pmatrix} -S_0 \\ -N_0 \\ 0 \\ 0 \\ S(L) = F \\ M(L) = 0 \end{pmatrix}$$

equivalent (by reducing) to 4x4 problem:

$$\frac{EI}{l^3}\begin{pmatrix} 24 & 0 & -12 & 6l \\ 0 & 8l^2 & -6l & 2l^2 \\ -12 & -6l & 12 & -6l \\ 6l & 2l^2 & -6l & 4l^2 \end{pmatrix}\begin{pmatrix} u_3 \\ u_4 \\ u_5 \\ u_6 \end{pmatrix} = \begin{pmatrix} fl \\ 0 \\ \frac{1}{2}fl + F \\ -fl^2/12 \end{pmatrix}$$

Solve to give u_3, u_4, u_5, u_6.

Then the first 2 eq^{n}s (of the 6x6 system) give:

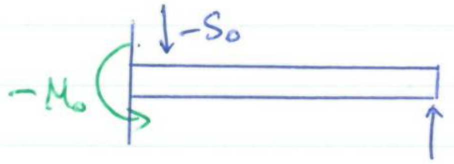
$$\frac{EI}{\ell^3}\left(-12u_3 + 6\ell u_4\right) = \frac{1}{2}f\ell - S_0$$

$$\frac{EI}{\ell^3}\left(-6\ell u_3 + 2\ell^2 u_4\right) = \frac{1}{12}f\ell^2 - M_0$$

$\Rightarrow$ solve to give S_0, M_0

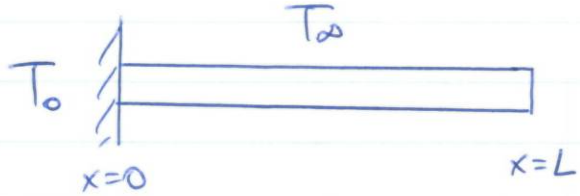
S_0 is the force exerted by the beam on the wall at $x=0$, which is balanced by an equal and opposite reaction force $-S_0$ on the beam.

$-M_0$ is the reaction moment (similarly) on the beam at $x=0$. (the fixed end)



Nonlinear problems

Consider radiative heat transfer in a fin, which goes as T^4.



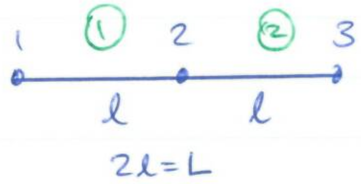
$$Ak\frac{d^2T}{dx} - A_s\sigma\left(T^4 - T_\infty^4\right) + Q = 0$$

$$T_0 = T(0).$$

$$q(L) = -k\frac{dT}{dx}(L) = \sigma\left(T^4(L) - T_\infty^4\right)$$

where σ is the Stefan-Boltzmann constant
T_∞ is the ambient reference temperature
A_s is the total lateral surface area

Take mesh of two 2-node elements.



Weak formulation:

$$\int_0^L \left(-\frac{dT}{dx} Ak \frac{dv}{dx} - A_s \sigma T^4 v + \bar{Q} v \right) dx - \left[Aqv \right]_0^L = 0$$

where $\bar{Q} = Q + A_s \sigma T_\infty^4$.

For single element:

$$T = \sum_j N_j T_j \,, \qquad v = N_i$$

$$\Rightarrow \text{FE eq}^n: \quad \left[-AqN_i \right]_0^\ell - \sum_j \int_0^\ell Ak \frac{dN_i}{dx} \frac{dN_j}{dx} T_j \, dx$$
$$+ \int_0^\ell \bar{Q} N_i \, dx - \int_0^\ell A_s \sigma N_i \left(\sum_j N_j T_j \right)^4 dx = 0.$$

$$\left. \begin{array}{l} N_1 = 1 - \frac{x}{\ell} \\ N_2 = \frac{x}{\ell} \end{array} \right\} \Rightarrow \quad K = \frac{Ak}{\ell} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

$$f_\ell = \frac{\bar{Q}\ell}{2} \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Thus:

$$-\frac{Ak}{\ell}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} T_1 \\ T_2 \end{pmatrix} + \frac{\bar{Q}\ell}{2}\begin{pmatrix} 1 \\ 1 \end{pmatrix} - A_s\sigma\begin{pmatrix} \int_0^\ell N_1 (N_1T_1 + N_2T_2)^4\, dx \\ \int_0^\ell N_2 (N_1T_1 + N_2T_2)^4\, dx \end{pmatrix}$$

$$+ \begin{pmatrix} [-AqN_1]_0^\ell \\ [-AqN_2]_0^\ell \end{pmatrix} = 0$$

Extra nonlinear term

$$-A_s\sigma\begin{pmatrix} \int N_1^5 & 4\int N_1^4N_2 & 6\int N_1^3N_2^2 & 4\int N_1^2N_2^3 & \int N_1N_2^4 \\ \int N_1^4N_2 & 4\int N_1^3N_2^2 & 6\int N_1^2N_2^3 & 4\int N_1N_2^4 & \int N_2^5 \end{pmatrix}\begin{pmatrix} T_1^4 \\ T_1^3T_2 \\ T_1^2T_2^2 \\ T_1T_2^3 \\ T_2^4 \end{pmatrix}$$

Use integration formula $\int_0^\ell N_1^n N_2^m\, dx = \frac{n!\,m!}{(n+m+1)!}\ell$

$$= -\ell A_s\sigma\begin{pmatrix} \frac{1}{6} & \frac{2}{15} & \frac{1}{10} & \frac{1}{15} & \frac{1}{30} \\ \frac{1}{30} & \frac{1}{15} & \frac{1}{10} & \frac{2}{15} & \frac{1}{6} \end{pmatrix}\begin{pmatrix} T_1^4 \\ T_1^3T_2 \\ T_1^2T_2^2 \\ T_1T_2^3 \\ T_2^4 \end{pmatrix}$$

Assembly

$$\frac{4k}{\ell}\begin{pmatrix} 1 & -1 & 0 \\ -1 & 1+1 & -1 \\ 0 & -1 & 1 \end{pmatrix}\begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix}$$

$$+\ell A_s \sigma \begin{pmatrix} \frac{1}{6} & \frac{2}{15} & \frac{1}{10} & \frac{1}{15} & \frac{1}{30} & 0 & 0 & 0 & 0 \\ \frac{1}{30} & \frac{1}{15} & \frac{1}{10} & \frac{2}{5} & \frac{1}{6}+\frac{1}{6} & \frac{2}{5} & \frac{1}{10} & \frac{1}{15} & \frac{1}{30} \\ 0 & 0 & 0 & 0 & \frac{1}{30} & \frac{1}{15} & \frac{1}{10} & \frac{2}{15} & \frac{1}{6} \end{pmatrix}\begin{pmatrix} T_1^4 \\ T_1^3 T_2 \\ T_1^2 T_2^2 \\ T_1 T_2^3 \\ T_2^4 \\ T_2^3 T_3 \\ T_2^2 T_3^2 \\ T_2 T_3^3 \\ T_3^4 \end{pmatrix}$$

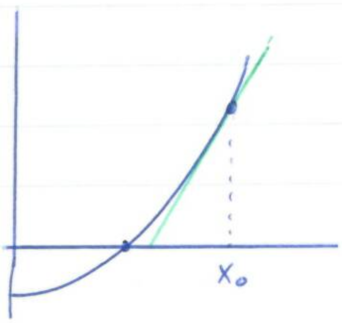
$$= \frac{\bar{Q}\ell}{2}\begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} + \begin{pmatrix} Aq(0) \\ 0 \\ -Aq(L) \end{pmatrix}$$

b.c.s $\quad T_1 = T_0$

$$q(L) = \sigma(T_3^4 - T_a^4)$$

Strategy: Solve 2nd + 3rd eq^n for T_2, T_3 then the first eq^n for $q(0)$.

Can solve nonlinear system of algebraic eq^{n}s using Newton-Raphson.



tangent at stage k of the iteration is determined by the slope $f'(x_k)$ and the point $(x_k, f(x_k))$.

$$y - f(x_k) = f'(x_k)(x - x_k)$$

x_{k+1} is then defined as that x s.t. $y = 0$.

$$0 - f(x_k) = f'(x_k)(x_{k+1} - x_k)$$

Solve for x_{k+1}:

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

It can be shown that, if $\bar{x}$ is the exact sol^{n},

$$|\bar{x} - x_{k+1}| \leq \alpha |\bar{x} - x_k|^2$$

where α is a constant independent of k, so the convergence is quadratic

$\Rightarrow$ in each step, you double the n° of correct digits.

14 Time-dependent problems

Heat flow: $\rho c A \frac{\partial T}{\partial t} = \frac{\partial}{\partial x}\left(Ak\frac{\partial T}{\partial x}\right) - \underbrace{\alpha(T - T_a)}_{\text{convective term}} + Q$

ρ mass density
c specific heat
T_a ambient temperature
A area of cross-section

Weak formulation:

$$\int_0^L \left[-Ak\frac{\partial T}{\partial x}\frac{\partial v}{\partial x} - \alpha T v + \bar{Q} v - \rho c A \frac{\partial T}{\partial t} v\right] dx + \left[Ak\frac{\partial T}{\partial x} v\right]_0^L = 0$$

$T = \sum_j N_j T_j$, $\quad v = N_i \quad$ (Galerkin) $\qquad \bar{Q} = Q + \alpha T_a$

2-node element.

FE eq$^{\underline{n}}$:

$$-Ak\sum_j \int_0^L \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\, dx\, T_j - \alpha \sum_j \int_0^L N_i N_j\, dx\, T_j$$
$$+ \bar{Q}\int_0^L N_i\, dx - \rho c A \sum_j N_i N_j\, dx\, \dot{T}_j$$
$$+ \left[-A q N_i\right]_0^L = 0$$

$$Ak\int_0^L \frac{\partial N_i}{\partial x}\frac{\partial N_j}{\partial x}\, dx \longrightarrow \frac{Ak}{L}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

$$\int_0^L N_i N_j\, dx \longrightarrow \begin{pmatrix} \int N_1^2 & \int N_1 N_2 \\ \int N_1 N_2 & \int N_2^2 \end{pmatrix} = \frac{L}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

$$\bar{Q}\int_0^L N_i\,dx \longrightarrow \frac{\bar{Q}L}{2}\begin{pmatrix}1\\1\end{pmatrix}$$

Thus:

$$\frac{\rho c A L}{6}\begin{pmatrix}2 & 1\\1 & 2\end{pmatrix}\begin{pmatrix}\dot{T}_1\\\dot{T}_2\end{pmatrix} + \frac{Ak}{L}\begin{pmatrix}1 & -1\\-1 & 1\end{pmatrix}\begin{pmatrix}T_1\\T_2\end{pmatrix}$$

$$+ \frac{\alpha L}{6}\begin{pmatrix}2 & 1\\1 & 2\end{pmatrix}\begin{pmatrix}T_1\\T_2\end{pmatrix} = \frac{QL}{2}\begin{pmatrix}1\\1\end{pmatrix} + \begin{pmatrix}-Aq(0)\\Aq(L)\end{pmatrix}$$

or $\quad M\dot{u} + Ku = f_e + f_b$

where M is the "mass matrix"

$$M = \frac{\rho c A L}{6}\begin{pmatrix}2 & 1\\1 & 2\end{pmatrix}.$$

and K is the stiffness matrix <u>plus</u> the advective term

$$K = \frac{Ak}{L}\begin{pmatrix}1 & -1\\-1 & 1\end{pmatrix} + \frac{\alpha L}{6}\begin{pmatrix}2 & 1\\1 & 2\end{pmatrix}$$

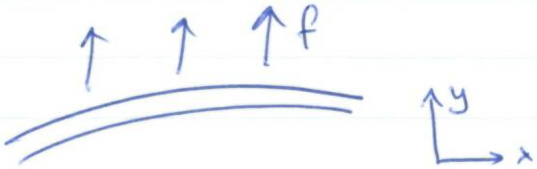
and u contains our unknown temperatures

$$u = (T_1, T_2).$$

Beam dynamics

$$\frac{\partial^2}{\partial x^2}\underbrace{\left(EI\,\frac{\partial^2 y}{\partial x^2}\right)}_{\tilde{M}} + \rho A\,\frac{\partial^2 y}{\partial t^2} = f$$

Euler-Bernoulli beam eqⁿ



Weak formulation (requires 2 integrations by parts):

$$\int_0^L EI\,\frac{\partial^2 y}{\partial x^2}\frac{\partial^2 v}{\partial x^2}\,dx + \int_0^L \rho A\,\frac{\partial^2 y}{\partial t^2}\,dx$$

$$= \int_0^L f v\,dx + \left[\overset{\text{shear force}}{S} v\right]_0^L + \left[\tilde{M}\,\frac{\partial v}{\partial x}\right]_0^L$$

$$y = \sum_j N_j u_j\,, \qquad v = N_i$$

Extra dynamics term:

$$\rho A \sum_j \int_0^L N_i N_j\,dx\;\ddot{u}_j$$

Take single 2-node element

$$N_1 = 1 - \frac{3x^2}{L^2} + \frac{2x^3}{L^3}$$

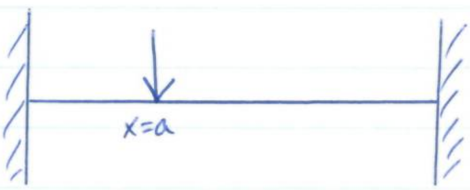
$\Rightarrow$ mass matrix M is

$$M = \frac{\rho A L}{420}\begin{pmatrix} 156 & 22L & 54 & -13L \\ 22L & 4L^2 & 13L & -3L^2 \\ 54 & 13L & 156 & -22L \\ -13L & -3L^2 & -22L & 4L^2 \end{pmatrix}$$

$\Rightarrow$ FE eq^n:

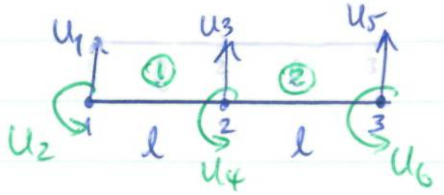
$$M\ddot{u} + Ku = f_e + f_b \qquad \text{(2}^{nd}\text{ order ODE).}$$

Example (periodically forced fixed-fixed beam)



$f(x,t) = P\sin\omega t\, \delta(x-a)$
(assume $a<\ell$, ie f acts on the first element)

Consider a 2-element mesh



$2\ell = L$

$K^1 = K^2, \quad M^1 = M^2$ ← mass matrix

$$f_{e_i}^1 = \int_0^\ell f\, N_i\, dx$$

$$= P\sin\omega t \int_0^\ell \delta(x-a)\, N_i(x)\, dx$$

$$= P N_i(a) \sin\omega t$$

$$f_{l_i}^2 = 0$$

$$K^1 = K^2 = \frac{EI}{l^3}\begin{pmatrix} 12 & 6l & \cdot & \cdot \\ 6l & 4l^2 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{pmatrix}$$

<u>Assembly</u>:

$$\begin{pmatrix} 156 & 22L & 54 & -13L & & \\ 22L & 4L^2 & 13L & -3L^2 & & \\ 54 & 13L & 156+156 & 22L-22L & 54 & -13L \\ -13L & -3L^2 & 22L-22L & 4L^2+4L^2 & 13L & -3L^2 \\ & & 54 & 13L & 156 & -22L \\ & & -13L & -3L^2 & -22L & 4L^2 \end{pmatrix}\begin{pmatrix} u_1 = 0 \\ u_2 = 0 \\ u_3 \\ u_4 \\ u_5 = 0 \\ u_6 = 0 \end{pmatrix}$$

$$= P\sin\omega t\begin{pmatrix} N_1(a) \\ N_2(a) \\ N_3(a) \\ N_4(a) \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -S(0) \\ -M(0) \\ 0 \\ 0 \\ S(L) \\ M(L) \end{pmatrix}$$

<u>BCs</u>: fixed-fixed : $u_1 = 0 = u_2$
$u_5 = 0 = u_6$ (green above)

allows us to cross out first and last 2 rows + columns of the matrix :

Reduced eq^{n}:

$$M = \frac{\rho A \ell}{420}\begin{pmatrix} 156+156 & -22\ell+22\ell \\ -22\ell+22\ell & 4\ell^2+4\ell^2 \end{pmatrix}$$

$$= \frac{\rho A \ell}{105}\begin{pmatrix} 78 & 0 \\ 0 & 2\ell^2 \end{pmatrix}$$

$$K = \frac{EI}{\ell^3}\begin{pmatrix} 12+12 & -6\ell+6\ell \\ -6\ell+6\ell & 4\ell^2+4\ell^2 \end{pmatrix}$$

$$= \frac{8EI}{\ell^3}\begin{pmatrix} 3 & 0 \\ 0 & \ell^2 \end{pmatrix}$$

Thus:

$$\frac{\rho A \ell}{105}\begin{pmatrix} 78 & 0 \\ 0 & 2\ell^2 \end{pmatrix}\begin{pmatrix} \ddot{u}_3 \\ \ddot{u}_4 \end{pmatrix} + \frac{8EI}{\ell^3}\begin{pmatrix} 3 & 0 \\ 0 & \ell^2 \end{pmatrix}\begin{pmatrix} u_3 \\ u_4 \end{pmatrix} = P\sin\omega t \begin{pmatrix} N_3(a) \\ N_4(a) \end{pmatrix}$$

Solve by trial sol^{n}s $\begin{pmatrix} u_3(t) \\ u_4(t) \end{pmatrix} = \begin{pmatrix} \bar{u}_3 \\ \bar{u}_4 \end{pmatrix} \sin\omega t$.

Advection-reaction dispersion eq^{n}:

$$\underset{①}{\frac{\partial C}{\partial t}} = \underbrace{-v\frac{\partial C}{\partial x}}_{\substack{\text{advection} \\ ②}} + \underset{③}{D_L \frac{\partial^2 C}{\partial x^2}} + \underbrace{\rho C(1-C)}_{\text{reaction}}$$

① ④

$C = \sum N_i c_i$ (2 node element).

③ $$\int_0^L \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} \, dx \longrightarrow \frac{1}{L}\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

① $$\int_0^L N_i N_j \, dx \longrightarrow \frac{L}{6}\begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}$$

② $$\int_0^L N_i \frac{\partial N_j}{\partial x} \, dx \longrightarrow \frac{1}{2}\begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix}$$

④ (nonlinear) $$\int_0^L (N_1 c_1 + N_2 c_2)^2 N_i \, dx$$

$$\longrightarrow \int_0^L \begin{pmatrix} N_1^3 & 2N_1^2 N_2 & N_1 N_2^2 \\ N_1^2 N_2 & 2N_1 N_2^2 & N_2^3 \end{pmatrix} dx \begin{pmatrix} c_1^2 \\ c_1 c_2 \\ c_2^2 \end{pmatrix}$$

$$= L \begin{pmatrix} \frac{1}{4} & \frac{1}{6} & \frac{1}{12} \\ \frac{1}{12} & \frac{1}{6} & \frac{1}{4} \end{pmatrix} \begin{pmatrix} c_1^2 \\ c_1 c_2 \\ c_2^2 \end{pmatrix}$$

$\Rightarrow$ after multiplying by M^{-1}:

$$\begin{pmatrix} \dot{c}_1 \\ \dot{c}_2 \end{pmatrix} = \left[-\frac{v}{L}\begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix} - \frac{2D_L}{L^2}\begin{pmatrix} 3 & -3 \\ -3 & 3 \end{pmatrix} + \rho \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \right] \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

$$- \frac{\rho}{6}\begin{pmatrix} 5c_1^2 + c_1 c_2 - c_2^2 \\ -c_1^2 + c_1 c_2 + 5c_2^2 \end{pmatrix} - \frac{2}{L}\begin{pmatrix} -2J(0) - J(L) \\ J(0) + 2J(L) \end{pmatrix}$$

constitutive relⁿ

where $J = -D_L \dfrac{\partial c}{\partial x}$ (Fick's law)

a 2D nonlinear ODE.

END OF PART I

PART II

INTRODUCTION

An ODE of order n is

$$F\left(t, y, \frac{dy}{dt}, \dots, \frac{d^n y}{dt^n}\right) = 0$$

Resolved,

$$\frac{d^n y}{dx^n} = f\left(t, y, \frac{dy}{dt}, \dots, \frac{d^{n-1}y}{dt^{n-1}}\right)$$

For unique sol^n, we require n initial conditions

$$y(t_0) = a_0 \qquad \frac{dy}{dt}(t_0) = a_1 \qquad \dots \qquad \frac{d^{n-1}y}{dt^{n-1}}(t_0) = a_{n-1}$$

or $\begin{cases} M \text{ conditions at } t = t_0 \\ n-M \text{ conditions at } t = T \end{cases}$, need sol^n for $t_0 \le t \le T$

These are initial-value problems
and boundary-value problems.

Common trick: Write n^{th} order ODE as a system of n 1^{st} order ODEs.

$$\text{If } \frac{d^n y}{dt^n} = f\left(t, y, \underset{u_1}{\frac{dy}{dt}}, \dots, \underset{u_{n-1}}{\frac{d^{n-1}y}{dt^{n-1}}}\right)$$

$$u_2 = \frac{du_1}{dt} \quad \text{etc}$$

to get $\left\{ \frac{du_{n-1}}{dt} = f(t, y, u_1, u_2, \dots, u_{n-1}) \right.$

$$\left\{\begin{aligned} \frac{du_{n-2}}{dt} &= u_{n-1} \\ \frac{du_{n-3}}{dt} &= u_{n-2} \\ &\vdots \\ \frac{du_1}{dt} &= u_2 \\ \frac{dy}{dt} &= u_1 \end{aligned}\right.$$

n eq^{n}s

<u>Plan</u>: to study 1st-order ODE

$$\frac{dy}{dt} = f(t, y)$$

and if we need to solve a system, generalise:

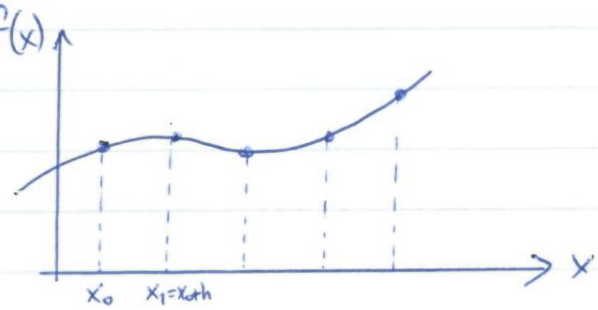
$$\frac{d\underline{y}}{dt} = \underline{f}(t, \underline{y}) .$$

APPROXIMATION OF A FIRST DERIVATIVE

We are given $f(x)$.

$f(x_n) = f(x_0 + nh)$

Approximate $f'(x)$ using $f_n = f(x_0 + nh)$



h - step size
constant $\Rightarrow$ grid is uniform.

Find, for example, $f'(x_0)$.

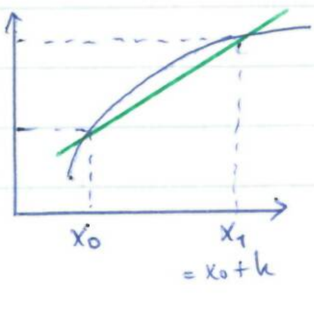
Use Taylor series:

$$\begin{aligned} f(x_1) &= f(x_0 + h) \\ &= f(x_0) + hf'(x_0) + O(h^2) \end{aligned}$$

Assume small h, ignore h^2 and get:

$$f_1 = f_0 + hf'(x_0)$$

$$\Rightarrow f'(x_0) = \frac{f_1 - f_0}{h}$$



FORWARD APPROXIMATION
($\because$ using point ahead of it)

Approximation error, due to series truncation:

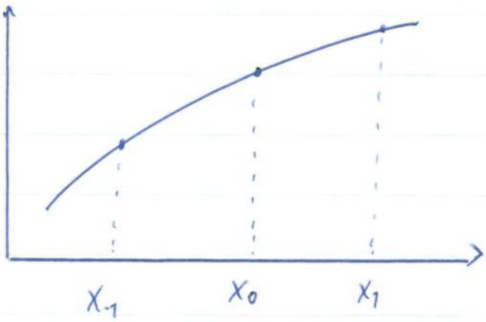
from the exact formula,

$$f_1 = f_0 + hf'(x_0) + O(h^2)$$

$$f'(x_0) = \frac{f_1 - f_0}{h} + O(h)$$

$\Rightarrow$ error is of order $O(h)$ $\Rightarrow$ "1st order approximation"

<u>Example</u>: Find a first-order <u>backward</u> approximation to $f'(x_0)$.



using $f_0 = f(x_0)$
$f_{-1} = f(x_{-1})$

answer: $f'(x_0) = \dfrac{f_0 - f_{-1}}{h}$

<u>Exercise</u>: Find a 2nd-order accurate forward approximation to $f'(x_0)$.

Use Taylor twice.

$f(x_0) = f_0$

$$f_1 = f(x_1) = f(x_0+h) = f_0 + hf'(x_0) + \frac{h^2}{2}f''(x_0) + O(h^3) \quad \cdots(1)$$

$$f_2 = f(x_2) = f(x_0+2h)$$
$$= f_0 + 2hf'(x_0) + \frac{(2h)^2}{2}f''(x_0) + O(h^3) \quad \cdots(2)$$

Eliminate $f''(x_0)$ between (1) and (2).
Find $f'(x_0)$

Instead of solving, guess: [solving is easy, but for sake of argument ...]

$$f'(x_0) = af_0 + bf_1 + cf_2$$

The Taylor expansion of a quadratic/linear eq^n is equivalent to our f^n.

?

Take $f = 1$. $\quad 0 = a+b+c$

Take $f = x$ $\quad 1 = 0 + bh + c2h$

Take $f = x^2$ $\quad 0 = 0 + bh^2 + c(2h)^2$

Solve to find a, b, c

$a = -\frac{2}{h} + \frac{1}{2h} \qquad b = \frac{2}{h} \qquad c = -\frac{1}{2h}$

$= -\frac{3}{2h}$

and $$f'(x_0) = \frac{-3f_0 + 4f_1 - f_2}{2h}$$

a 2nd-order accurate forward approximation for $f'(x_0)$

<u>Exercise</u>: A second-order <u>central</u> approximation for $f'(x_0)$ using x_{-1}, x_0, x_1.

second central [1st order]

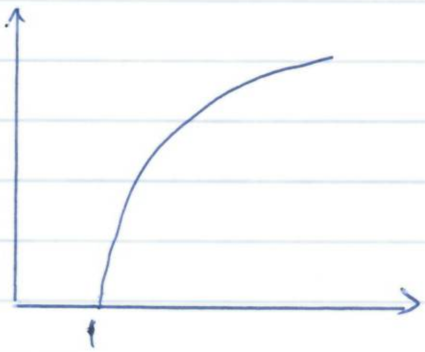
$$f'(x_0) = \frac{f_1 - f_{-1}}{2h} + O(h^2)$$

Find a backward 2nd-order accurate approximation for $f'(x_0)$ using f_{-2}, f_{-1}, f_0.

$$\frac{f_{-2} - 4f_{-1} + 3f_0}{2h}$$

<u>Exercise</u>: $f(x) = \sqrt{x-1}, \quad x \geq 1$.

What can you say about finite difference approximations at $x=1$?



Try $f'(x_0) = \dfrac{f_1 - f_0}{h}$. (use fwd ∵ at 1 there is nothing to the left).

"What's a nice word for screwed?"

$$f'(1) = \frac{\sqrt{h} - 0}{h} = \frac{1}{\sqrt{h}}$$

hmm... doesn't seem right.

$x_1 = x_0 + h$
$x_1 = 0 + h$

Also, in error estimate,

$$f'(x_0) = \frac{f_1 - f_0}{h} + O(f''(x_0))$$

If f'' is large numerically, the error is large.

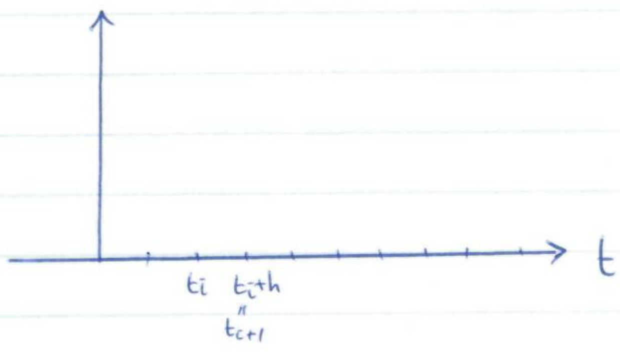
Error for Euler method in ODEs

Solving $\frac{dy}{dt} = f(y,t)$

First-order forward-finite

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

$i = 0, 1, \ldots$
h = step size in time



From Taylor for exact solⁿ,

$$y(t_{i+1} = t_i + h) = y(t_i) + h \left.\frac{dy}{dt}\right|_{t_i} + \frac{h^2}{2} \left.\frac{d^2y}{dt^2}\right|_{t=\xi}$$

where $t_i \leq t = \xi \leq t_{i+1}$

Recall $y_{i+1} = y_i + hf(y_i, t_i)$

Denote $e_i = y(t_i) - y_i$.

Then $e_{i+1} = e_i + h\left[\left.\frac{dy}{dt}\right|_{t_i} - f(y_i, t_i)\right] + \frac{h^2}{2}\left.\frac{d^2y}{dt^2}\right|_{t=\xi}$

But $\frac{dy}{dt} = f(y,t)$

$\left.\frac{dy}{dt}\right|_{t_i} = f\big(y(t_i), t_i\big)$

Then $e_{i+1} = e_i + h\big[\underbrace{f(y(t_i), t_i)}_{\text{exact}} - \underbrace{f(y_i, t_i)}_{\text{finite-diff approx.}}\big] + \frac{h^2}{2}\left.\frac{d^2y}{dt^2}\right|_{t=\xi}$

$t_i \leq \xi_i \leq t_{i+1}$

Assume $\left|\frac{d^2y(\xi_i)}{dt^2}\right| \leq M \quad \forall \xi_i$

and Lipshitz for f

$$|f(y_1,t) - f(y_2,t)| \leq L|y_1 - y_2| \qquad \forall t$$

Worst case scenario if $|e_{i+1}| = |e_i| + hL|e_i| + \frac{1}{2}h^2M$,

ie $|e_{i+1}| = (1 + hL)|e_i| + \frac{1}{2}h^2M$

Difference equation of the form

"$x_{n+1} = ax_n + b$"

Try $x_n = A = \text{const}$: $\quad A = aA + b \Rightarrow A = \frac{b}{1-a}$

particular sol^{n}

Solve homogeneous eq^{n}

$$x_{n+1} = a x_n$$

Try sol^{n} as $x_n = \beta^n$ for some β.

Find $\beta = a$.

(Compare with $y' = ay$: $y = e^{\lambda x}$ get $\lambda = a$.)

Conclude that general sol^{n}

$$x_n = Ca^n + \frac{b}{1-a}$$

hence

$$|e_i| = C(1+hL)^i - \frac{h^2M}{2hL}$$

Is this always positive? ($\because$ it had better be (modulus!))

If $i=0$, $\quad |e_0| = C - \frac{hM}{2L} \Rightarrow C = |e_0| + \frac{hM}{2L}$

$\Rightarrow$

$$|e_i| = \left(|e_0| + \frac{hM}{2L}\right)(1+hL)^i - \frac{hM}{2L}$$

and this is > 0. Or is it?

Use $(1+\alpha)^x \le e^{\alpha x}$

$$|e_i| = \left(|e_0| + \frac{hM}{2L}\right)e^{ihL} - \frac{hM}{2L}$$

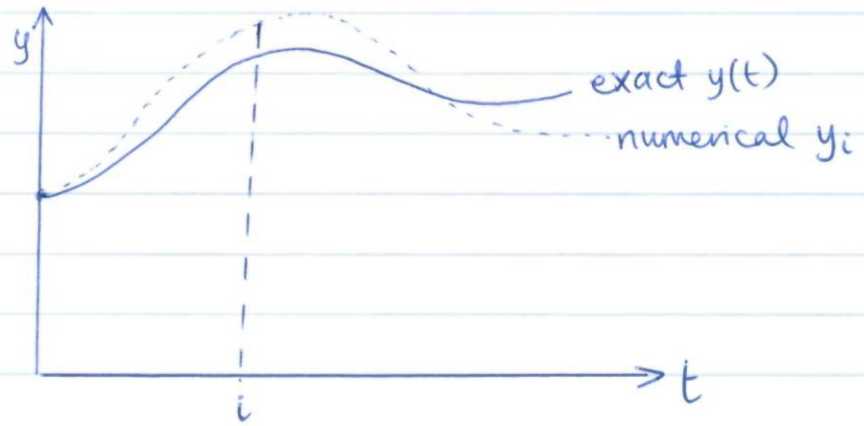
this could be huge. Be careful...

$$= |e_0| e^{ihL} + \frac{hM}{2L}\left(e^{ihL} - 1\right)$$

Let $|e_0| = 0$.

Then $|e_i| \leq \frac{hM}{2L}\left(e^{ihL} - 1\right)$,

Conclusion:



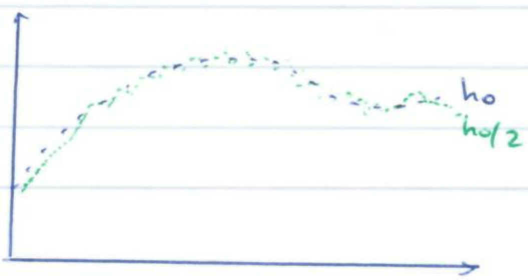
Local truncation error $= O(h^2)$

Global truncation error $= O(h)$

In practice, y(t) exact is unknown.
How do you find the global truncation error?

Take, e.g., step size $\Delta t = h_0$

Repeat for $\Delta t = \frac{h_0}{2}$

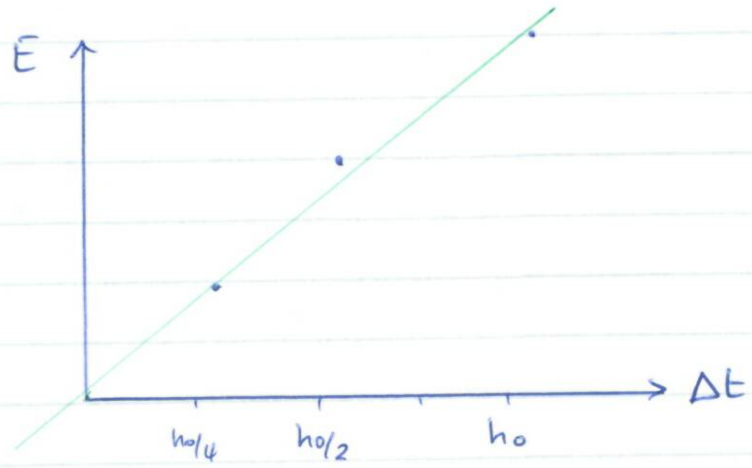


Take $E(\Delta t) = \max_i \left| y|_{h_0} - y|_{h_0/2} \right|$

And repeat for $\Delta t = \frac{h_0}{4}$.

Take $E(\Delta t) = \max_i \left| y|_{h_0/2} - y|_{h_0/4} \right|$

and plot



this line should go through the origin.

Taylor methods

Solving $\frac{dy}{dt} = f(y,t)$

write $y(t_i + \Delta t) = y(t_i) + \Delta t\, y'(t_i) + \frac{(\Delta t)^2}{2} y''(t_i)$

Can take 2, 3 or 4 terms.

but need $y'(t_i) = f(y_i, t_i)$

$$y''(t_i) = \frac{d}{dt} f(y,t)\Big|_{t=t_i} = \frac{\partial f}{\partial t}\Big|_{t=t_i} + \frac{\partial f}{\partial y}\frac{dy}{dt}\Big|_{t=t_i}$$

$$= \left.\frac{\partial f}{\partial t}\right|_{t_i} + f \left.\frac{\partial f}{\partial y}\right|_{\substack{t=t_i \\ y=t_i}}$$

need derivatives numerically
very time-consuming
and inefficient.

Runge-Kutta methods

Starting from Taylor expansion,

$$y(t_i+h) = y(t_i) + h\left.\frac{dy}{dt}\right|_{t_i} + \frac{h^2}{2}\frac{d^2y}{dt^2} + O(h^3)$$

for equation $\dfrac{dy}{dt} = f(y,t)$

then
$$\frac{d^2y}{dt^2} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial y}\frac{dy}{dt}$$
$$= \frac{\partial f}{\partial t} + f\frac{\partial f}{\partial y}$$

Approximation to order $O(h^3)$

$$y_{i+1} = y_i + hf(y_i, t_i) + \frac{h^2}{2}\left.\frac{\partial f}{\partial t}\right|_{\substack{t_i \\ y_i}} + \frac{h^2}{2} f \left.\frac{\partial f}{\partial y}\right|_{\substack{t_i \\ y_i}} \quad \text{- - - -} (*)$$

Idea: $y_{i+1} = y_i + h\left[\omega_1 f(y_i, t_i) + \omega_2 f(y_i + Y, t_i + T)\right]$

Assume Y, T are small and expand:

$$y_{i+1} = y_i + h\Big[\omega_1 f(y_i, t_i) + \omega_2 f(y_i, t_i)$$
$$+ \omega_2 T \frac{\partial f}{\partial t}(y_i, t_i) + \omega_2 Y \frac{\partial f}{\partial y}(y_i, t_i)\Big] + hO(T^2, Y^2, YT)$$

$$\dots (**)$$

Want $(*) = (**)$ to $O(h^3)$.

Take $T = \lambda h$ $\qquad\qquad \lambda = \text{const}.$
$Y = \alpha h f(y_i, t_i)$

Then $(*) = (**)$

$$\text{if} \begin{cases} h(\omega_1 + \omega_2) = h \\ h\omega_2 \lambda h = \frac{h^2}{2} \\ h\omega_2 \alpha h f = \frac{h^2}{2} f \end{cases}$$

Error in $(**) = h \cdot O(h^2, h^2, h^2) = h \cdot O(h^2) =$ Error in $(*)$.

Need to solve: $\begin{cases} \omega_1 + \omega_2 = 1 \\ \omega_2 \lambda = \frac{1}{2} \\ \omega_2 \alpha = \frac{1}{2} \end{cases}$

$\Rightarrow \alpha = \lambda$, and can take, e.g. ω_1 as a free parameter

$\Rightarrow \omega_2 = 1 - \omega_1$

$\Rightarrow \lambda = \alpha = \dfrac{1}{2(1-\omega_1)}$. Infinitely many sol^{n}s.

and so infinitely many RK algorithms.

__Modified Euler__: $\begin{cases} w_1 = 0 \\ w_2 = 1 \\ \alpha = \lambda = \frac{1}{2} \end{cases}$

$$y_{i+1} = y_i + hf\left(y_i + \frac{h}{2}f_i,\ t_i + \frac{h}{2}\right)$$

__Heun's__: $\begin{cases} w_1 = w_2 = \frac{1}{2} \\ \lambda = \alpha = 1 \end{cases}$

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

For Runge-Kutta 2nd order ("RK2")

$$\text{Local truncation error} = O(h^3)$$
$$\text{global truncation error} = O(h^2)$$

__Question__: Using Euler with $0 \le t \le 1$, $\Delta t = 10^{-4}$

and RK2 with $\Delta t = 10^{-2}$;

we get same accuracy. Which is better?

Euler: $y_{i+1} = y_i + \underbrace{hf(y_i, t_i)}_{\text{one main calculation}}$

$$\#\text{computations} = \frac{1}{\Delta t} = 10^4$$

RK2: $y_{i+1} = y_i + \frac{h}{2}\Big[\underbrace{f(y_i, t_i) + f(y_i + hf_i, t_i + h)}_{\text{2 main calculations}}\Big]$

$$\# \text{computations} = \frac{2}{\Delta t} = 2 \cdot 10^2$$

much more efficient

Difference eq^n

The difference eq^n of order N is

$$a_N y_{N+k} + a_{N-1} y_{(N-1)+k} + a_{N-2} y_{(N-2)+k} + \dots + a_0 y_k = b_k$$

ie a linear combination of $\{y_k\}$

e.g. $y_{n+1} - y_n = 1 \qquad (n = 0, 1, \dots)$

e.g. A difference eq^n with nonconstant coefficients

$$y_{n+1} - (n+1) y_n = 0$$

$$y_{n+1} = y_0 (n+1)!$$

The general sol^n is: the particular solution
+ general sol^n of homogeneous eq^n

If we have constant coefficients a_n then we have a general homogeneous sol^n, found by trial,

$$y_n = \beta^n$$

with β to be found. We expect N values of β.

If the β's merge, try

$$y_n = n\beta^n$$

(an^2+bn) for triple root.

Example $\quad 2y_{n+1} - 3y_n + y_{n-1} = 1$

Solve with $y_0 = 5$, $\quad y_1 = 3$.

1. Particular sol^n: $\quad$ • $y_n = C$ is not working

• $y_n = nC$:

$$2(n+1)C - 3nC + (n-1)C = 1$$
$$2C - C = 1$$
$$\Rightarrow C = 1$$

$\Rightarrow \quad y_n = n$ is a sol^n.

2. Solve $\quad 2y_{n+1} - 3y_n + y_{n-1} = 0$

Try $y_n = \beta^n$:

$$2\beta^{n+1} - 3\beta^n + \beta^{n-1} = 0$$

$$\beta^{n-1}(2\beta^2 - 3\beta + 1) = 0$$

$\beta = 0 \qquad \beta = \dfrac{3 \pm \sqrt{9-8}}{4} = 1, \tfrac{1}{2}$

$\Rightarrow \quad y_n = 1^n = 1$
or $\frac{1}{2}^n$

$\Rightarrow$ General sol^n is

$$y_n = n + C_1 \cdot 1 + C_2 \left(\tfrac{1}{2}\right)^n \qquad \forall C_1, C_2$$

Then use initial conditions $y_0 = 5$
$y_1 = 3$

$$\Rightarrow \left.\begin{aligned} 5 &= C_1 + C_2 \\ 3 &= 1 + C_1 + C_2 \tfrac{1}{2} \end{aligned}\right\} \text{ and can solve.}$$

&

$C_1 = -1$ (thanks Gin)
$C_2 = 6$

$$y_n = n - 1 + \frac{6}{2^n}$$

Note: Suppose the eq^n is homogeneous, so
$y_n = C\beta^n$
is one of the sol^{n}s and
$y_n \equiv 0$
is also a solution.

Question: Is the sol^n $y_n \equiv 0$ stable to small changes in initial conditions?

e.g. $y_{n+1} - 2y_n = 0$

If $y_0 = 0$ then $y_n \equiv 0$.

General sol^n: $y_n = \beta^n$

$$\beta^{n+1} - 2\beta^n = 0 \quad \Rightarrow \quad \beta = 2$$

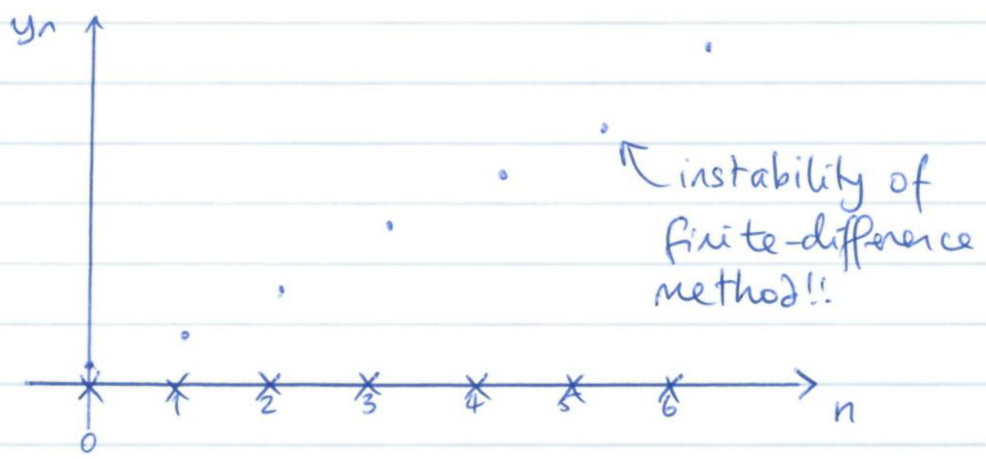
$$\Downarrow$$

$$y_n = 2^n$$

$$\Downarrow$$

general sol^n is $y_n = y_0 \beta^n$

If $y_0 \neq 0$ then $y_n \to \infty$ as $n \to \infty$



Conclusion: if at least one β has property $|\beta| > 1$ then we have instability.

- Instability can be monotone if the offensive β is real and positive
- it can be oscillatory on the step if β is real and negative

  e.g. $\beta = -2$, $y_n = y_0(-2)^n$
- it can be oscillatory if β is complex

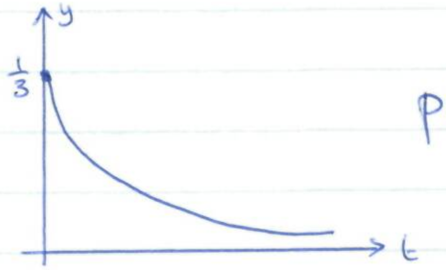
e.g $\beta_1 = 2+i, \quad \beta_2 = 2-i$

$y_n = c_1(2+i)^n + c_2(2-i)^n$

oscillates like $\cos n$, $\sin n$.

<u>Example</u> Solve $\begin{cases} y' = -30y \\ 0 \le t \le 1.5 \\ y(0) = \frac{1}{3} \end{cases}$

Exactly: $y = \frac{1}{3}e^{-30t}$



where are horses

pretty darn stable

Finite-diff. Euler $\dfrac{y_{n+1} - y_n}{h} = -30y_n$

$\begin{cases} y_{n+1} = y_n - 30hy_n \\ y_0 = \frac{1}{3} \end{cases}$

Results: exact $y(1.5) = 9.54 \times 10^{-21}$

Euler, $h=0.1$: $y(1.5) = -1.1 \times 10^4$

RK4, $h=0.1$: $y(1.5) = 3.96 \times 10^1$

nowhere near!!

Why is it so far out?

For Euler, $y_{n+1} = (1-30h)\,y_n$, $\quad \beta = 1-30h$.

Instability if $|\beta| > 1$

$$\downarrow$$

$$|1-30h| > 1$$

$$\downarrow$$

$$1-30h > 1 \quad \text{or} \quad 1-30h < -1$$

$$\Rightarrow \quad h < 0 \quad \text{or} \quad h > \tfrac{1}{15}$$

$\Rightarrow$ Range of stability is $0 < h < \frac{1}{15}$.

<u>Example</u> $\quad y' = \lambda y. \qquad \lambda \in \mathbb{C}$

What is stability of Euler:

$$y_{n+1} = y_n + \lambda h y_n \quad ?$$

$$= (1+\lambda h) y_n$$

$$\Rightarrow \beta = 1 + \lambda h$$

$|\beta| < 1$ gives stable, i.e. $-1 < |1+\lambda h| < 1$

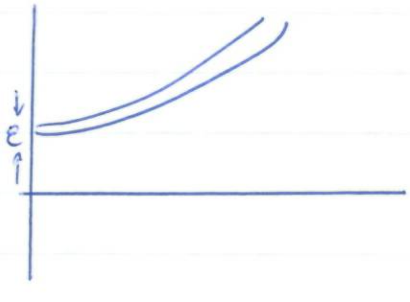
L

<u>General idea for stability calculations</u>

Unstable ODE

e.g. $\dfrac{dy}{dt} = y \quad \Rightarrow \quad y = y_0 e^t$

ε-difference in initial conditions



Difference between solutions at time t is εe^t.

In this problem, solution is stable if t is bounded.
It can be said to be unstable if $t \to \infty$.

Let $y_{n+1} = T(y_n, h)$.

Introduce perturbation

$$y_{n+1} + \varepsilon_{n+1} = T(y_n + \varepsilon_n, h)$$

Assume small ε_n $\forall n$

$$y_{n+1} + \varepsilon_{n+1} = T(y_n, h) + \varepsilon_n \frac{\partial T}{\partial y_n} + O(\varepsilon_n^2)$$

$$\Rightarrow \qquad \varepsilon_{n+1} = \varepsilon_n \frac{\partial T}{\partial y_n}(y_n, h)$$

Pretend $\dfrac{\partial T}{\partial y_n} = \text{const} = g$.

$$\Rightarrow \qquad \varepsilon_{n+1} = g\varepsilon_n$$
$$\Rightarrow \qquad \varepsilon_n = g^n \varepsilon_0$$

INSTABLE
- $\to |g| > 1$ $\quad \varepsilon_n \to \infty$ as $n \to \infty$
- $\to |g| = 1$ $\quad$ in between
- $\to |g| < 1$ $\quad \varepsilon_n \to 0$ as $n \to \infty$

STABLE

<u>Exercise</u> RK2:

$$\begin{cases} y_{n+\frac{1}{2}} = y_n + \frac{h}{2} f(y_n, t_n) \\ y_{n+1} = y_n + h f(y_{n+\frac{1}{2}}, t_{n+\frac{1}{2}}) \end{cases}$$

Need $g = \dfrac{\partial y_{n+1}}{\partial y_n}$ assuming $y_{n+1} = T(y_n, t_n)$
$= y_{n+1}(y_n, t_n)$.

We have $$\frac{\partial y_{n+1}}{\partial y_n} = 1 + h \frac{\partial f}{\partial y_{n+\frac{1}{2}}} \frac{\partial y_{n+\frac{1}{2}}}{\partial y_n} \qquad (1)$$

and $$\frac{\partial y_{n+\frac{1}{2}}}{\partial y_n} = 1 + \frac{h}{2} \frac{\partial f}{\partial y_n} \qquad (2)$$

Want to plug (2) into (1).
We have $\dfrac{\partial f}{\partial y_{n+\frac{1}{2}}}$ and $\dfrac{\partial f}{\partial y_n}$ ← fine

trouble.

approximate this by $\dfrac{\partial f}{\partial y_n}$.

$$(1) \Rightarrow \frac{\partial y_{n+1}}{\partial y_n} = 1 + \underbrace{h \frac{\partial f}{\partial y_n}}_{\Delta} \left(1 + \frac{h}{2} \frac{\partial f}{\partial y_n}\right).$$

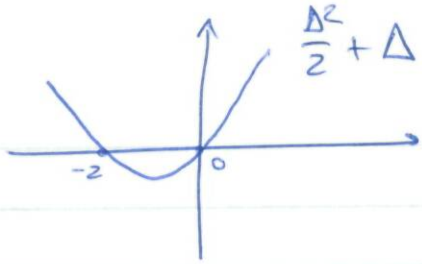
For stability, $\left|\dfrac{\partial y_{n+1}}{\partial y_n}\right| < 1$

ie $$\left| 1 + \Delta\left(1 + \frac{\Delta}{2}\right) \right| < 1$$

ie $$\Big| \underbrace{\frac{\Delta^2}{2} + \Delta + 1}_{\text{this quadratic is always} > 0} \Big| < 1$$

ie $$\frac{\Delta^2}{2} + \Delta + 1 < 1$$

ie $$\frac{\Delta^2}{2} + \Delta < 0$$



ie $\quad -2 < \Delta < 0$

ie $\quad -2 < h \dfrac{\partial f}{\partial y_n} < 0.$

+ve

For stability, need $\dfrac{\partial f}{\partial y_n} < 0$

e.g. $\dfrac{dy}{dt} = y \quad \Rightarrow \quad f = y.$

$$-2 < h\frac{\partial f}{\partial y} \quad \text{with} \quad \frac{\partial f}{\partial y} = 0$$

means $\quad h < \dfrac{2}{\left|\frac{\partial f}{\partial y}\right|}.$

For stability, $\quad g = 1 + h\dfrac{\partial f}{\partial y_n}$

must satisfy $|g| < 1$

true for complex f as well.

Write $\quad |g|^2 < 1$

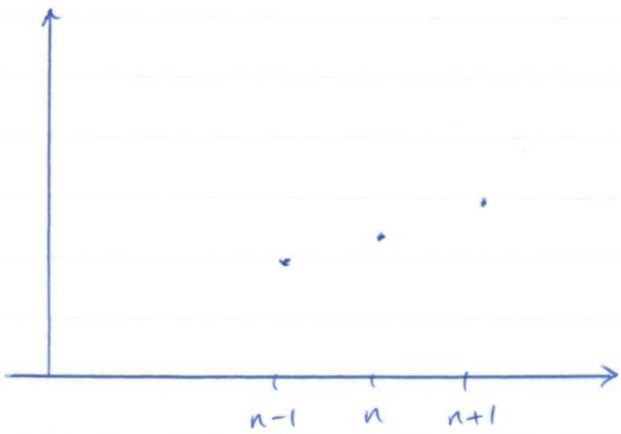
$\qquad\qquad gg^* < 1$

$$\left(1 + h\frac{\partial f}{\partial y_n}\right)\left(1 + h\left(\frac{\partial f}{\partial y_n}\right)^*\right) < 1.$$

Leap-frog method

Solve $\quad \dfrac{dy}{dt} = f(y,t)$

by $\quad y_{n+1} = y_{n-1} + 2hf(y_n, t_n)$



local truncation error $= O(h^{\overset{2}{\cancel{3}}})$ $\quad$ higher order than Euler.

Problem: it is hard to start from initial condition at $t=0$.

Exercise: Suggest starting procedure for leap-frog method if

$$\frac{dy}{dt} = y^2 \qquad y(1)=1 \qquad t \geq 1$$

So $\quad y_{n+1} = y_{n-1} + 2hy_n^2$
$\quad t_n = 1 + nh$
and $y_0 = 1$.

How do we get y_1?

Options: (1) Take $y_1 = y_0$. — NOT ORDER h^3

(2) At $t=0$, write alternative approximation to ODE using forward derivative of $O(h^2)$.

$$\frac{dy}{dt} = \frac{-3y_0 + 4y_1 - y_2}{2h}$$

$$\Rightarrow \frac{dy}{dt} = y^2$$

$$\Rightarrow \frac{-3y_0 + 4y_1 - y_2}{2h} = y_0^2 \longrightarrow \text{find } y_1$$

and then use leap-frog $y_2 = y_0 + 2hy_1^2$ to find $y_2, y_3, \ldots$

<u>Exercise</u>: Stability of leap frog.

$$y_{n+1} = y_{n-1} + 2hf(y_n, t_n)$$

Compute $\frac{\partial y_{n+1}}{\partial y_n}$

$$\frac{\partial y_{n+1}}{\partial y_n} = \frac{\partial y_{n-1}}{\partial y_n} + 2h\frac{\partial f}{\partial y_n}$$

Assume $\frac{\partial y_{n+1}}{\partial y_n} \approx \frac{\partial y_n}{\partial y_{n-1}} = g$ const.

$$\Rightarrow g = \frac{1}{g} + 2h\frac{\partial f}{\partial y_n}$$

$$\Rightarrow g^2 = 1 + 2\Delta g$$

$$\Rightarrow g^2 - 2\Delta g - 1 = 0.$$

$$\Rightarrow \quad g = \frac{2\Delta \pm \sqrt{4\Delta^2 + 4}}{2}$$

$$= \Delta \pm \sqrt{\Delta^2 + 1}.$$

Recall $y_{n+1} = y_{n-1} + 2hf(y_n, t_n)$

2nd-order difference eq^n
$\Rightarrow$ 2 linearly indpt sol^{n}s

For stability, $|g| < 1$

$$\left| \Delta \pm \sqrt{\Delta^2 + 1} \right| < 1.$$

If Δ is real then the method is always unstable

Explicit scheme: $y_{n+1} = T(\underbrace{y_n, y_{n-1}, \ldots, t_n}_{\text{from previous steps}})$

Unconditional stability = stable $\forall h$

Explicit methods tend to be not unconditionally stable.

Implicit methods

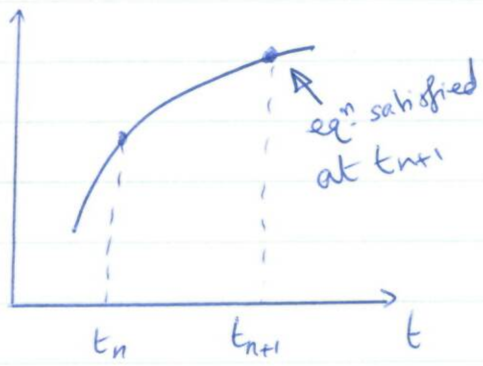
Ex: Euler

given $\frac{dy}{dt} = f(y,t)$, $\quad y(0) = y_0$

write a backward derivative at $t = t_n$

$$\frac{y_{n+1} - y_n}{h} = f(y_{n+1}, t_{n+1})$$

$$\Rightarrow \quad y_{n+1} = y_n + hf(y_{n+1}, t_{n+1})$$



Stability | want $\frac{\partial y_{n+1}}{\partial y_n}$.

$$\frac{\partial y_{n+1}}{\partial y_n} = 1 + h \frac{\partial f}{\partial y_{n+1}} \frac{\partial y_{n+1}}{\partial y_n}$$

Denote $h \frac{\partial f}{\partial y_{n+1}} = \Delta$, $\quad g = \frac{\partial y_{n+1}}{\partial y_n}$

$$\Rightarrow \quad g = 1 + \Delta g$$

$$\Downarrow$$

$$g(1-\Delta) = 1 \quad \Rightarrow \quad g = \frac{1}{1-\Delta}$$

For stability, want $|g| < 1$, ie

$$-1 < \frac{1}{1-\Delta} < 1$$

ie "$\underbrace{-1 > \underbrace{1-\Delta > 1}_{\Delta<0}}_{\Delta>2}$"

what does this mean?!

$\boxed{\text{-}}$ If $\frac{\partial f}{\partial y} < 0$ then stable $\forall h$ (unconditional stability)

e.g. $\frac{dy}{dt} = \lambda y$

Backward Euler $y_{n+1} - y_n = h\lambda y_{n+1}$

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

If $\lambda<0$, $y_{n+1} = \frac{y_n}{1+|\lambda|h}$

Question: Backward Euler is also stable if $h\frac{\partial f}{\partial y} > 2$

e.g. $\frac{dy}{dt} = \lambda y$, $\lambda > 0$

So if $\lambda = 100$, $h = 0.3$, $h\frac{\partial f}{\partial y} = \lambda h = 3$

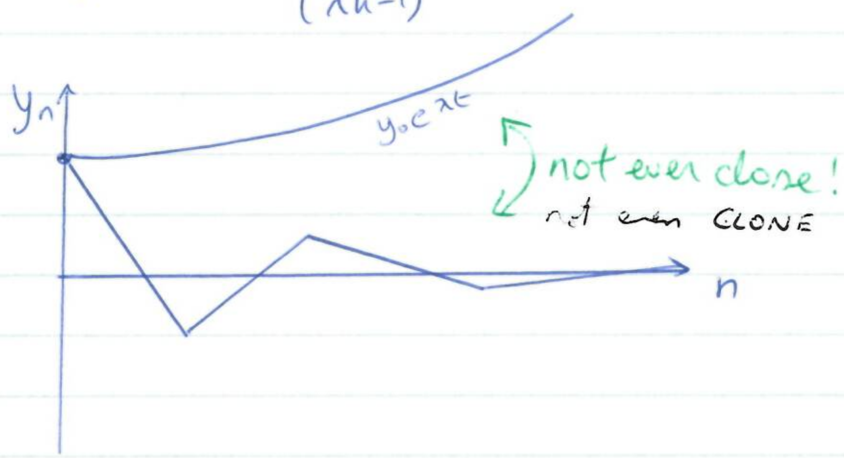
[stable]

Exact: $y = y_0 e^{\lambda t}$

Backward Euler: $y_{n+1} = \frac{y_n}{1-\lambda h}$

$$y_n = y_0 \frac{1}{(1-\lambda h)^n}$$

$$y_n = y_0 \frac{(-1)^n}{(\lambda h - 1)^n}$$

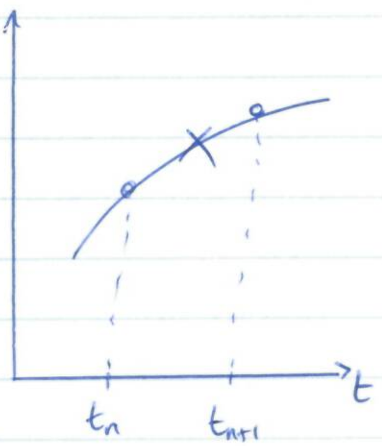


2 conditions needed
(1) approximately
(2) stability
$\Big\} \overset{\text{gives}}{\Longrightarrow}$ convergence ☺

Here the numerics are rubbish ∴ we haven't made the right approximation.

Crank-Nicholson approximation

(unconditionally stable ← CHECK!)



$$\frac{dy}{dt} = f(y,t)$$

Idea: using points (t_n, y_n), (t_{n+1}, y_{n+1})

Create a 2nd order accurate method.

Approximate $\dfrac{dy}{dt} = f(y,t)$

at mid-point $t = t_{n+\frac{1}{2}}$

central 2^{nd} order about $n+\frac{1}{2}$.

$$\left.\frac{dy}{dt}\right|_{\substack{t=t_{n+\frac{1}{2}} \\ =t_n+\frac{h}{2}}} = \frac{y_{n+1}-y_n}{h} + O(h^2)$$

To get $f(y_{n+\frac{1}{2}}, t_{n+\frac{1}{2}})$ given $f(y_n,t_n)$ and $f(y_{n+1}, t_{n+1})$, do Taylor.

$$f_n = f(y_n, t_n)$$
$$f_{n+1} = f(y_{n+1}, t_{n+1})$$

Method: Taylor for f_n, expanding about $t_{n+\frac{1}{2}}$, then Taylor for f_{n+1}, expanding about $t_{n+\frac{1}{2}}$ then eliminate unknown constants.

$$f_n = f_{(n+\frac{1}{2})-\frac{1}{2}} = f(t_{n+\frac{1}{2}} - \tfrac{h}{2})$$
$$= f_{n+\frac{1}{2}} - \frac{h}{2} f'_{n+\frac{1}{2}} + \left(\frac{h}{2}\right)^2 \frac{1}{2} f''_{n+\frac{1}{2}} + \cdots$$

and $f_{n+1} = f(t_{n+\frac{1}{2}} + \frac{h}{2})$

$$= f_{n+\frac{1}{2}} + \frac{h}{2} f'_{n+\frac{1}{2}} + \left(\frac{h}{2}\right)^2 \frac{1}{2} f''_{n+\frac{1}{2}}$$

Add results: $f_n + f_{n+1} = 2f_{n+\frac{1}{2}} + \left(\frac{h}{2}\right)^2 f''_{n+\frac{1}{2}} + \cdots$

so $f_{n+\frac{1}{2}} = \frac{1}{2}(f_n + f_{n+1}) + O(h^2)$.

$\Rightarrow$

<u>Crank-Nicholson:</u>

*
$$\frac{y_{n+1} - y_n}{h} = \frac{1}{2}\left[f(y_n, t_n) + f(y_{n+1}, t_{n+1})\right] + O(h^2)$$

implicit, second-order accurate.

Systems of 1st-order ODEs

$$\frac{d\underline{y}}{dt} = \underline{F}(\underline{y}, t) \quad \text{where} \quad \begin{aligned} \frac{dy_1}{dt} &= F_1(y_1, \ldots, y_N, t) \\ \frac{dy_2}{dt} &= F_2(y_1, \ldots, y_N, t) \\ &\vdots \\ \frac{dy_N}{dt} &= F_N(y_1, \ldots, y_N, t) \end{aligned}$$

Forward-Euler: $\underline{y}_{k+1} = \underline{y}_k + h\underline{F}\big|_{t=t_k}$

Stability: $\underline{\underline{J}} = \left(\frac{\partial F_i}{\partial y_j}\right)_{i,j}$

Stable if $|\text{eigenvector}| < 1$

F) Hyperbolic PDEs

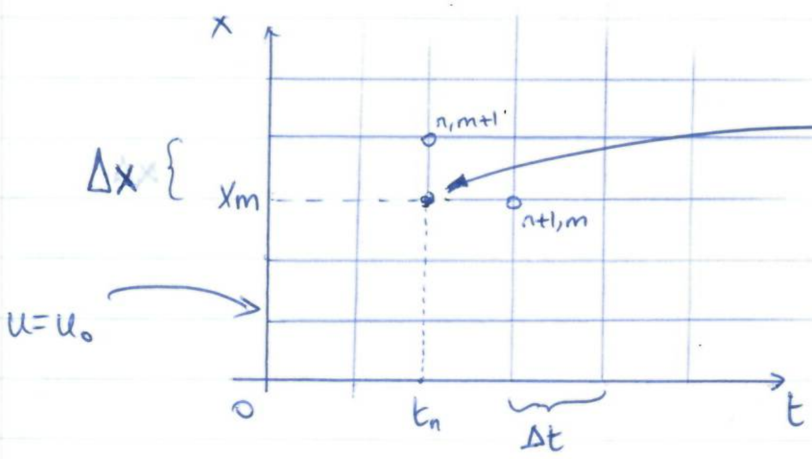
Exercise:
$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0$$

$$u = u(x,t). \qquad a, \text{ const} > 0$$

has sol$^{\underline{n}}$ $\quad u = u(x-at)$.

Give initial conditions $\quad u|_{t=0} = u_0(x)$.

Try a first-order in both t and x, and approximate with forward t, x-derivatives.



$$u_{nm} = u(t = \Delta t\cdot n,\ x = \Delta x \cdot m)$$

$$\frac{\partial u}{\partial t} = \frac{u_{n+1,m} - u_{n,m}}{\Delta t} \quad \leftarrow \text{first forward [1st der.]}$$

$$\frac{\partial u}{\partial x} = \frac{u_{n,m+1} - u_{n,m}}{\Delta x} \quad \leftarrow \text{first forward [1st der.]}$$

Sub into equation:

$$\frac{u_{n+1,m} - u_{n,m}}{\Delta t} + a\,\frac{u_{n,m+1} - u_{n,m}}{\Delta x} = 0$$

If there's anything I hate in life - it's derivatives and instability.

$$u_{n+1,m} = u_{n,m}\left(1 + a\frac{\Delta t}{\Delta x}\right) - a\frac{\Delta t}{\Delta x}\,u_{n,m+1}$$

Fourier?

For stability, write

$$u_{n,m} = \lambda^n e^{i\omega m} \qquad \omega, \lambda \text{ const}.$$

[Recall: In ODEs, $u_n \sim \beta^n$]

$$\lambda^{n+1} e^{i\omega m} = \lambda^n e^{i\omega m}\left(1 + a\frac{\Delta t}{\Delta x}\right) - a\frac{\Delta t}{\Delta x}\lambda^n e^{i\omega(m+1)}$$

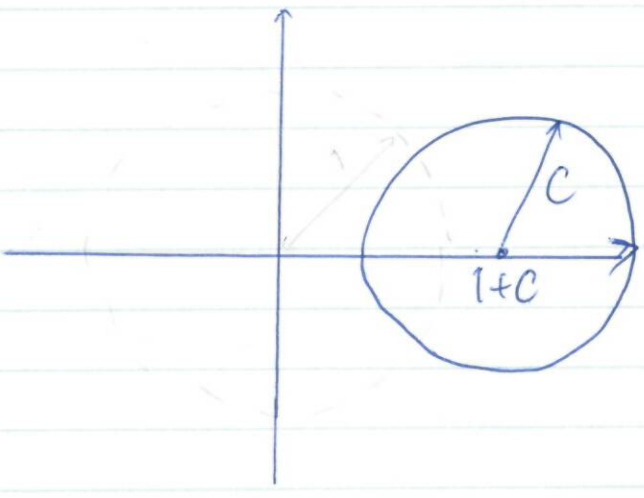
$$\Rightarrow \lambda = 1 + a\frac{\Delta t}{\Delta x} - a\frac{\Delta t}{\Delta x}e^{i\omega} \qquad = 1 + C - Ce^{i\omega}$$

So we have $\lambda = \lambda(\omega)$.

For stability, we need $|\lambda| < 1$.

Since $a > 0$, $(\Delta t, \Delta x) > 0$, $1 + a\frac{\Delta t}{\Delta x} > 1$

$\Rightarrow |\lambda| > 1$ for some ω. [e.g. pick $\omega = \pi$]

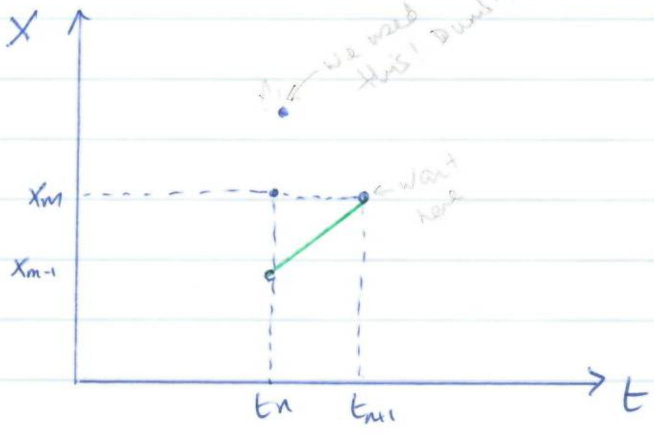


$$C = a\frac{\Delta t}{\Delta x}$$

$\Rightarrow$ method is unstable for any Δt, Δx !!

<u>Explanation</u>: $$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0 .$$

Exact sol^{n} : $u = u(x - at)$

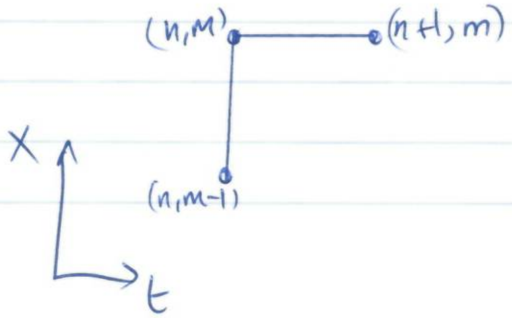


Suppose I know value at (n,m) but want to find it at $(n+1,m)$.

u is constant on the "characteristic" lines $x - at = \text{const}$

Exact sol^{n} tells us that the sol^{n} at $(m,n+1)$ is the same as $(m-1,n)$. So we made a big mistake: we used a forward x-derivative, ie we used $(n,m+1)$ which had nothing to do with it !!

Try backward x-derivative
& forward t-derivative.



equation $\quad \dfrac{\partial u}{\partial t} + a\dfrac{\partial u}{\partial x} = 0$

First forward [1st der.]

First backward [1st der.]

$$\frac{u_{n+1,m} - u_{n,m}}{\Delta t} + a\,\frac{u_{n,m} - u_{n,m-1}}{\Delta x} = 0$$

$$\Rightarrow\quad u_{n+1,m} = u_{n,m}\left(1 - \frac{a\Delta t}{\Delta x}\right) + a\frac{\Delta t}{\Delta x}\,u_{n,m-1}$$

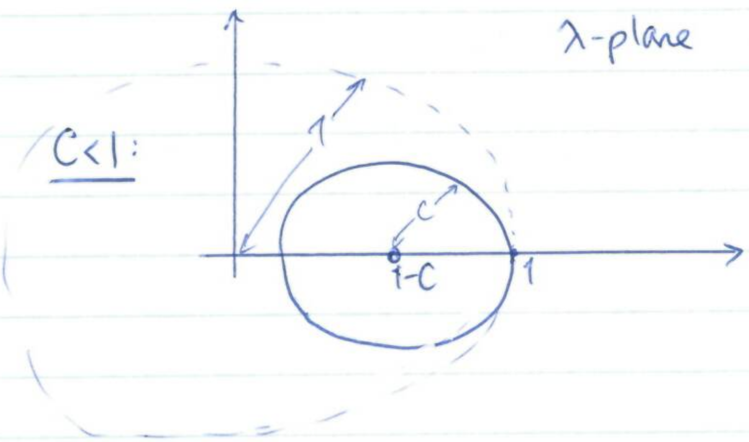
Let $C = \dfrac{a\Delta t}{\Delta x}$: $\longleftarrow$ COURANT N°

$$u_{n+1,m} = u_{n,m}(1-C) + C u_{n,m-1}.$$

<u>Stability</u>: $\quad u_{nm} = \lambda^n e^{i\omega m}$

$$\Rightarrow\quad \lambda = 1 - C + Ce^{-i\omega}$$

<u>C<1</u>:

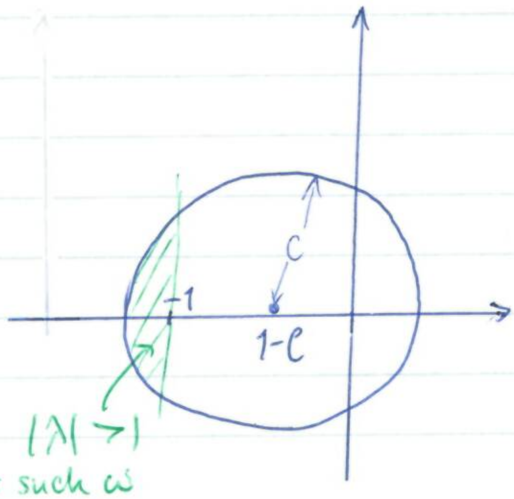


$|\lambda| < 1$ except $\omega = 0$

$\Downarrow$

stable if $C < 1$

<u>C>1</u>:



$|\lambda| > 1$
for such ω
<u>instability.</u>

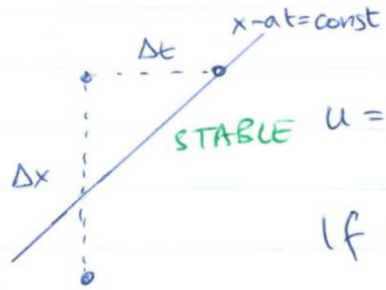
The method is stable if $C = \frac{a\Delta t}{\Delta x} < 1$

Courant

~~Cart~~- Friedrichs-Lewy condition

C is called the Courant number

We need $\Delta t < \frac{\Delta x}{a}$ ← need sufficiently small Δt.

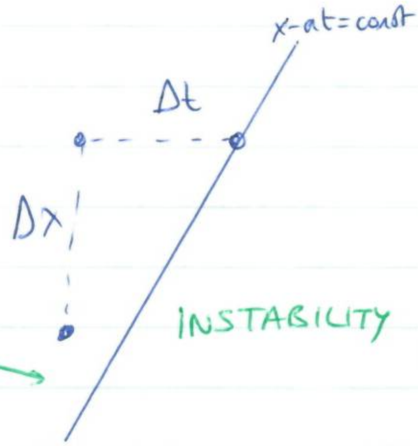


$u = u(x-at)$ is exact.

If $C<1 \Rightarrow \frac{\Delta x}{\Delta t} > a$

$\Rightarrow \frac{\Delta x}{\Delta t} < a$

information comes in this way. You want this info to be between your nodes.

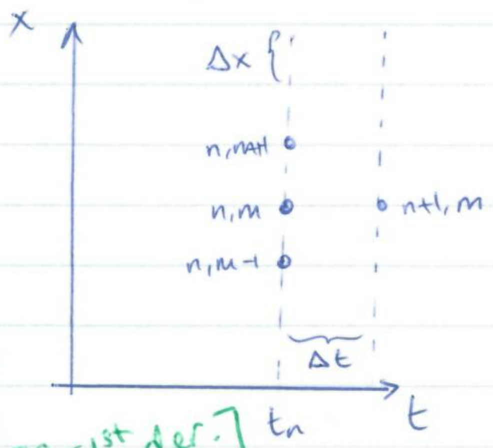


fails here ⇒ INSTABILITY

Parabolic equations

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

Try forward 1^{st}-order in t
central 2^{nd}-order in x



First order forward [for 1st der.]

Second order central [for 2nd der.]

$$\frac{u_{n+1,m}-u_{n,m}}{\Delta t} = \frac{u_{n,m+1}-2u_{n,m}+u_{n,m-1}}{(\Delta x)^2}$$

$$u_{n+1,m} = u_{n,m}\left(1-2\frac{\Delta t}{(\Delta x)^2}\right) + u_{n,m+1}\frac{\Delta t}{(\Delta x)^2} + u_{n,m-1}\frac{\Delta t}{(\Delta x)^2}$$

Stability: $u_{n,m} = \lambda^n e^{i\omega m}$

$$\Rightarrow \lambda = 1-2\frac{\Delta t}{(\Delta x)^2} + e^{i\omega}\frac{\Delta t}{(\Delta t)^2} + e^{-i\omega}\frac{\Delta t}{(\Delta x)^2}$$

Let $C = \frac{\Delta t}{(\Delta x)^2}$

$$\Rightarrow \lambda = 1-2C+C\,2\cos\omega$$
$$= 1+2C(\cos\omega-1)$$
$$= 1-4C\sin^2\frac{\omega}{2}$$

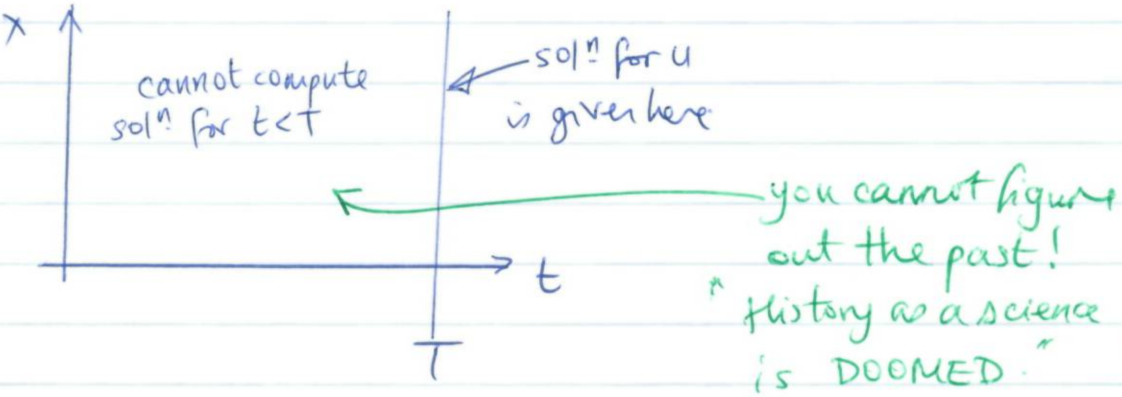
Stability if $|\lambda|<1$, ie $C>0$. for $\lambda<1$ $(\forall\omega)$

$$2>4C\sin^2\frac{\omega}{2} \quad \text{for } \lambda>-1$$
$$1>2C\sin^2\frac{\omega}{2}$$
$$\Rightarrow C<\frac{1}{2}$$

$\Rightarrow$ stable if $\underline{0<C<\frac{1}{2}}$.

ie $\Delta t < \frac{1}{2}(\Delta x)^2$

Condition $C > 0$
$\Rightarrow \Delta t > 0$



Exercise: Same in a finite x-range

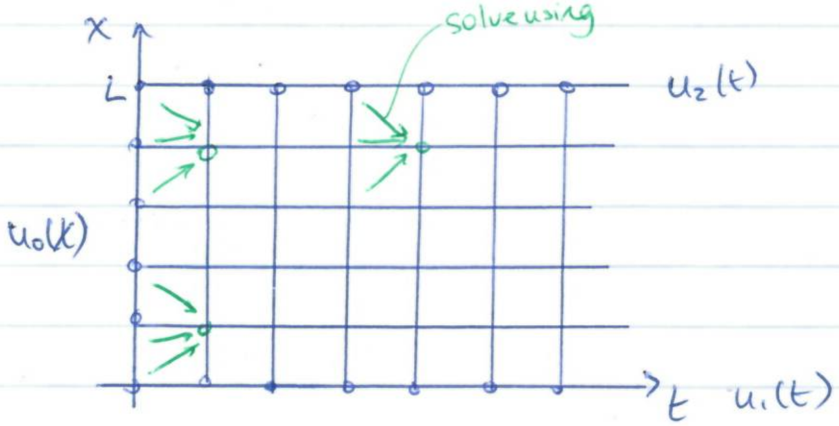
$u_t = u_{xx}$

$u|_{t=0} = u_0(x)$ initial condition

$0 \le x \le L$ $\qquad$ $u|_{x=0} = u_1(t)$

$\qquad\qquad\qquad\quad$ $u|_{x=L} = u_2(t)$

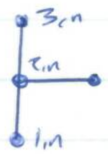
$u_0, u_1, u_2(t)$ given.



No extra work to be done.

__Exercise:__ If at $x=0$, b.c. is

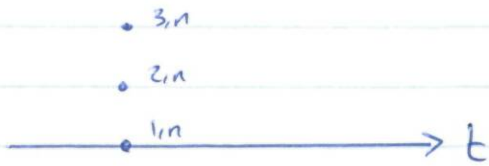
$$\left.\frac{\partial u}{\partial x}\right|_{x=0} = q(t)$$

Use the same stencil  2nd order in x

$\Rightarrow$ need 2nd-order approximation for the equation.

$$\left.\frac{\partial u}{\partial x}\right|_{x=0} = q(t)$$

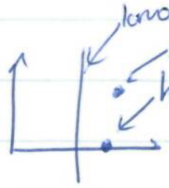
At $t=t_n$,



$$\frac{3u_{3,n} - 4u_{2,n} + u_{1,n}}{2\Delta x} = q_n \qquad (*)$$

Use this together with approximated equation

$u_{2,n} = f(u_{n-1})$, known

[the problem is ]

use the ~~3, ...~~ eq^n $(*)$.

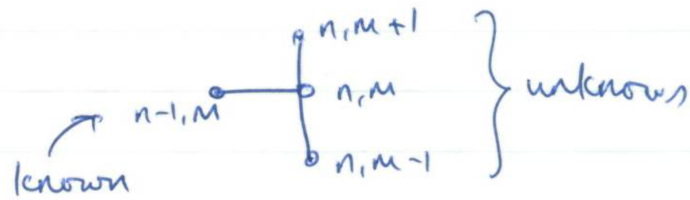
$u_{3,n} = f(u_{n-1})$, known

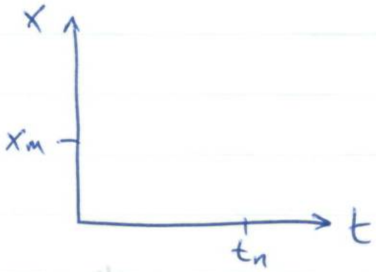
$\Rightarrow$ solve to find $u_{1,n}$

Similar question: $u_t + a u_x = u_{xx}$ ← highest derivative determines nature of problem.

<u>Implicit for parabolic equations</u>

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





back — central

$$\frac{u_{n,m} - u_{n-1,m}}{\Delta t} = \frac{u_{n,m+1} - 2u_{n,m} + u_{n,m-1}}{(\Delta x)^2}$$

$$-\frac{\Delta t}{(\Delta x)^2} u_{n,m+1} + \left[1 + 2\frac{\Delta t}{(\Delta x)^2}\right] u_{n,m}$$

$$-\frac{\Delta t}{(\Delta x)^2} u_{n,m-1} - u_{n-1,m} = 0$$

Let $C = \frac{\Delta t}{(\Delta x)^2}$

known

$u_{n-1,m} = -C(u_{n,m-1}) + (1+2C)u_{n,m} - C(u_{n,m+1})$

$$-C u_{n,m+1} + (1+2C) u_{n,m}$$
$$-C u_{n,m-1} - u_{n-1,m} = 0$$

$$u_{n,m} = \lambda^n e^{im\omega}$$

$$-Ce^{i\omega} + (1+2C) - Ce^{-i\omega} - \lambda^{-1} = 0$$

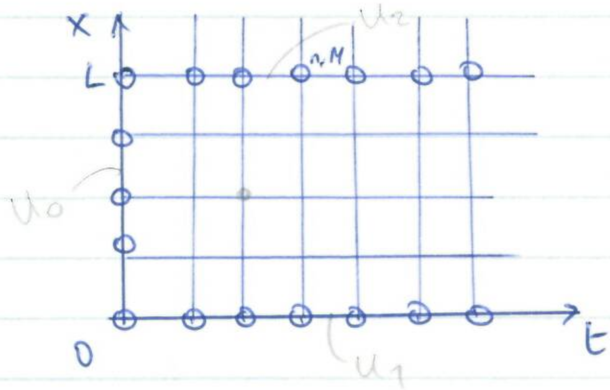
$$\lambda^{-1} = 1 + 2C - 2C\cos\omega$$

$$= 1 + 2C\,2\sin^2\frac{\omega}{2}$$

$$\lambda = \frac{1}{1+4C\sin^2\frac{\omega}{2}} < 1 \qquad \text{unless } \omega = n\pi$$

$\Rightarrow$ unconditionally stable.

Suppose we are solving $u_t = u_{xx}$



$L = M \cdot \Delta x$

with b.c.s $\quad u|_{x=0} = u_1(t)$

$\qquad\qquad\quad u|_{x=L} = u_2(t)$

and initial conditions $u|_{t=0} = u_0(x)$

To find $u_{n,m}$ at time t_n, write as a matrix equation

$$\begin{pmatrix} 1 & 0 & 0 & 0 & & \\ -C & 1+2C & -C & & & \\ 0 & -C & 1+2C & -C & & \\ 0 & 0 & -C & 1+2C & -C & \\ & & \vdots & & & \\ 0 & 0 & 0 & 0 & & 1 \end{pmatrix} \begin{pmatrix} u_{n,1} \\ u_{n,2} \\ u_{n,3} \\ \vdots \\ u_{n,M} \end{pmatrix} = \begin{pmatrix} u_1(t_n) \\ u_{n-1,2} \\ u_{n-1,3} \\ \\ u_2(t_n) \end{pmatrix}$$

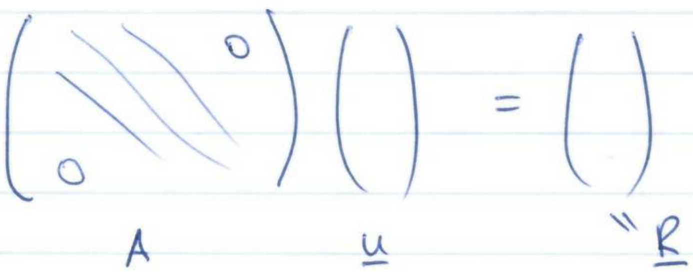
Simpler, $u_{n,1} = u_1(t_n)$
$u_{n,M} = u_2(t_n)$

for $u_{n,m}$, $2 \le m \le M-1$

$$
\begin{pmatrix}
-c & 1+2c & -c & 0 & 0 & 0 \\
0 & -c & 1+2c & -c & 0 & 0 \\
0 & 0 & -c & 1+2c & -c & 0 \\
 & & & & & \\
 & & & & &
\end{pmatrix}
\begin{pmatrix}
u_1(t_n) \\ u_{n,2} \\ u_{n,3} \\ \vdots \\ u_2(t_n)
\end{pmatrix}
=
\begin{pmatrix}
 \\ \\ \\ \\ \\
\end{pmatrix}
$$

↑ three-diagonal matrix

System of tri-diagonal kind

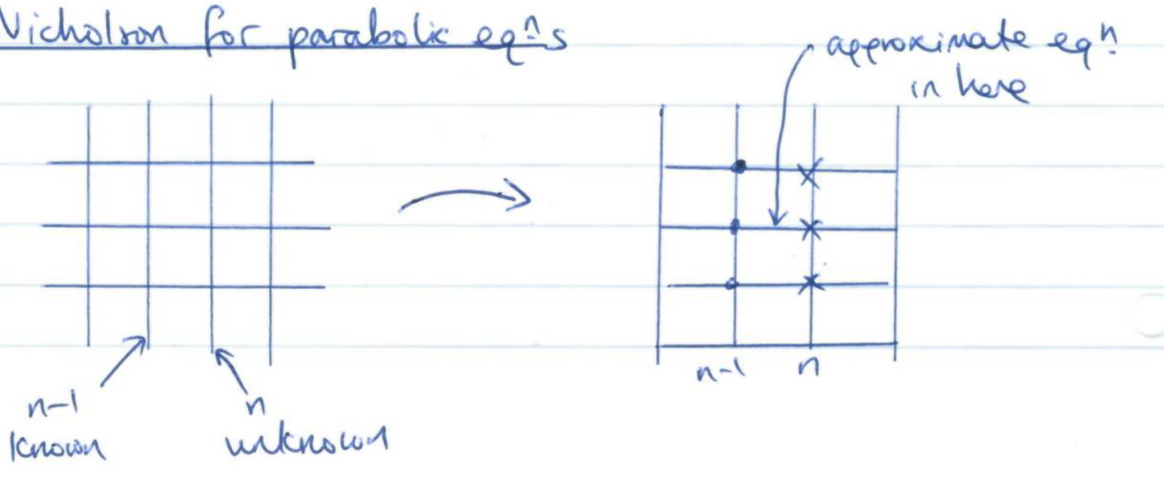


$A \underline{u} = \underline{R}$

$\underline{u} = \underline{A}^{-1} \underline{R}$. Alternatively use Thomas algorithm

Crank-Nicholson for parabolic eq^ns

Idea:

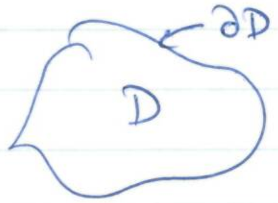


For $u_t = u_{xx}$

$$\frac{u_{n,m} - u_{n-1,m}}{\Delta t} = \frac{1}{2}\left[\frac{u_{n,m+1} - 2u_{n,m} + u_{n,m-1}}{(\Delta x)^2} + \frac{u_{n-1,m+1} - 2u_{n-1,m} + u_{n-1,m-1}}{(\Delta x)^2}\right]$$

Elliptic equations

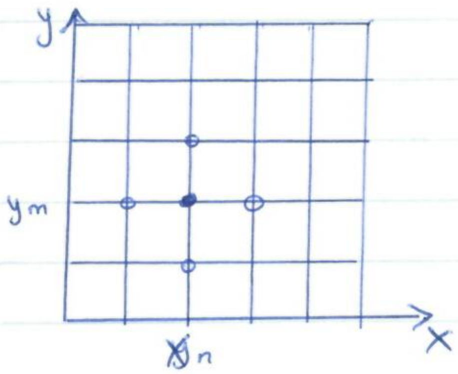
Classic example: $\nabla^2 u = 0$,



solve in D

with $u(x,y)$ given on ∂D.

Example: D is a square domain



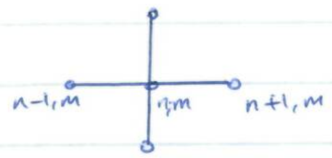
Split into square grid

$x_n = n \cdot h$

$y_n = n \cdot h$

Write approximation at (x_n, y_m)

use this stencil $\longrightarrow$



$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0.$$

second central

$$\Rightarrow 0 = \frac{u_{n+1,m} - 2u_{n,m} + u_{n-1,m}}{h^2} + \frac{u_{n,m+1} - 2u_{n,m} + u_{n,m-1}}{h^2}$$

Note: have fixed $\delta x = \delta y = h$ but we can have $\delta x \neq \delta y$.

$$\Rightarrow \quad u_{n+1,m} + u_{n-1,m} - 4u_{n,m} + u_{n,m+1} + u_{n,m-1} = 0$$

Write as $\quad A\underline{u} = \underline{0}$.

but $\underline{u}$ is a column vector – how do we fit $u_{n,m}$ into it?

We could label them $\begin{smallmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & \cdots \end{smallmatrix}$ etc and say

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & -4 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \vdots \\ u_{n,m+1} \\ \vdots \\ u_{n-1,m} \\ u_{n,m} \\ u_{n+1,m} \\ \vdots \\ u_{n,m-1} \\ \vdots \end{pmatrix} = \underline{0}$$

Instead of inverting A, we can use iterations.

Let ℓ = nº of iteration.

Suppose $u^{\ell}_{n,m}$ is an approximation known at iteration ℓ.
We need a rule to update $u^{\ell+1}_{n,m}$.

Let $\boxed{u^{\ell}_{n+1,m} + u^{\ell}_{n-1,m} - 4u^{\ell+1}_{n,m} + u^{\ell}_{n,m+1} + u^{\ell}_{n,m-1} = 0}$ ← Jacobi iterations (but only for this approximation!) (†)

ie $u^{\ell+1}_{n,m}$ is $\frac{1}{4}$ of the sum of u's at neighbouring pts at iteration ℓ.

Question: do the iterations converge?

ie does $\lim_{l \to \infty} u^l_{n,m}$ exist?

We write $u^l_{n,m} = \rho^l e^{i(k_1 n + k_2 m)}$ $\qquad (*)$

with real k_1, k_2 and need to find ρ.
and assume zero boundary conditions.

Substitute $(*)$ into $(+)$:

$$e^{ik_1} + e^{-ik_1} - 4\rho + e^{-ik_2} + e^{ik_2} = 0,$$

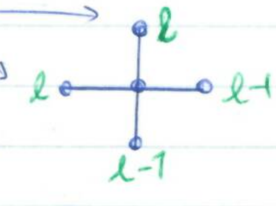
$$2\cos k_1 - 4\rho + 2\cos k_2 = 0$$

$$\Rightarrow \quad \rho = \tfrac{1}{2}(\cos k_1 + \cos k_2)$$

$\Rightarrow \quad |\rho| < 1$ unless both $\cos k_1 = \cos k_2 = 1$

ie Iterations converge. but convergence is <u>slow</u>.

But we can improve this method by using values we have already found in that iteration



Get

Gauss-Seidel iterations

$$\boxed{u^l_{n+1,m} + u^{l+1}_{n-1,m} - 4u^{l+1}_{n,m} + u^{l+1}_{n,m+1} + u^l_{n,m-1} = 0}$$

but again only for this approximation

if computing from top-left

Does this converge?

Again $u^{l}_{n,m} = \rho^{l} e^{i(k_1 n + k_2 m)}$.

Substitute in to get

$$e^{ik_1} + \rho e^{-ik_1} - 4\rho + e^{-ik_2} + \rho e^{ik_2} = 0$$

$$\Rightarrow \rho\left[e^{-ik_1} + e^{ik_2} - 4\right] = -e^{ik_1} - e^{-ik_2}$$

$$\Rightarrow \rho = \frac{e^{ik_1} + e^{-ik_2}}{4 - (e^{-ik_1} + e^{ik_2})}$$

Clearly $|\rho| < 1$ since top is at most 2, bottom at least 2. To tidy this up though, if $z = e^{ik_1} + e^{-ik_2}$,

$$\rho = \cancel{\frac{z}{4-\bar{z}}}$$ let's not do this.

$$= \cancel{\frac{z(4-z)}{4^2 - |z|^2}}$$

$$\Rightarrow \cancel{|\rho|} = \cancel{\frac{4|z| - |z|^2}{4^2 - |z|^2}}$$

$$|\rho| = \left|\frac{z}{4-\bar{z}}\right|$$

$|z|<2$

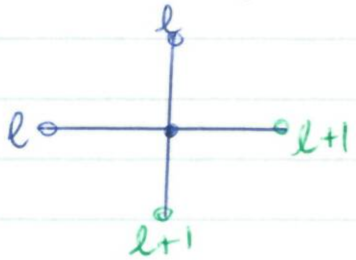
$$\left|\frac{1}{\rho}\right| = \left|\frac{4-\bar{z}}{z}\right| \quad \Rightarrow \quad 1$$

$$= \left|\frac{4}{z} - \frac{\bar{z}}{z}\right| \leq \left|\frac{4}{z}\right| + \left|\frac{\bar{z}}{z}\right|$$

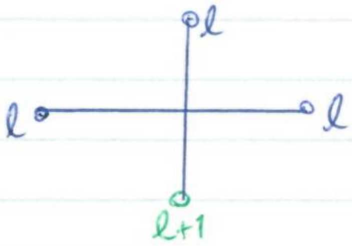
$|1|$?!

Convergence you would expect to be faster but you can't see it here.

Other versions of Gauss-Seidel



computing from bottom-right



updating in rows.

Exercise: Solve $u_{xx} + u_{yy} = u_x + u_y$

where u is given on the boundaries of a square domain

or Solve $u_{xx} + u_{yy} = u_x + u_y + u^{102}$

ie $-4u_{nm}^{l+1} + \cdots = (u_{nm}^{102})^{l}$

use iterations like this.