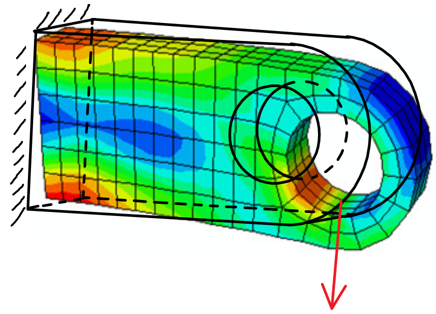
Governing equations in 3D:



$$(S)\begin{cases}
\operatorname{div} \underset{\sim}{S} + \underset{\sim}{b} = \underset{\sim}{0} & \text{— (EQ)} \quad \forall \underset{\sim}{x} \in \phi(B) \\
\underset{\sim}{S} = \underset{\sim}{S}^T & \forall \underset{\sim}{x} \in \phi(B) \\
\underset{\sim}{\epsilon} = \tfrac{1}{2}(\nabla \underset{\sim}{u} + \nabla \underset{\sim}{u}^T) & \text{— (SD)} \quad \forall \underset{\sim}{x} \in \phi(B) \\
\underset{\sim}{S} = \lambda\, tr(\underset{\sim}{\epsilon})\, \underset{\sim}{I} + 2\mu \underset{\sim}{\epsilon} & \text{— (SS)} \quad \forall \underset{\sim}{x} \in \phi(B) \\
\underset{\sim}{u} = \underset{\sim}{u}_D & \forall \underset{\sim}{x} \in \phi(A_D) \\
\underset{\sim}{t}_{(n)} = \underset{\sim}{S}\, \underset{\sim}{n} = \underset{\sim}{h}_N & \forall \underset{\sim}{x} \in \phi(A_N)
\end{cases}$$

(W)

(PVW)

Find displacement field $\boldsymbol{u}(\boldsymbol{x})$, strain field $\boldsymbol{\varepsilon}(\boldsymbol{x})$, and stress field $\mathbf{S}(\boldsymbol{x})$; $\underset{\sim}{u}(\underset{\sim}{x}) \in H^1(\phi(B))$

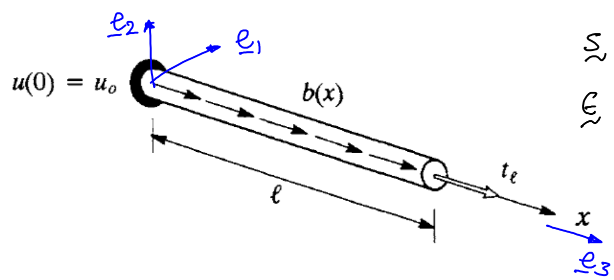
that satisfy SD and SS relationships above for all $\boldsymbol{x}$, and $\boxed{\boldsymbol{u}(\boldsymbol{x}) = \boldsymbol{u}_D \text{ on } \phi(A_D)}$ EBC

and $\quad G(\underset{\sim}{u}, \bar{\underset{\sim}{u}}) = 0 \quad \forall \quad \bar{\underset{\sim}{u}}(\underset{\sim}{x}) \in H^1_E(\phi(B))$

where $$\boxed{G(\underset{\sim}{u}, \bar{\underset{\sim}{u}}) = \underbrace{\int_{\Omega=\phi(B)} \bar{\underset{\sim}{\epsilon}} : \underset{\sim}{S}\, d\Omega}_{W_I} \underbrace{- \int_{\Omega=\phi(B)} \bar{\underset{\sim}{u}} \cdot \underset{\sim}{b}\, d\Omega - \int_{\Gamma_N=\phi(A_N)} \bar{\underset{\sim}{u}} \cdot \underset{\sim}{t}_{(n)}\, d\Gamma}_{-W_E}}$$

Simplified (little) BVP in 1D:

$$(S)\begin{cases}
\sigma' + b = 0 & \forall x \in (0, \ell) \\
u(0) = u_0 & @\ x = 0 \\
t_L = \sigma(\ell)(+1) & @\ x = L
\end{cases}$$



$\underset{\sim}{S} = \sigma\, (\underset{\sim}{e}_3 \otimes \underset{\sim}{e}_3)$

$\underset{\sim}{\epsilon} \sim \begin{bmatrix} \epsilon_{11} & & \\ & \epsilon_{22} & \\ & & \epsilon_{33} \end{bmatrix}$

$\epsilon_{33} = u'$

$\sigma = C u'$

PVW

(W) :

$$\begin{cases}
\text{Find } u(x) \in H^1(0,\ell) \text{ such that } \boxed{u(0) = u_0} \text{ EBC} \\
\text{and } G(\sigma, \bar{u}) = 0 \quad \forall \quad \bar{u}(x) \in H^1_E(0,\ell) \\
\text{where } G(\sigma, \bar{u}) \equiv \underbrace{\int_0^\ell \bar{u}' \sigma\, dx}_{W_I} - \underbrace{\left[\int_0^\ell \bar{u}\, b\, dx + \bar{u}(\ell)\, t_\ell \right]}_{W_E}
\end{cases}$$

Recall Big Picture:

(E) $\quad \Pi(\underset{\sim}{u}) \quad \underset{\text{VAINBERG}}{\overset{D\Pi(\underset{\sim}{u}) \cdot \bar{\underset{\sim}{u}}}{\rightleftarrows}} \quad$ PVW (W) $\quad \underbrace{G(\underset{\sim}{u}, \bar{\underset{\sim}{u}}) = 0 \ \ \forall \bar{\underset{\sim}{u}}}$ $\quad \underset{\text{MWR + IBP}}{\overset{\text{RIBP + FTCV}}{\rightleftarrows}} \quad$ (S) $\quad \operatorname{div} \underset{\sim}{S} + \underset{\sim}{b} = \underset{\sim}{0}$

$(Cu')' + b = 0$

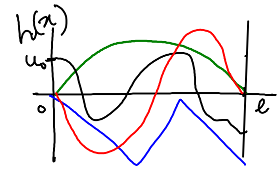
APPROXIMATION

$G^h(\underset{\sim}{u}^h, \bar{\underset{\sim}{u}}^h) = 0 \quad \forall \bar{\underset{\sim}{u}}^h$

$\Rightarrow \quad \bar{\underset{\sim}{a}}^T \left[\underset{\sim}{K}\, \underset{\sim}{a} - \underset{\sim}{f} \right] = 0 \quad \forall\ \bar{\underset{\sim}{a}} \quad$ (G)

(W) form of the problem is still infinite dimensional.

Introduce approximation:
(Assume a certain form of the solution)



$$u(x) \cong u^h(x) = \sum_{i=1}^{N} a_i\, h_i(x) + h_0(x)$$

Discrete unknown

$$= [a_1 \; a_2 \; \cdots \; a_N] \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix}$$

$$\boxed{u(x) \cong u^h(x) = \underline{a}^T \underline{h}(x) + h_0(x)}$$

h: smooth enough
: complete

$\{h_i\}_{i=1:N} \subset H^1_E(0,\ell)$

(basis)
(shape)

Note: Essential BC $u(0) = u_0$ is satisfied by $h_0(x)$

Examples of $h_i(x)$:

- Polynomials $\left\{1, \frac{x}{\ell}, \left(\frac{x}{\ell}\right)^2, \ldots\right\}$
- Trigonometric $\left\{1, \sin\left(n\pi\frac{x}{\ell}\right), \cos\left(n\frac{\pi x}{\ell}\right)\right\}$ $n = 1, 2, 3 \ldots$
- Piecewise Polynomial (FE)

What about $\bar{u}(x)$?

Galerkin Approximation $\longrightarrow$ $\bar{u}(x) \in H^1_E(0,\ell)$ {same space as $u(x)$}

{Petrov-Galerkin: if different space}

$$\bar{u}(x) \cong \bar{u}^h(x) = \sum_{i=1}^{N} \bar{a}_i\, h_i(x)$$

arbitrary

$$= [\bar{a}_1 \; \bar{a}_2 \; \cdots \; \bar{a}_N] \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix}$$

$$\boxed{\bar{u}(x) \approx \bar{u}^h(x) = \underline{\bar{a}}^T \underline{h}(x)}$$

Note for ALL $\{\bar{a}_i\}$

Substitute the approximations:

Discretized Galerkin Form:

(G)

Find $\underline{a} = \{a_1 \; a_2 \; \cdots \; a_N\}$

such that:

$$\tilde{G}^h(\underline{a}, \underline{\bar{a}}) \equiv \int_0^\ell \underbrace{(\underline{\bar{a}}^T \underline{h})'}_{\bar{u}^{h\prime}}\, C\, \underbrace{(\underline{a}^T \underline{h} + h_0)'}_{u^{h\prime}} dx - \int_0^\ell \underbrace{(\underline{\bar{a}}^T \underline{h})}_{\bar{u}^h}\, b\, dx - \underbrace{(\underline{\bar{a}}^T \underline{h}(\ell))}_{\bar{u}^h(\ell)} (t_\ell)$$

$G^h(\underline{a}, \underline{\bar{a}}) = 0$ FOR ALL $\underline{\bar{a}}$

This is called the discretized Galerkin Form (G)

Note

$$(S) \Longleftrightarrow (W) \overset{\text{Approx}}{\approx} (G)$$

Upon simplification:

$$\tilde{G}^h(\underline{a},\underline{\bar{a}}) = \underbrace{\underline{\bar{a}}^T \left[\int_0^l C \underline{h}' \underline{h}'^T dx \right] \underline{a}}_{\underline{\bar{a}}^T \underset{\sim}{K} \underline{a}} + \underline{\bar{a}}^T \left\{ \int_0^l \underline{h}' C h_0' \, dx \right\} \underbrace{- \underline{\bar{a}}^T \left\{ \int_0^l \underline{h} \, b \, dx \right\} - \underline{\bar{a}}^T \left\{ \underline{h}(l) \, t_l \right\}}_{-\underline{\bar{a}}^T \underline{f}}$$

ie.

$$\tilde{G}^h(\underline{a},\underline{\bar{a}}) = \underline{\bar{a}}^T \underbrace{\left(\underset{\sim}{K} \underline{a} - \underline{f} \right)}_{\underline{g}(\underline{a})} = 0$$

← residual vector (function of $\underline{a}$)

Note:

- This equation would be satisfied FOR ALL $\underline{\bar{a}}^T$

  if $\boxed{\underline{g}(\underline{a}) \equiv \left(\underset{\sim}{K} \underline{a} - \underline{f} \right) = \underline{0}}$

- For linear problems, this may be simply solved as:

$$\underline{a} = [\underset{\sim}{K}]^{-1} \underline{f}$$

- For Non-linear problems, we must use a Newton-type iterative method: $\boxed{\underline{g}(\underline{a}) = \left(\underline{f}_{int}(\underline{a}) - \underline{f} \right) = \underline{0}}$

(i) Initialize iterations $i=0$

Assume a solution $\underline{a}_0$

(ii) Loop until $\left\| \dfrac{\underline{g}(\underline{a}_i)}{\underline{g}_0} \right\| < \epsilon_{tol}$ AND $\left\| \dfrac{\Delta \underline{a}_i}{\underline{a}_i} \right\| < \epsilon_{tol}$

$i \leftarrow i+1$

$$\underline{g}(\underline{a}_{i+1}) \approx \underline{g}(\underline{a}_i) + \left[\frac{\partial \underline{g}(\underline{a}_i)}{\partial \underline{a}} \right] \left\{ \Delta \underline{a}_i \right\}$$

iteration

We want

$$\underline{g}(\underline{a}_{i+1}) \longrightarrow \underline{0} \quad \Rightarrow \quad \boxed{\left\{ \Delta \underline{a}_i \right\} = \left[\frac{\partial \underline{g}(\underline{a}_i)}{\partial \underline{a}} \right]^{-1} \left\{ -\underline{g}(\underline{a}_i) \right\}}$$

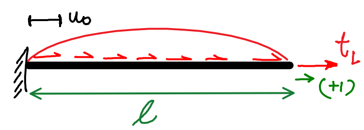
$$\underline{a}_{i+1} = \underline{a}_i + \Delta \underline{a}_i$$

(iii) Report convergence and approximate solution $\underline{a}_i$.

Example: 1D Problem:

$$\textcircled{S}\ \begin{cases} (C u')' + b = 0 & \forall x \in (0,\ell) \\ u = u_0 & @\ x = 0 \\ \sigma(\ell)(+1) = t_L & @\ x = L \end{cases}$$

C: const.



Say $b(x) = b_0 \sin\left(\frac{x\pi}{L}\right)$

Note: Exact solution: $\sigma(\ell) - \sigma(x) = \int_x^\ell -b(x)\,dx = \int_x^\ell -b_0 \sin\left(\frac{x\pi}{L}\right)dx$

$$\Rightarrow \sigma(x) = -\frac{b_0 L}{\pi}\left[\cos\left(\frac{x\pi}{L}\right)\right]_0^\ell + t_L = \boxed{\frac{b_0 L}{\pi}\left(\cos\left(\frac{x\pi}{L}\right)+1\right) + t_L}$$

$$u(x) - u(0) = \int_0^x \frac{1}{C}\,\sigma(x)\,dx \Rightarrow u(x) = \boxed{u_0 + \frac{b_0 L}{\pi C}\left(\frac{L}{\pi}\sin\left(\frac{x\pi}{L}\right) + x\right) + \frac{t_L x}{C}}$$

Recall:

PVW $\textcircled{W}$:

Find $u(x) \in H^1(0,\ell)$ such that $\boxed{u(0) = u_0}$ EBC

and $G(\sigma, \bar{u}) = 0 \quad \forall\ \bar{u}(x) \in H^1_E(0,\ell)$

where $G(\sigma,\bar{u}) \equiv \underbrace{\int_0^\ell \bar{u}'\sigma\,dx}_{W_I} - \underbrace{\left[\int_0^\ell \bar{u}\,b\,dx + \bar{u}(\ell)\,t_\ell\right]}_{W_E}$

$\textcircled{G}$

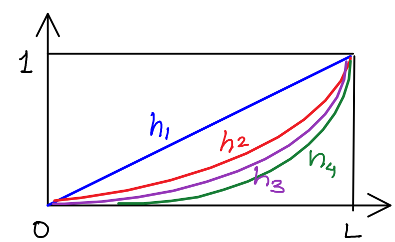
Assume $u(x) \cong h_0(x) + \underline{a}^T \underline{h}(x)$; $\bar{u}(x) \cong \underline{\bar{a}}^T \underline{h}(x)$

Find $\underline{a}$ such that $G^h(\underline{a},\underline{\bar{a}}) = \underline{\bar{a}}^T\boxed{\left(\underset{\sim}{K}\,\underline{a} - \underline{f}\right)} = 0 \quad \forall\ \underline{\bar{a}}$

where $\underset{\sim}{K} = \int_0^\ell C\,[\underline{h}'][\underline{h}'^T]\,dx$; $\underline{f} = \int_0^\ell \underline{h}\,b\,dx + \underline{h}(\ell)\,t_L$

Let $h_0(x) = u_0 \Rightarrow h_0' = 0$

$h_i(x) = \left(\frac{x}{L}\right)^i \Rightarrow h_i'(x) = \frac{i}{L}\left(\frac{x}{L}\right)^{i-1}$



Thus

$$\underset{\sim}{K} = C\int_0^\ell \begin{bmatrix} h_1'(x) \\ h_2'(x) \\ \vdots \\ h_N'(x) \end{bmatrix} \begin{bmatrix} h_1'(x) & h_2'(x) & \cdots & h_N'(x) \end{bmatrix} dx$$

i, j, K_{ij}

$$\Rightarrow K_{ij} = C\int_0^\ell \frac{i}{L}\left(\frac{x}{L}\right)^{i-1} \cdot \frac{j}{L}\left(\frac{x}{L}\right)^{j-1} dx = \frac{C\,ij}{L^{(i+j)}}\left[\frac{(x)^{i+j-1}}{(i+j-1)}\right]_0^L = \boxed{\frac{C}{L}\frac{ij}{(i+j-1)}}$$

$$\text{and } \underline{f} = \int_0^\ell \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix} b_0 \sin\left(\frac{x\pi}{L}\right)dx + \begin{bmatrix} h_1(L) \\ h_2(L) \\ \vdots \\ h_N(L) \end{bmatrix} t_L \Rightarrow f_i = \boxed{\frac{b_0 L}{\pi} I_i + t_L}$$

i, f_i

where $I_i = 1 - \left(\frac{i^2 - i}{\pi^2}\right) I_{i-2}$

and $I_0 = 2$; $I_1 = 1$

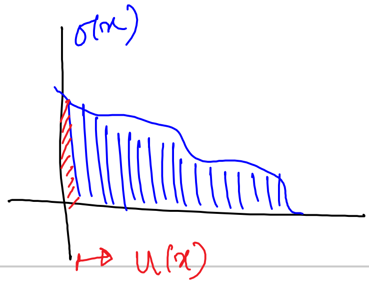
Matlab code for 1D Ritz method

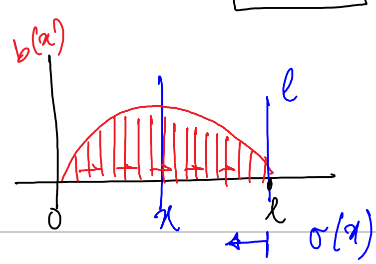

```
%% General 1D Problem fixed at x=0; free at x=L;
% For constant Area of cross-section = 1;
% BUT possibly varying Young's modulus 'C(x)'

clear all; clc; close all; % clear all the existing variables (new start)

% Allow the user to specify the problem parameters:
name='Problem Parameters';
prompt={'Length of the bar (Len):', ...
    'Specified displacement at x=0: (u0):', ...
    'End traction (tL):', ...
    'Youngs Modulus Cmod(x):', ...
    'Applied body force BodyForce(x):', ...
    'Number of divisions for Trapezoidal rule (Ndiv):', ...
    'Number of polynomial terms for the Ritz method (NRitz):', ...
    'Exact Solution Stress s_exact(x):', ...
    'Exact Solution Displacement u_exact(x):'};

numlines=1;

% Default options for problem parameters - uncomment one option below

% defaultanswer={'10','0','10','@(x)1000*ones(size(x))','@(x)10*ones(size(x))','100','1:3', ...
%     '@(x)10*(Len-x)+tL','@(x)u0+1/1000*(-0.5*10*x.^2+(tL+10*Len)*x)'};

% defaultanswer={'10','1','10','@(x)1000+1000*(Len-x)/Len','@(x)10+10*sin(x/Len*pi)','100','1:3', ...
%     '@(x)zeros(size(x))','@(x)zeros(size(x))'};

defaultanswer={'10','0','10','@(x)1000*ones(size(x))','@(x)10*sin(x/Len*pi)','100','[1 2 3 4]', ...
    '@(x)10*Len/pi*(cos(x*pi/Len)+1)+tL','@(x)10*Len/(pi*1000)*(Len/pi*sin(x*pi/Len)+x)+tL/1000*x+u0'};

% Ask the user:
useranswer=inputdlg(prompt,name,numlines,defaultanswer);

Len = str2num(useranswer{1});
u0  = str2num(useranswer{2});
tL  = str2num(useranswer{3});
Cmod = eval(useranswer{4});
BodyForce = eval(useranswer{5});
Ndiv = str2num(useranswer{6});
Ritz = str2num(useranswer{7});
s_exact = eval(useranswer{8});
u_exact = eval(useranswer{9});

dL = Len/Ndiv;
x = 0:dL:Len;  % x-locations along the length of the bar

%% Exact / Reference solution

% Compute an approximate reference solution by direct numerical integration
% First obtain stress by integrating body-force from x-to-L
sx = zeros(size(x));
sx(end) = tL;
for ii = length(x)-1:-1:1
    xcenter = (x(ii)+x(ii+1))/2;
    sx(ii) = sx(ii+1) + BodyForce(xcenter)*dL;
end
% Then obtain displacement by integrating stress from 0-to-x
ux = zeros(size(x));
ux(1) = u0;
for ii = 2:length(x)
```

$$\sigma(l) - \sigma(x) = \int_x^l -b\, dx$$

$$\Rightarrow \sigma(x) = \sigma(l) + \underbrace{\int_x^l b\, dx}$$

$$u(x) - u_0 = \int_0^x \frac{1}{C}\sigma(x)\, dx$$

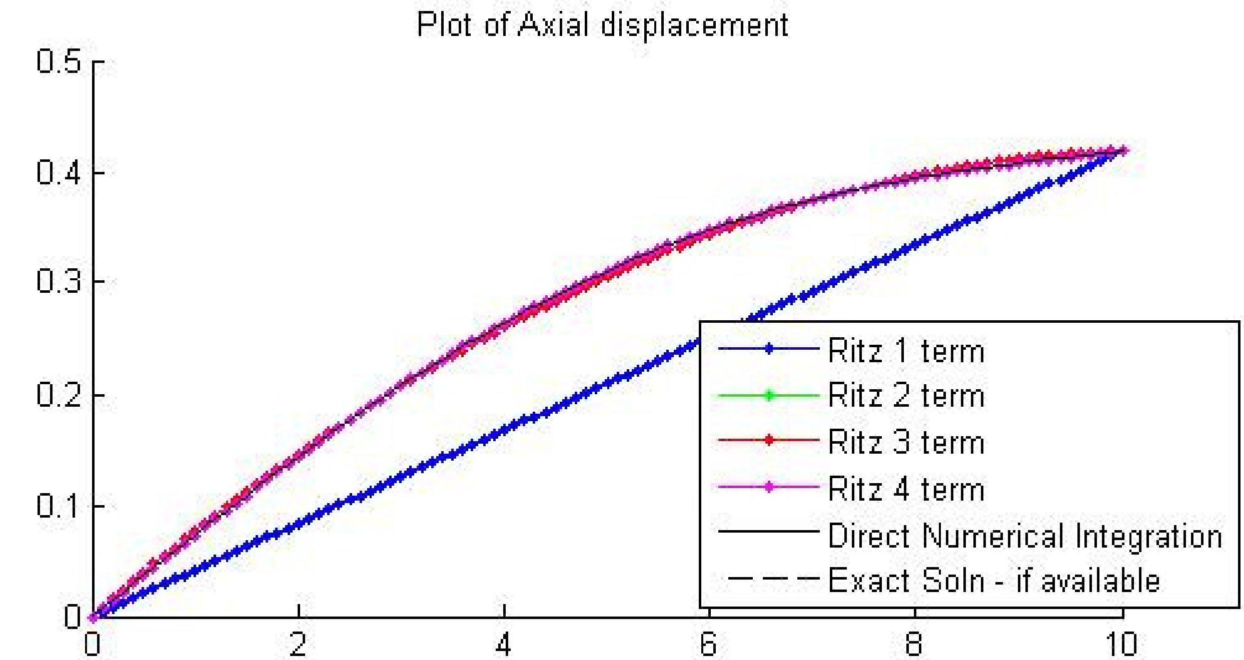




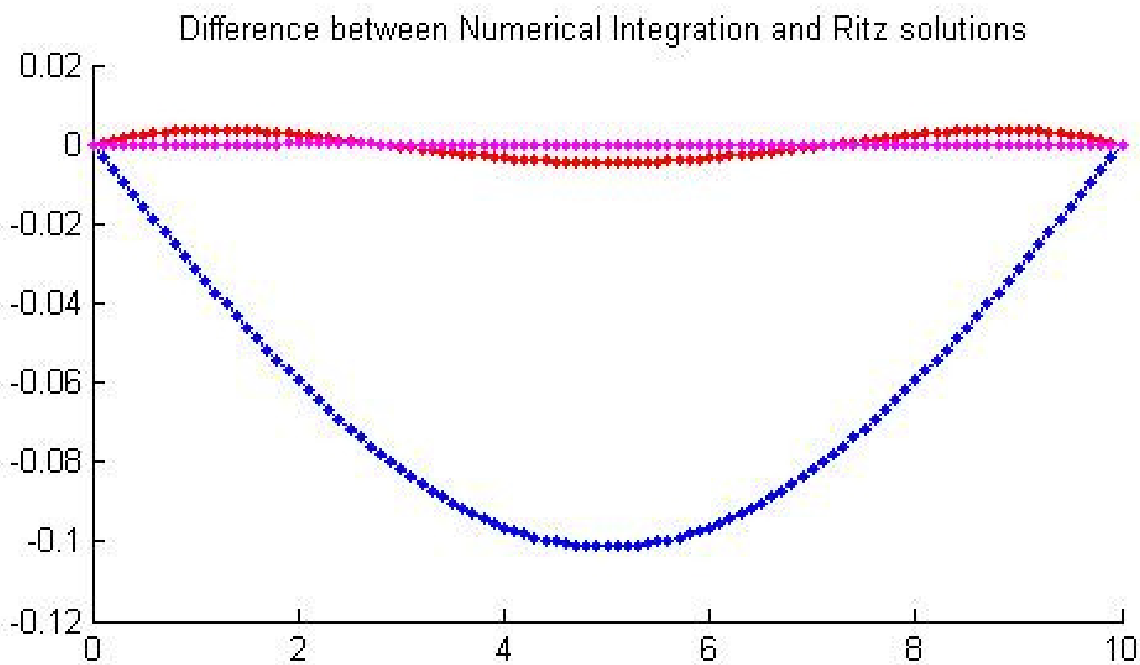
Matlab code for 1D Ritz method (continued)


```
    xcenter = (x(ii-1)+x(ii))/2;
    sxcenter = (sx(ii-1)+sx(ii))/2;
    ux(ii) = ux(ii-1) + sxcenter/Cmod(xcenter)*dL;
end

%% Compute an approximate solution using the Ritz Method
% From an n-term polynomial approximation:
% uh(x) = h0 + a1*(x/L) + a2*(x/L)^2 + a3*(x/L)^3 + ... + aN*(x/L)^N
% h0 = u0 (EBC)

legendcell={};
icount = 0;
plotstyle = {'b.-','g.-','r.-','m.-','c.-'};
for NRitz = Ritz
    icount = icount + 1;
    % Compute the Stiffness Matrix K(NxN) and Load Vector f(Nx1) by numerical integration
    KRitz = zeros(NRitz);
    fRitz = zeros(NRitz,1);
    for kk = 2:length(x)
        xcenter = (x(kk-1)+x(kk))/2;
        for ii = 1:NRitz
            for jj = 1:NRitz
                % Compute the K(ii,jj) term of the stiffness matrix by numerical integration
                KRitz(ii,jj) = KRitz(ii,jj) + ...
                    Cmod(xcenter) * (ii*xcenter^(ii-1)/(Len^ii)) * (jj*xcenter^(jj-1)/(Len^jj)) * dL;
            end
            % Compute the f(ii) term of the load vector by numerical integration
            fRitz(ii) = fRitz(ii) + (xcenter/Len)^ii*BodyForce(xcenter) * dL;
        end
    end
    fRitz = fRitz + tL;

    % Solve for the unknown coefficients:
    aRitz = KRitz \ fRitz; % Always avoid using "inv()"

    % Compute the approximate solution for plotting:
    uRitz = u0 + zeros(size(x));
    sRitz = zeros(size(x));
    for ii = 1:NRitz
        uRitz = uRitz + aRitz(ii)*(x/Len).^ii;
        sRitz = sRitz + Cmod(x) .* ( aRitz(ii)*(ii/Len^ii) * x.^(ii-1) ) ;
    end

    % Plot the Ritz solution for displacement:
    figure(1);
    subplot(2,1,1); hold on;
    plot(x,uRitz,plotstyle{mod(NRitz-1,length(plotstyle))+1});
    subplot(2,1,2); hold on;
    plot(x,uRitz-ux,plotstyle{mod(NRitz-1,length(plotstyle))+1}); title('Difference between Numerical
Integration and Ritz solutions')

    % Plot the Ritz solution for stress:
    figure(2);
    subplot(2,1,1); hold on;
    plot(x,sRitz,plotstyle{mod(NRitz-1,length(plotstyle))+1});
    subplot(2,1,2); hold on;
    plot(x,sRitz-sx,plotstyle{mod(NRitz-1,length(plotstyle))+1}); title('Difference between Numerical
Integration and Ritz solutions')

    legendcell{icount} = ['Ritz ' num2str(NRitz) ' term'];
end
```

Results

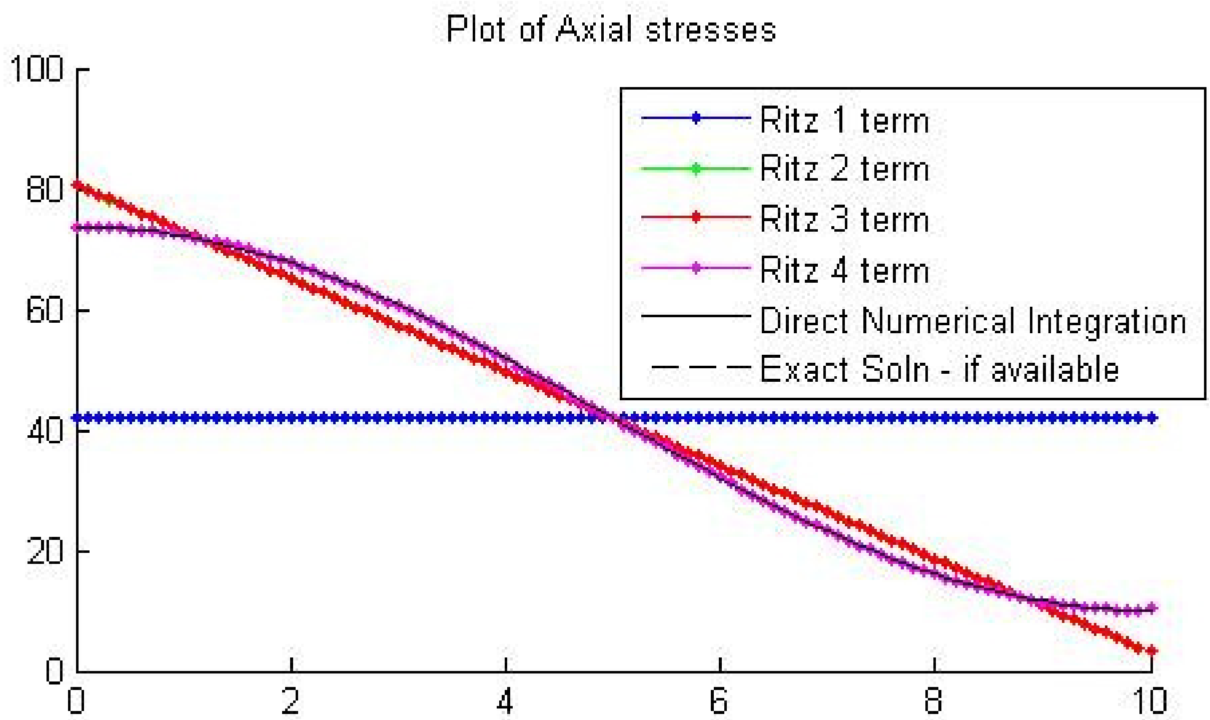


Plot of Axial displacement
0.5
0.4
0.3
0.2
0.1
0
0
2
4
6
8
10
Ritz 1 term
Ritz 2 term
Ritz 3 term
Ritz 4 term
Direct Numerical Integration
Exact Soln - if available

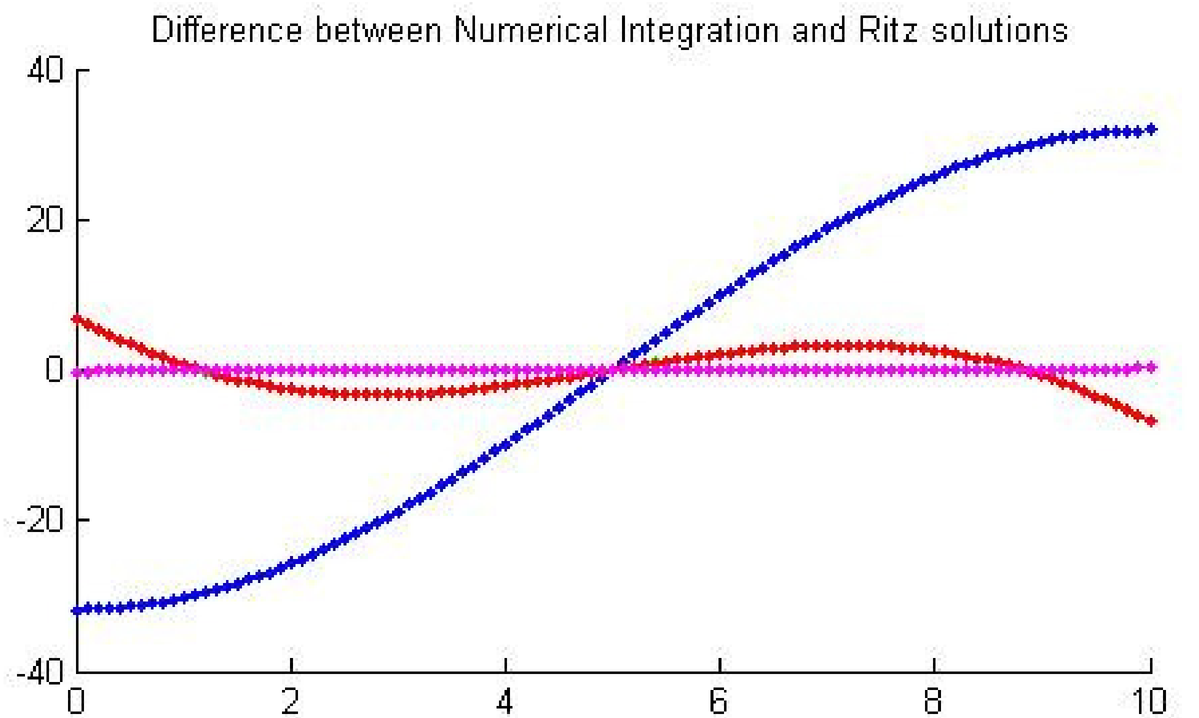


Difference between Numerical Integration and Ritz solutions
0.02
0
-0.02
-0.04
-0.06
-0.08
-0.1
-0.12
0
2
4
6
8
10

Results



Plot of Axial stresses
100
80
60
40
20
0
0
2
4
6
8
10
Ritz 1 term
Ritz 2 term
Ritz 3 term
Ritz 4 term
Direct Numerical Integration
Exact Soln - if available



Difference between Numerical Integration and Ritz solutions
40
20
0
-20
-40
0
2
4
6
8
10

Summary of the method for obtaining approximate solutions:

$$
\begin{array}{ccccc}
\text{E} & & \text{PVW (W)} & & \text{(S)} \\
\Pi(\underline{u}) & \underset{\text{VAINBERG}}{\overset{D\Pi(\underline{u})\cdot\bar{\underline{u}}}{\rightleftarrows}} & G(\underline{u},\bar{\underline{u}})=0 \;\; \forall \bar{\underline{u}} & \underset{\text{MWR + IBP}}{\overset{\text{RIBP + FTCV}}{\rightleftarrows}} & \begin{array}{l} \operatorname{div} \underset{\sim}{\sigma} + \underline{b} = \underline{0} \\ (cu')' + b = 0 \end{array}
\end{array}
$$

APPROXIMATION

$$G^h(u^h, \bar{u}^h) = 0 \quad \forall \bar{u}^h$$

$$\Rightarrow \bar{\underline{a}}^T \left[\underset{\sim}{K}\, \underline{a} - \underline{f} \right] = 0 \quad \forall \bar{\underline{a}} \qquad \text{(G)}$$

1) Take the strong form (S) of a given GDE, and apply MWR: pre-multiply it with $\bar{u}(x)$ and integrate over the domain.

2) Obtain the weak form (W) by "integrating by parts" to balance the derivatives on $u(x)$ & $\bar{u}(x)$ and add any boundary terms to get PVW (W).

Instead of starting from the strong form (S) and following 1) and 2) above, one may start from the Energy form (E):

Minimize $\Pi(\underline{u})$ to get $D\Pi(\underline{u})\cdot\bar{\underline{u}} = G(\underline{u},\bar{\underline{u}})$: PVW (W)

---

3) Define the function spaces for $u(x)$ & $\bar{u}(x)$.

4) Choose approximating functions:

$$u(x) \approx u^h(x) = \underline{a}^T\, \underline{h}(x) + h_0(x)$$

$$\bar{u}(x) \approx \bar{u}^h(x) = \bar{\underline{a}}^T\, \underline{h}(x)$$

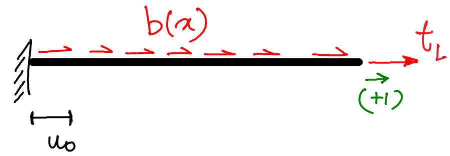
5) Derive the discretized (Galerkin) weak form in terms of matrices.

6) Solve with the Newton's method or $[\underset{\sim}{K}]\{\underline{a}\} = \{\underline{f}\}$

Derivation of weak forms

(S) Strong form $\quad Cu'' + b = 0 \quad \forall x \in (0,\ell)$

(+ some BCs)



(W) Sign convention for W_I & W_E (assume $A = 1$)

MWR: $G(u,\bar{u}) = -\int_0^\ell \bar{u}\,(Cu'' + b)\,dx - \bar{u}(\ell)\,(t_L - \sigma(\ell)) - \bar{u}(0)\,(t_0 + \sigma(0))$

Note: $t_L - \sigma(\ell)(+1) = 0$

$t_0 - \sigma(0)(-1) = 0$

Note: $\left\{ \begin{array}{l} \bar{u} \in L_2(0,\ell) = H^0 \\ u \in H^2(0,\ell) \text{ and satisfies EBC} \end{array} \right\}$

IBP

$$G(u,\bar{u}) = \int_0^\ell \bar{u}'Cu'\,dx - \left[\bar{u}Cu'\right]_0^\ell - \int_0^\ell \bar{u}\,b\,dx - \bar{u}(\ell)\left[t_L - \sigma(\ell)\right] - \bar{u}(0)\left(t_0 + \sigma(0)\right)$$

$$= \int_0^\ell \bar{u}'Cu'\,dx - \bar{u}(\ell)\cancel{\sigma(\ell)} + \bar{u}(0)\cancel{\sigma(0)} - \int_0^\ell \bar{u}\,b\,dx$$

$$- \bar{u}(\ell)\left[t_L - \cancel{\sigma(\ell)}\right] - \bar{u}(0)\left[t_0 + \cancel{\sigma(0)}\right]$$

(W) PVW: Find u, t_0 such that

$\bar{u}(0) = 0$ HEBC

$$G(u,\bar{u}) = \int_0^\ell \bar{u}'Cu'\,dx - \int_0^\ell \bar{u}\,b\,dx - \bar{u}(\ell)\,t_L - \cancel{\bar{u}(0)}\,t_0 = 0 \quad \forall \bar{u}$$

Note: $\left\{ \begin{array}{l} \bar{u}(x) \in H^1_E(0,\ell) \text{ and HEBC} \\ u(x) \in H^1(0,\ell) \text{ and satisfies EBC} \end{array} \right\}$

Approximation Spaces:

Real $u(x) \in H^1(0,\ell)$

$$\boxed{u(x)} \cong \boxed{h_0(x)} + \left[a_1 \;\; a_2 \; \cdots \; a_N\right] \boxed{\begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix}}$$

must satisfy EBC

must satisfy HEBC

Virtual $\bar{u}(x) \in H^1_E(0,\ell)$

denotes HEBC @ $x = 0$ & $x = L$

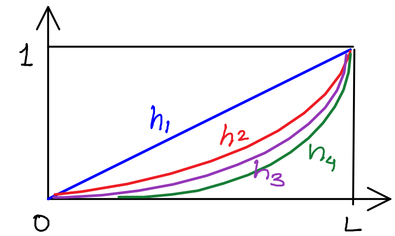
$$\boxed{\bar{u}(x)} \cong \boxed{0} + \left[\bar{a}_1 \;\; \bar{a}_2 \; \cdots \; \bar{a}_N\right] \boxed{\begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix}}$$

must satisfy HEBC

must satisfy HEBC

Approximation functions

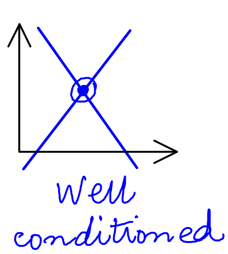
When choosing functions to approximate the real displacement $u(x)$ and the virtual displacement $\bar{u}(x)$,
Make sure that they satisfy the continuity requirements imposed by the weak form (H^1 for PVW) and,
The approximation functions $h_i(x)$ are *complete*
i.e. they can converge to the exact solution in the limit $i \to \infty$.

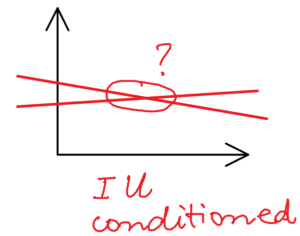


In addition, one should try to make sure that the approximation functions are sufficiently *different* from one another. If the functions are not sufficiently different, it can lead to poorly conditioned system of equations $\boldsymbol{K}\,\boldsymbol{a} = \boldsymbol{f}$

Condition number of $\boldsymbol{K}$

$$\rho(\underset{\sim}{K}) = \frac{|\text{max Eigenvalue } \lambda_{max}(\underset{\sim}{K})|}{|\text{min Eigenvalue } \lambda_{min}(\underset{\sim}{K})|}$$





If condition number is large (~10^5 or larger) the computer will not be able to solve the system accurately.
In order to keep the condition number small, we should use $h_i(x)$ functions that are linearly independent.

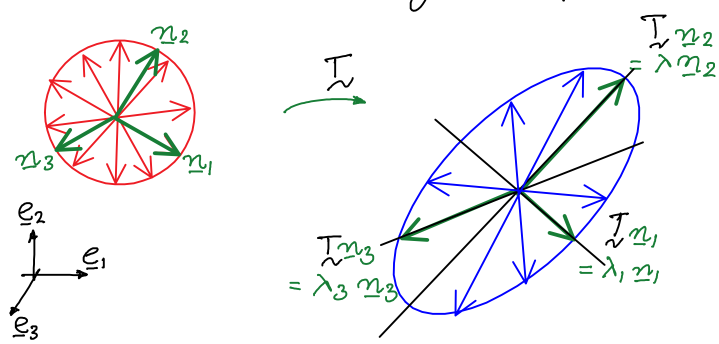
Orthogonal vectors and Orthogonal functions

Recall vectors $\underset{\sim}{v}_1, \underset{\sim}{v}_2, \underset{\sim}{v}_3$ in $\mathbb{R}^3$ are called orthogonal iff

$$\boxed{\underset{\sim}{v}_i \cdot \underset{\sim}{v}_j = 0 \quad \text{when } i \neq j}$$

Examples

- $\underset{\sim}{e}_1, \underset{\sim}{e}_2, \underset{\sim}{e}_3$
- $\underset{\sim}{n}_1, \underset{\sim}{n}_2, \underset{\sim}{n}_3$] orthonormal
- $\underset{\sim}{T}\underset{\sim}{n}_1, \underset{\sim}{T}\underset{\sim}{n}_2, \underset{\sim}{T}\underset{\sim}{n}_3$



The same concept extends to functions as well:
Given a set of functions: $h_1(x), h_2(x)$ ------ $h_N(x)$
The set is called orthogonal iff

$$\boxed{\text{Inner Product: } \langle h_i, h_j \rangle \equiv \int_0^l h_i(x) \cdot h_j(x)\,dx = 0 \quad (\text{when } i \neq j)}$$

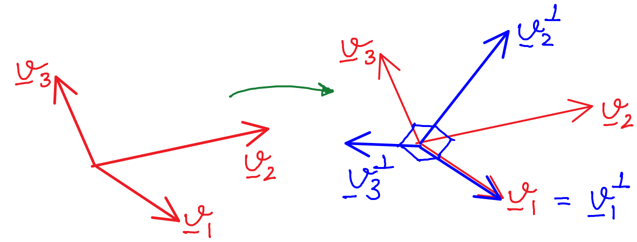
Example:
$$\int_0^{\pi} \underbrace{\sin(x)\cos(x)}_{\frac{\sin 2x}{2}}\,dx = \frac{1}{2}\cdot\frac{1}{2}\Big[-\cos 2x\Big]_0^{\pi} = -\frac{1}{4}\Big[\underset{1}{\underset{\shortparallel}{\cos 2\pi}} - \underset{1}{\underset{\shortparallel}{\cos 0}}\Big] = 0$$

$\Rightarrow \sin(x), \cos(x)$ are orthogonal over $(0, \pi)$

Gram-Schmidt orthogonalization

In order to get smaller (better) condition numbers for the system matrix $\boldsymbol{K}$, one should choose different (possibly orthogonal) approximating functions.
It is possible to generate a set of orthogonal vectors (or functions) from a given set of linearly independent vectors (or functions) which are not necessarily orthogonal to each other.

Example: Given $\underline{v}_1, \underline{v}_2, \underline{v}_3$



Let

$$\underline{v}_1^{\perp} = \underline{v}_1$$

$$\underline{v}_2^{\perp} = \underline{v}_2 - \left(\underline{v}_2 \cdot \frac{\underline{v}_1^{\perp}}{\|\underline{v}_1^{\perp}\|}\right)\frac{\underline{v}_1^{\perp}}{\|\underline{v}_1^{\perp}\|}$$

$$\Rightarrow \quad \underline{v}_2^{\perp} = \underline{v}_2 - \frac{(\underline{v}_2 \cdot \underline{v}_1^{\perp})}{(\underline{v}_1^{\perp} \cdot \underline{v}_1^{\perp})} \cdot \underline{v}_1^{\perp}$$

$$\underline{v}_3^{\perp} = \underline{v}_3 - \left(\frac{\underline{v}_3 \cdot \underline{v}_1^{\perp}}{\underline{v}_1^{\perp} \cdot \underline{v}_1^{\perp}}\right)\underline{v}_1^{\perp} - \left(\frac{\underline{v}_3 \cdot \underline{v}_2^{\perp}}{\underline{v}_2^{\perp} \cdot \underline{v}_2^{\perp}}\right)\underline{v}_2^{\perp}$$

In general, for $\mathbb{R}^n$:
(n-dimensional space)

$$\boxed{\underline{v}_j^{\perp} = \underline{v}_j - \sum_{i=1}^{j-1}\left[\left(\frac{\underline{v}_j \cdot \underline{v}_i^{\perp}}{\underline{v}_i^{\perp} \cdot \underline{v}_i^{\perp}}\right)\underline{v}_i^{\perp}\right]}$$

Note:

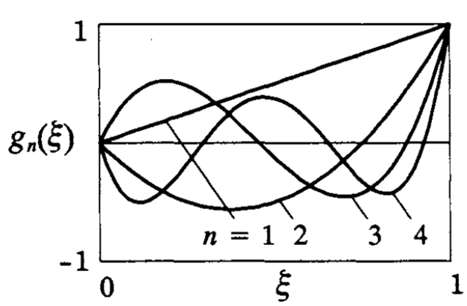
- $$\underline{v}_j^{\perp} \cdot \underline{v}_k^{\perp} = \underline{v}_j \cdot \underline{v}_k^{\perp} - \frac{(\underline{v}_j \cdot \underline{v}_k^{\perp})\cancel{(\underline{v}_k^{\perp} \cdot \underline{v}_k^{\perp})}}{\cancel{(\underline{v}_k^{\perp} \cdot \underline{v}_k^{\perp})}} = 0 \Rightarrow \underline{v}_j^{\perp} \perp \underline{v}_k^{\perp}$$
  (for $k < j$) (orthogonal)
- Vectors $\underline{v}_j^{\perp}$ are not orthonormal i.e. $\|\underline{v}_j^{\perp}\| \neq 1$
  To obtain orthonormal vectors: $\hat{\underline{v}}_j^{\perp} = \dfrac{\underline{v}_j^{\perp}}{\|\underline{v}_j^{\perp}\|}$

Similarly for functions:

$$\boxed{h_j^{\perp}(x) = h_j(x) - \sum_{i=1}^{j-1}\left[\frac{\langle h_j(x), h_i^{\perp}(x)\rangle}{\langle h_i^{\perp}(x), h_i^{\perp}(x)\rangle} h_i^{\perp}(x)\right]}$$

orthonormal functions: $\hat{h}_j^{\perp}(x) = \dfrac{h_j^{\perp}(x)}{\sqrt{\langle h_j^{\perp}(x), h_j^{\perp}(x)\rangle}}$

Read Example 37
from Hjemlstad (2005).



$$g_1(\xi) = \xi$$
$$g_2(\xi) = 4\xi^2 - 3\xi$$
$$g_3(\xi) = 15\xi^3 - 20\xi^2 + 6\xi$$
$$g_4(\xi) = 56\xi^4 - 105\xi^3 + 60\xi^2 - 10\xi$$

1D Finite Element Basis

(S)

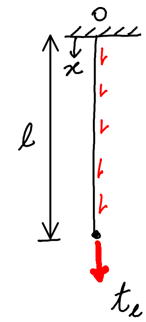
Find $u(x) \in H^2(0,\ell)$, t_0 such that

BC

$$(Cu')' + b = 0 \qquad \text{on} \quad x \in (0,\ell)$$

$$u(0) = u_0 \qquad \text{at} \quad x = 0$$

$$\sigma(\ell) = (Cu')(\ell) = t_\ell \qquad \text{at} \quad x = \ell$$



$b = \rho g$ (1, 10)
$C = 1$
$u_0 = 0$
$t_\ell = 10$

(W) PVW

$$G(u,\bar{u}) = \int_0^\ell \bar{u}'(Cu')\,dx - \int_0^\ell \bar{u}\, b\, dx - \bar{u}(\ell)\, t_\ell - \bar{u}(0)\, t_0$$

Find $u(x) \in \{ H^1(0,\ell)$ and $\overbrace{u(0) = u_0}^{\text{EBC}} \}$ and t_0

$$G(u,\bar{u}) = 0 \qquad \forall \underbrace{\bar{u} \in H^1_E(0,\ell)}_{\text{HEBC}}$$

Approximation (Alternative implementation of HEBC)

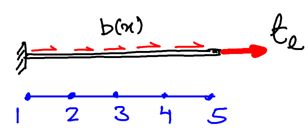
$$u(x) = \sum_{i=1}^{N} a_i h_i(x) = [a_1\ a_2 \,--\, a_N] \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_N(x) \end{bmatrix}$$

Real displacement $\boxed{u(x) \simeq \underline{a}^T \underline{h}(x)}$

Virtual displacement $\boxed{\bar{u}(x) \simeq \bar{\underline{a}}^T \underline{h}(x)}$ $\Rightarrow \tilde{G}^h(\underline{a}, \bar{\underline{a}}) = \bar{\underline{a}}^T\left(\underset{\sim}{K}\, \underline{a} - \underline{f}\right)$

(Galerkin)

1D Finite Element Basis Functions



$$l_e = \frac{L}{N} = x_i - x_{i-1}$$

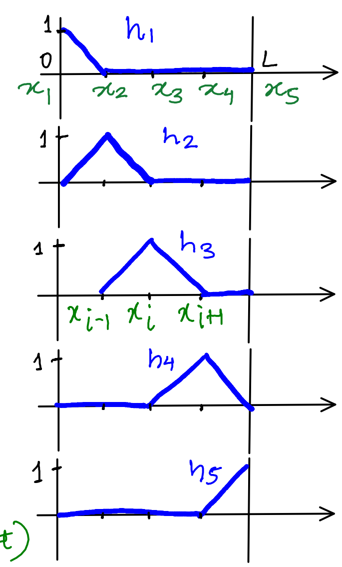
($N = 4$ here)

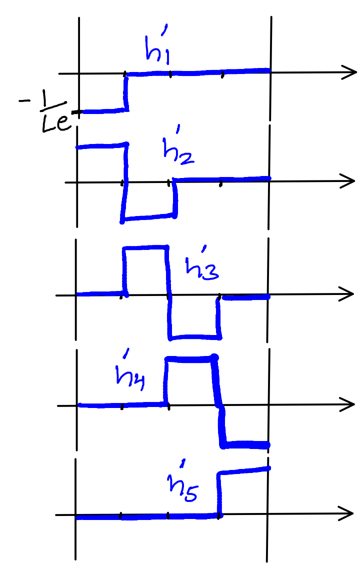
Note : $H^1(0,\ell)$

$$h_i(x) = \begin{cases} \dfrac{x - x_{i-1}}{x_i - x_{i-1}} & : x \in [x_{i-1}, x_i] \\[2ex] \dfrac{x_{i+1} - x}{x_{i+1} - x_i} & : x \in [x_i, x_{i+1}] \end{cases}$$

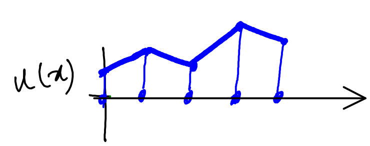
$$h_i'(x) = \begin{cases} 1/l_e & : x \in [x_{i-1}, x_i] \\ -1/l_e & : x \in [x_i, x_{i+1}] \end{cases}$$

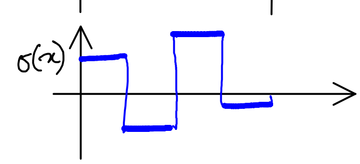
(local support)





N pts : N unknowns





From the weak form (W):

$$G(u,\bar{u}) = \int_0^l \bar{u}'(Cu')\,dx - \int_0^l \bar{u}\,b\,dx - \bar{u}(l)\,t_l - \bar{u}(0)\,t_0$$

Substitute the approximations $u(x) = \underline{a}^T \underline{h}$

$\bar{u}(x) = \underline{\bar{a}}^T \underline{h}$

$$\Rightarrow \tilde{G}^h(\underline{a},\underline{\bar{a}}) = \underline{\bar{a}}^T(\underset{\sim}{K}\,\underline{a} - \underline{f}) = 0 \quad \forall\ \underline{\bar{a}}$$

where

$$\underset{\sim}{K} = \int_0^l C\,\underline{h}'\,\underline{h}'^T\,dx \qquad \underline{f} = \int_0^l \underline{h}^T b + \underline{h}^T(l)\,t_l + \underline{h}^T(0)\,t_0$$

$$\underset{\sim}{K} = \int_0^l C \begin{bmatrix} h_1' \\ h_2' \\ \vdots \\ h_N' \end{bmatrix} \begin{bmatrix} h_1' & h_2' & \cdots & h_N' \end{bmatrix} dx = \begin{bmatrix} K_{11} & K_{12} & 0 & \cdots & 0 \\ & K_{22} & K_{23} & \cdots & 0 \\ & & \ddots & \ddots & \\ & \text{Symmetric} & & K_{N-1,N-1} & K_{N-1,N} \\ & & & & K_{NN} \end{bmatrix}$$

<u>Note</u>:
- Diagonal terms

$$\underset{\text{(no summation)}}{K_{ii}} = \int_0^l C\,h_i'\,h_i'\,dx = \int_{x_{i-1}}^{x_{i+1}} C\,h_i'\,h_i'\,dx \quad \text{(local support)}$$

$$= \int_{x_{i-1}}^{x_i} C\left(\frac{+1}{l_e}\right)\left(\frac{+1}{l_e}\right)dx + \int_{x_i}^{x_{i+1}} C\left(\frac{-1}{l_e}\right)\left(\frac{-1}{l_e}\right)dx = \boxed{\frac{2C}{l_e}} \quad 2 \le i \le N-1$$

For $i=1$ and $i=N$: $K_{ii} = \frac{C}{l_e}$

- Off diagonal terms:

$$K_{ij} = \int_0^l C\,h_i'\,h_j'\,dx = \begin{cases} 0 & \text{if } |j-i|>1 \\ \displaystyle\int_{x_i}^{x_{i+1}} C\left(\frac{1}{l_e}\right)\left(\frac{-1}{l_e}\right)dx & \text{if } |j-i|=1 \end{cases} = \begin{cases} 0 & \text{if } |j-i|>1 \\ \boxed{\frac{-C}{l_e}} & \text{if } |j-i|=1 \end{cases}$$

$$\boxed{\Rightarrow [\underset{\sim}{K}] = \frac{C}{l_e}\begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}}$$

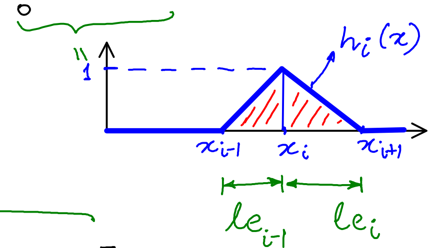
Similarly for the force vector

$$\underline{f} = \int_0^{\ell} \underline{h}\, b\, dx + \underline{h}(\ell)\, t_\ell + \underline{h}(0)\, t_0$$

$$\begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_N \end{bmatrix} = \int_0^{\ell} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \end{bmatrix} b\, dx + \begin{bmatrix} \cancel{h_1(\ell)} \rightarrow 0 \\ \cancel{h_2(\ell)} \rightarrow 0 \\ \cancel{h_3(\ell)} \rightarrow 0 \\ \cancel{h_4(\ell)} \rightarrow 0 \\ h_5(\ell) = 1 \end{bmatrix} t_\ell + \begin{bmatrix} h_1(0) = 1 \\ \cancel{h_2(0)} \rightarrow 0 \\ \cancel{h_3(0)} \rightarrow 0 \\ \cancel{h_4(0)} \rightarrow 0 \\ \cancel{h_5(0)} \rightarrow 0 \end{bmatrix} t_0$$

Note:

$$f_i = \int_0^{\ell} h_i\, b\, dx + h_i(\ell)\, t_\ell + h_i(0)\, t_0 = b \underbrace{\int_0^{\ell} h_i\, dx} + h_i(\ell)\, t_\ell$$



$$\Rightarrow f_i = b \left[\overbrace{\int_{x_{i-1}}^{x_i} \left(\frac{x - x_{i-1}}{x_i - x_{i-1}} \right) dx}^{= \frac{1}{2} le_{i-1}} + \overbrace{\int_{x_i}^{x_{i+1}} \left(\frac{x - x_i}{x_{i+1} - x_i} \right) dx}^{\frac{1}{2} le_i} \right] + h_i(\ell)\, t_\ell$$

$$\Rightarrow \underline{f} = b \begin{bmatrix} le/2 \\ le \\ le \\ le \\ le/2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} t_\ell + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} t_0 = \begin{bmatrix} b\,le/2 + t_0 \\ b\,le \\ b\,le \\ b\,le \\ b\,le/2 + t_L \end{bmatrix}$$

$$\Rightarrow \tilde{G}^h(\underline{a}, \underline{\bar{a}}) = \underline{\bar{a}}^T (\underset{\sim}{K}\, \underline{a} - \underline{f}) = 0 \quad \forall\ \bar{a}$$

specified

Unknown

$$\Rightarrow [\cancel{\bar{a}_1}\ \bar{a}_2\ \bar{a}_3\ \bar{a}_4\ \bar{a}_5] \left\{ \frac{C}{le} \left[\begin{array}{c|cccc} 1 & -1 & 0 & 0 & 0 \\ \hline -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & -1 & 1 \end{array}\right] \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} - \left[\begin{array}{c} b\,le/2 + t_0 \\ \hline b\,le \\ b\,le \\ b\,le \\ b\,le/2 + t_L \end{array}\right] \right\} = 0$$

$\bar{a}_1 \rightarrow 0$ (HEBC)

$$\Rightarrow \frac{C}{le} \begin{bmatrix} 2 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 \\ 0 & -1 & 2 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} a_2 \\ a_3 \\ a_4 \\ a_5 \end{bmatrix} = \begin{bmatrix} b\,le \\ b\,le \\ b\,le \\ b\,le/2 + t_L \end{bmatrix} - \frac{C}{le} \begin{bmatrix} -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} a_1$$

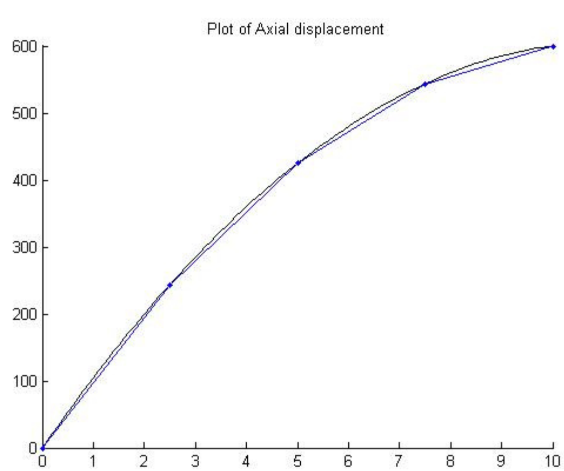
Solution of the linear problem:

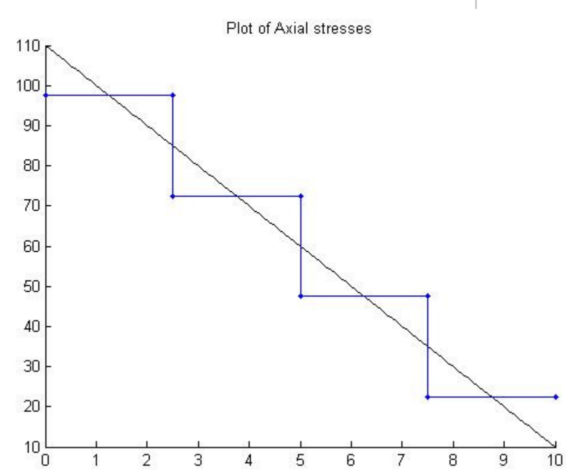
$$\underline{a} = \underset{\sim}{K}^{-1} \underline{f}$$

(in MATLAB: refer to code posted on Blackboard)

MATLAB Code


```
% 1D Problem - 4 element FE basis

clear all; clc; close all; % clear all the existing variables (new start)

L = 10;
N=100; dL = L/N;
E = 1; A=1; dens = 1;
C = E*A;
rho = dens*A;
g = 10;
u0 = 0;
tl = 10;

x = 0:dL:L;
u_exact = -0.5*(rho*g)/C*x.^2 + (tl+rho*g*L)/C*x;
s_exact = -rho*g*x + (tl+rho*g*L);

figure(1); hold on;
plot(x,u_exact,'k-'); title('Plot of Axial displacement')
figure(2); hold on;
plot(x,s_exact,'k-'); title('Plot of Axial stresses')

Nel = 4; % dont change yet
Le = L/Nel;
xvec = 0:Le:L;

% Stiffness Matrix from a 5-shape function FE approximation
% a0 = 0 (HEBC)
K = C/Le*[  1  -1    0   0   0 ; ...
           -1   2  -1    0   0 ; ...
            0  -1   2   -1   0 ; ...
            0   0  -1    2  -1 ; ...
            0   0   0   -1   1 ]

% Load vector
f = [rho*g*Le/2; rho*g*Le ; rho*g*Le ; rho*g*Le ; rho*g*Le/2+tl  ]

% Solve
a = zeros(5,1);
dofspec = 1;
doffree = 2:5;

a(dofspec) = 0;
a(doffree) = K(doffree,doffree)\(f(doffree)-K(doffree,dofspec)*a(dofspec));
disp('Displacements')
a
disp('Nodal forces');
f = K*a
disp('Reaction t0');   % t0 = f(dofspec) - rho*g*Le/2
t0 = f(dofspec) - rho*g*Le/2
```

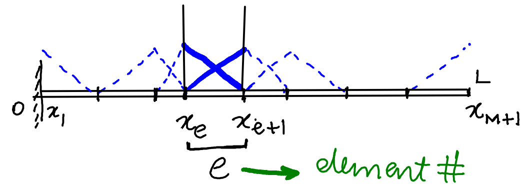




Element-wise computations for finite elements

Recall:

$$G(u,\bar{u}) = \int_0^{\ell} \bar{u}' C u' \, dx - \int_0^{\ell} \bar{u}\, b\, dx - \bar{u}(\ell)\, t_\ell - \bar{u}(0)\, t_0$$



$x_1 \rightarrow x_{M+1}$

$M+1$: Nodes

This integral may be written as a sum over "elements":

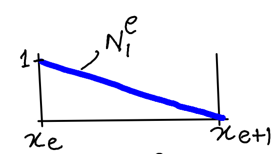
$$G(u,\bar{u}) = \sum_{e=1}^{M} \left[\int_{x_e}^{x_{e+1}} \bar{u}' C u' \, dx \right] - \sum_{e=1}^{M} \left[\int_{x_e}^{x_{e+1}} \bar{u}\, b\, dx \right] - \bar{u}(\ell)\, t_\ell$$

Approximating $u(x)$ and $\bar{u}(x)$ within element "e" as:

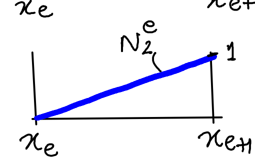
$$u(x) \approx u_e^h(x) = N_1^e(x)\, d_1^e + N_2^e(x)\, d_2^e$$

$$= \underbrace{\begin{bmatrix} N_1^e(x) & N_2^e(x) \end{bmatrix}}_{\text{shape function Matrix } \underset{\sim}{N}} \underbrace{\begin{bmatrix} d_1^e \\ d_2^e \end{bmatrix}}_{\text{Element dof}} \qquad \text{i.e. } \boxed{u_e^h(x) = \underset{\sim}{N}\, \underline{d}}$$

where $N_1^e(x) = h_e(x)\Big|_{(x_e < x < x_{e+1})} = \left(\dfrac{x_{e+1} - x}{x_{e+1} - x_e} \right)$



and $N_2^e(x) = h_{e+1}(x)\Big|_{(x_e < x < x_{e+1})} = \left(\dfrac{x - x_e}{x_{e+1} - x_e} \right)$



Similarly

$$\bar{u}(x) \approx \bar{u}_e^h(x) = \begin{bmatrix} N_1^e(x) & N_2^e(x) \end{bmatrix} \underbrace{\begin{bmatrix} \bar{d}_1^e \\ \bar{d}_2^e \end{bmatrix}}_{\text{virtual element dofs}} \qquad \boxed{\bar{u}_e^h(x) = \underset{\sim}{N}\, \underline{\bar{d}}}$$

Using this approximation:

$$u'(x) \approx \frac{d u_e^h}{dx} = \underbrace{\begin{bmatrix} \dfrac{dN_1^e}{dx} & \dfrac{dN_2^e}{dx} \end{bmatrix}}_{\underset{\sim}{B}} \begin{bmatrix} d_1^e \\ d_2^e \end{bmatrix} \qquad \text{ie. } \boxed{\epsilon_e^h(x) = \underset{\sim}{B}\, \underline{d}}$$

and

$$\bar{u}'(x) \approx \frac{d \bar{u}_e^h}{dx} = \begin{bmatrix} \dfrac{dN_1^e}{dx} & \dfrac{dN_2^e}{dx} \end{bmatrix} \begin{bmatrix} \bar{d}_1^e \\ \bar{d}_2^e \end{bmatrix} \qquad \text{ie } \boxed{\bar{\epsilon}_e^h(x) = \underset{\sim}{B}\, \underline{\bar{d}}}$$

Substituting the boxed equations into the weak form:

$$\text{(W)} \quad G(u,\bar{u}) = \sum_{e=1}^{M}\left[\int_{x_e}^{x_{e+1}} \bar{u}'\, C u'\, dx\right] - \sum_{e=1}^{M}\left[\int_{x_e}^{x_{e+1}} \bar{u}\, b\, dx\right] - \bar{u}(l)\, t_l - \bar{u}(0)\, t_0$$

$$\underset{\text{(W)}}{G(u,\bar{u})} \approx \underset{\text{(G)}}{\tilde{G}^h}\left(\{\underset{\sim}{d}\}_{e=1}^{M}, \{\bar{\underset{\sim}{d}}\}_{e=1}^{M}\right)$$

Note: $(\bar{u}_e')^T = \bar{\underset{\sim}{d}}_e^T \underset{\sim}{B}_e^T$

$$= \sum_{e=1}^{M} \bar{\underset{\sim}{d}}^{e^T} \underbrace{\left[\int_{x_e}^{x_{e+1}} [\underset{\sim}{B}^T C \underset{\sim}{B}]\, dx\right]}\, \underset{\sim}{d}^e - \sum_{e=1}^{M} \bar{\underset{\sim}{d}}^{e^T} \underbrace{\left[\int_{x_e}^{x_{e+1}} \underset{\sim}{N}^T b\, dx\right]} - \bar{u}(l)\, t_l$$

element stiffness matrix

$$\boxed{\underset{\sim}{K}^e = \int_{x_e}^{x_{e+1}} \begin{bmatrix} N_1^{e'} \\ N_2^{e'} \end{bmatrix} C \begin{bmatrix} N_1^{e'} & N_2^{e'} \end{bmatrix} dx}$$

$$\boxed{\underset{\sim}{f}^e = \int_{x_e}^{x_{e+1}} \begin{bmatrix} N_1^e \\ N_2^e \end{bmatrix} b\, dx}$$

element force vector

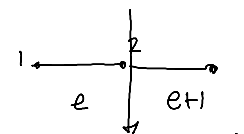
In the expanded form:

$$\tilde{G}^h\left(\{\underset{\sim}{d}\}_{e=1}^{M}, \{\bar{\underset{\sim}{d}}\}_{e=1}^{M}\right) = [\bar{d}_1^1 \mid \bar{d}_2^1]\left\{\begin{bmatrix} K_{11}^1 & K_{12}^1 \\ K_{21}^1 & K_{22}^1 \end{bmatrix}\begin{bmatrix} d_1^1 \\ d_2^1 \end{bmatrix} - \begin{bmatrix} f_1^1 \\ f_2^1 \end{bmatrix}\right\}$$

$$+ [\bar{d}_1^2 \mid \bar{d}_2^2]\left\{\begin{bmatrix} K_{11}^2 & K_{12}^2 \\ K_{21}^2 & K_{22}^2 \end{bmatrix}\begin{bmatrix} d_1^2 \\ d_{22}^2 \end{bmatrix} - \begin{bmatrix} f_1^2 \\ f_2^2 \end{bmatrix}\right\}$$

$$+ \quad \vdots$$

$$+ [\bar{d}_1^M \mid \bar{d}_2^M]\left\{\begin{bmatrix} K_{11}^M & K_{12}^M \\ K_{21}^M & K_{22}^M \end{bmatrix}\begin{bmatrix} d_1^M \\ d_{22}^M \end{bmatrix} - \begin{bmatrix} f_1^M \\ f_2^M \end{bmatrix}\right\} - \bar{u}(l)\, t_l - \bar{u}(0)\, t_0 - \sum_{J=1}^{N-1} \bar{u}(x_J)\, F_J$$



$$d_2^e = d_1^{e+1}$$

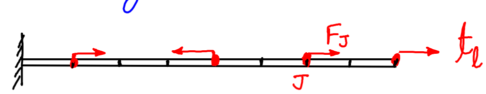
$$\tilde{G}^h\left(\{\underset{\sim}{d}\}_{e=1}^{M}, \{\bar{\underset{\sim}{d}}\}_{e=1}^{M}\right) = 0$$

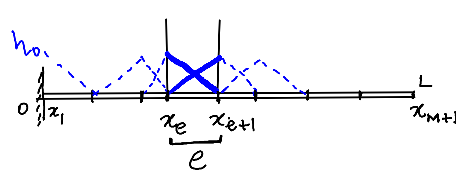
$\boxed{\text{Note}}$:

Global dof

$$d_1^G = d_1^1$$
$$d_2^G = d_2^1 = d_1^2$$
$$d_3^G = d_2^2 = d_1^3$$
$$\vdots \qquad \vdots \qquad \vdots$$
$$d_M^G = d_2^{M-1} = d_1^M$$
$$d_{M+1}^G = d_2^M$$

In addition to t_l, we may have Nodal loads F_J at any node x_J





This means that "Global" equation can be $\boxed{\text{ASSEMBLED}}$ by taking these terms common :-

Discretized weak form

$$\tilde{G}^h\left(\{\underline{d}\}_{e=1}^{M},\ \{\bar{\underline{d}}\}_{e=1}^{M}\right)$$

$$\{\bar d_1^G\ \ \bar d_2^G \cdots \bar d_M^G\ \ \bar d_{M+1}^G\}
\begin{bmatrix}
K_{11}^1 & K_{12}^1 & & & & 0\\
K_{21}^1 & (K_{22}^1+K_{11}^2) & K_{12}^2 & & & \\
 & K_{21}^2 & (K_{22}^2+K_{11}^3) & \ddots & & \\
 & & \ddots & \ddots & & \\
 & & & & (K_{22}^{M-1}+K_{11}^M) & K_{12}^M\\
0 & & & & K_{21}^M & K_{22}^M
\end{bmatrix}
\begin{Bmatrix} d_1^G\\ d_2^G\\ d_3^G\\ \vdots\\ d_{M-1}^G\\ d_M^G\end{Bmatrix}$$

$\underset{\sim}{K}^1$, $\underset{\sim}{K}^2$, $\underset{\sim}{K}^3$, $\underset{\sim}{K}^{M-1}$, $\underset{\sim}{K}^M$

$\bar d_2^1 = \bar d_1^2$

$= d_2^1 = d_1^2$

$$-\{\bar d_1^G\ \ \bar d_2^G \cdots \bar d_M^G\ \ \bar d_{M+1}^G\}
\begin{Bmatrix}
f_1^1 + F_1\\
f_2^1 + f_1^2 + F_2\\
f_2^2 + f_1^3 + F_3\\
\vdots\\
f_2^{M-1} + f_1^M + F_{N-1}\\
f_2^M + F_N
\end{Bmatrix}$$

global nodal loads

Finally

global residual

$$\boxed{\tilde{G}^h(\underline{d}^G, \bar{\underline{d}}^G) = \bar{\underline{d}}^{G^T}[\underline{g}^G] = \bar{\underline{d}}^{G}[\underset{\sim}{K}^G\underline{d}^G - \underline{f}^G] = 0}\quad \forall\ \bar{\underline{d}}^G$$

Boundary Conditions:

Note that our "N" functions do NOT satisfy essential boundary conditions.

To enforce BCs, we divide our "dofs" into "free" & "specified"

Say

$$\underline{d}^{G^T} = \{d_1^G\ \ d_2^G\ \ d_3^G\ \cdots\cdots\ d_M^G\ \ d_{M+1}^G\}$$

Free

Spec

Thus rearrange so that:

$$\left[\begin{array}{c|c} \underset{\sim}{K}_{ff}^G & \underset{\sim}{K}_{fs}^G\\ \hline \underset{\sim}{K}_{sf}^G & \underset{\sim}{K}_{ss}^G\end{array}\right]\begin{Bmatrix}\underline{d}_f^G\\ \underline{d}_s^G\end{Bmatrix} = \begin{Bmatrix}\underline{f}_f^G\\ \underline{f}_s^G\end{Bmatrix}$$

unknown

given

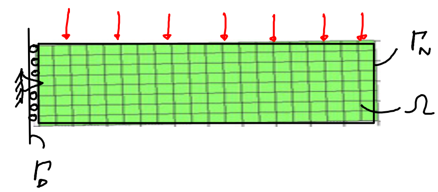
$\Rightarrow$ Solve $\boxed{[\underset{\sim}{K}_{ff}^G]\{\underline{d}_f^G\} = \{\underline{f}_f^G\} - [\underset{\sim}{K}_{fs}^G]\{\underline{d}_s^G\}}$ for $\underline{d}_f^G$

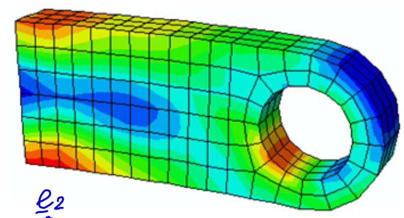
and

$$\underline{f}_s^G = [\underset{\sim}{K}_{sf}^G]\{\underline{d}_f^G\} + [\underset{\sim}{K}_{ss}^G]\{\underline{d}_s^G\}\quad \text{(support reactions)}$$

Postprocessing: Using $\underline{d}^G$ calculate stresses in each element.

Ritz and finite element methods for 2D and 3D problems





Recall:

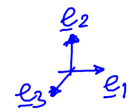
$$G(\underset{\sim}{u},\bar{\underset{\sim}{u}}) = \underbrace{\int_\Omega \bar{\underset{\sim}{\epsilon}} : \underset{\sim}{S}\, d\Omega}_{W_I} \underbrace{- \int_\Omega \bar{\underset{\sim}{u}}\cdot \underset{\sim}{b}\, d\Omega - \int_{\Gamma_N} \bar{\underset{\sim}{u}}\cdot \underset{\sim}{t}_{(n)}\, d\Gamma}_{-W_E}$$

W

(PVW)

Or using the Voight notation:

$$G(\underset{\sim}{u},\bar{\underset{\sim}{u}}) = \int_\Omega \underbrace{\bar{\underline{\epsilon}}^T \underline{S}}\, d\Omega - \int_\Omega \bar{\underline{u}}^T \underline{b}\, d\Omega - \int_{\Gamma_N} \bar{\underline{u}}^T \underline{t}_{(n)}\, d\Gamma$$



Note $\underline{\epsilon} = [\boxed{\epsilon_{11}\ \ \epsilon_{22}}\ \ \epsilon_{33}\ \ \boxed{2\epsilon_{12}}\ \ 2\epsilon_{23}\ \ 2\epsilon_{31}]^T$ 2D/3D

$\underline{S} = [\boxed{S_{11}\ \ S_{22}}\ \ S_{33}\ \ \boxed{S_{12}}\ \ S_{23}\ \ S_{31}]^T$ 2D/3D

Ritz method Approximation:

$$\underset{\sim}{u}(\underset{\sim}{x}) \cong \underbrace{\underset{\sim}{h}_o(\underset{\sim}{x})}_{EBC} + \sum_{i=1}^{N} a_i \underbrace{\underset{\sim}{h}_i(\underset{\sim}{x})}_{HEBC}$$

$$\begin{bmatrix} u_1(\underset{\sim}{x}) \\ u_2(\underset{\sim}{x}) \\ u_3(\underset{\sim}{x}) \end{bmatrix} \cong \begin{bmatrix} h_{01}(x,y,z) \\ h_{02}(x,y,z) \\ h_{03}(x,y,z) \end{bmatrix} + \sum_{i=1}^{N} \begin{bmatrix} a_{3i-2} \\ a_{3i-1} \\ a_{3i} \end{bmatrix} [h_i(x,y,z)] \qquad \begin{bmatrix} a_{2i-1} \\ a_{2i} \end{bmatrix}$$

$$\cong \begin{bmatrix} h_{01}(x,y,z) \\ h_{02}(x,y,z) \\ h_{03}(x,y,z) \end{bmatrix} + \underbrace{\begin{bmatrix} a_1 & a_4 & --- & a_{3N-2} \\ a_2 & a_5 & --- & a_{3N-1} \\ a_3 & a_6 & --- & a_{3N} \end{bmatrix}}_{3\times N} \underbrace{\begin{bmatrix} h_1(\underset{\sim}{x}) \\ h_2(\underset{\sim}{x}) \\ \vdots \\ h_N(\underset{\sim}{x}) \end{bmatrix}}_{N\times 1}$$

$$\Rightarrow \begin{bmatrix} u_1(\underset{\sim}{x}) \\ u_2(\underset{\sim}{x}) \\ u_3(\underset{\sim}{x}) \end{bmatrix} \cong \begin{bmatrix} h_{01}(x,y,z) \\ h_{02}(x,y,z) \\ h_{03}(x,y,z) \end{bmatrix} + \underbrace{\underbrace{\left[\begin{array}{ccc|ccc|c|ccc} h_1 & 0 & 0 & h_2 & 0 & 0 & \ddots & h_N & 0 & 0 \\ 0 & h_1 & 0 & 0 & h_2 & 0 & & 0 & h_N & 0 \\ 0 & 0 & h_1 & 0 & 0 & h_2 & & 0 & 0 & h_N \end{array}\right]}_{3\times 3N}}_{\underset{\sim}{N}} \qquad \underbrace{\underbrace{\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \\ a_5 \\ a_6 \\ \vdots \\ a_{3N-2} \\ a_{3N-1} \\ a_{3N} \end{bmatrix}}_{3N\times 1}}_{\underset{\sim}{d}}$$

$$\Rightarrow \underset{\sim}{u}(\underset{\sim}{x}) \cong \underbrace{\underset{\sim}{h}_o(\underset{\sim}{x})}_{EBC} + \underbrace{\underset{\sim}{N}(\underset{\sim}{x})}_{HEBC}\, \underset{\sim}{d}$$ Unknowns to solve for

Similarly $\bar{\underset{\sim}{u}}(\underset{\sim}{x}) \cong \underbrace{\underset{\sim}{N}(\underset{\sim}{x})}_{HEBC}\, \bar{\underset{\sim}{d}}$ arbitrary constants

Note: It may not always be easy find such functions for complicated shapes and boundary conditions.

Approximations of strains and stresses

$$
\underset{\sim}{\epsilon} = \begin{bmatrix} \boxed{\epsilon_{11}} \\ \boxed{\epsilon_{22}} \\ \epsilon_{33} \\ \boxed{2\epsilon_{12}} \\ 2\epsilon_{23} \\ 2\epsilon_{31} \end{bmatrix} = \begin{bmatrix} \boxed{u_{1,1}} \\ \boxed{u_{2,2}} \\ u_{3,3} \\ \boxed{u_{1,2} + u_{2,1}} \\ u_{2,3} + u_{3,2} \\ u_{3,1} + u_{1,3} \end{bmatrix} \cong \underbrace{\left[\begin{array}{ccc|ccc|c|ccc} h_{1,1} & & & h_{2,1} & & & & h_{N,1} & & \\ & h_{1,2} & & & h_{2,2} & & & & h_{N,2} & \\ & & h_{1,3} & & & h_{2,3} & - - - - & & & h_{N,3} \\ \hline h_{1,2} & h_{1,1} & & h_{2,2} & h_{2,1} & & & h_{N,2} & h_{N,1} & \\ & h_{1,3} & h_{1,2} & & h_{2,3} & h_{2,2} & & & h_{N,3} & h_{N,2} \\ h_{1,3} & & h_{1,1} & h_{2,3} & & h_{2,1} & & h_{N,3} & & h_{N,1} \end{array}\right]}_{\underset{\sim}{B}} \underbrace{\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \hline a_4 \\ a_5 \\ a_6 \\ \hline \vdots \\ \hline a_{3N-2} \\ a_{3N-1} \\ a_{3N} \end{bmatrix}_{3N\times 1}}_{\underset{\sim}{d}}
$$

$$
\Rightarrow \quad \underset{\sim}{\epsilon}(\underset{\sim}{x}) \cong \underset{\sim}{B}(\underset{\sim}{x})\, \underset{\sim}{d}
$$

$$
\bar{\underset{\sim}{\epsilon}}(\underset{\sim}{x}) \cong \underset{\sim}{B}(\underset{\sim}{x})\, \bar{\underset{\sim}{d}}
$$

$$
\text{and} \quad \underset{\sim}{\sigma} \cong \underset{\sim}{D}\, \underset{\sim}{B}(\underset{\sim}{x})\, \underset{\sim}{d}
$$

Discretized weak form:

$$
\tilde{G}^h(\underset{\sim}{u}, \bar{\underset{\sim}{u}}) = \int_\Omega \bar{\underset{\sim}{\epsilon}}^T \underset{\sim}{\sigma}\, d\Omega - \int_\Omega \bar{\underset{\sim}{u}}^T \underset{\sim}{b}\, d\Omega - \int_{\Gamma_N} \bar{\underset{\sim}{u}}^T \underset{\sim}{t}_{(n)}\, d\Gamma
$$

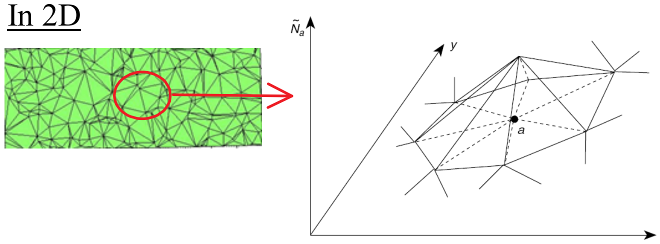
$$
= \bar{\underset{\sim}{d}}^T \underbrace{\left[\int_\Omega \left(\underset{\sim}{B}^T \underset{\sim}{D}\, \underset{\sim}{B}\right) d\Omega\right]}_{\underset{\sim}{K}} \underset{\sim}{d} - \bar{\underset{\sim}{d}}^T \underbrace{\left[\int_\Omega \left(\underset{\sim}{N}^T \underset{\sim}{b}\right) d\Omega + \int_{\Gamma_N} \underset{\sim}{N}^T \underset{\sim}{t}_{(n)}\, d\Gamma\right]}_{\underset{\sim}{f}}
$$

$$
\Rightarrow \quad \tilde{G}^h(\underset{\sim}{d}, \bar{\underset{\sim}{d}}) = \bar{\underset{\sim}{d}}^T \left[\underset{\sim}{K}\, \underset{\sim}{d} - \underset{\sim}{f}\right] = 0 \quad \forall\, \bar{\underset{\sim}{a}}
$$

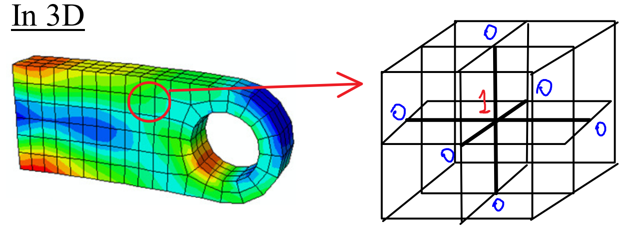
Finite Element approximations

Once again FE is a special case of the Ritz method.
We just use specific FE shape functions for $h_i(\underset{\sim}{x})$:

In 2D



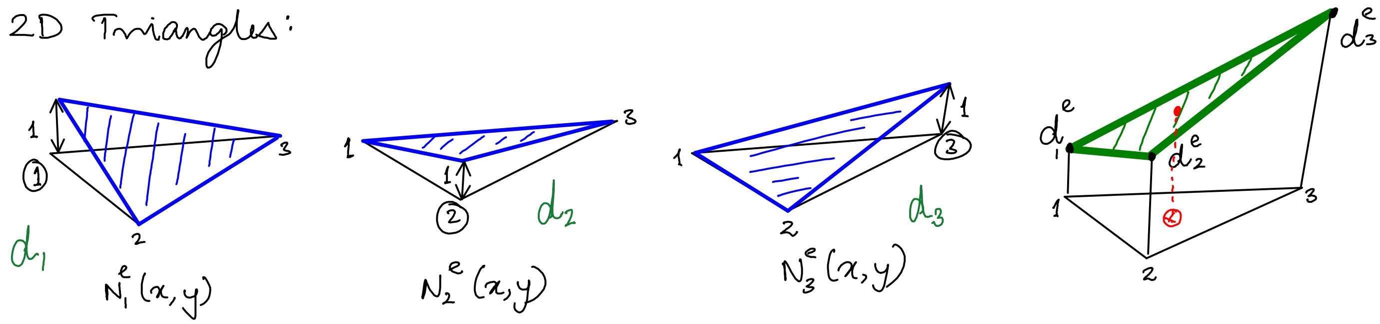
In 3D



Note:
The "hat/tent" shape functions of finite elements are $\in H^1(\Omega)$

Finite Element "**Element-wise**" approximations 2D

2D Triangles:



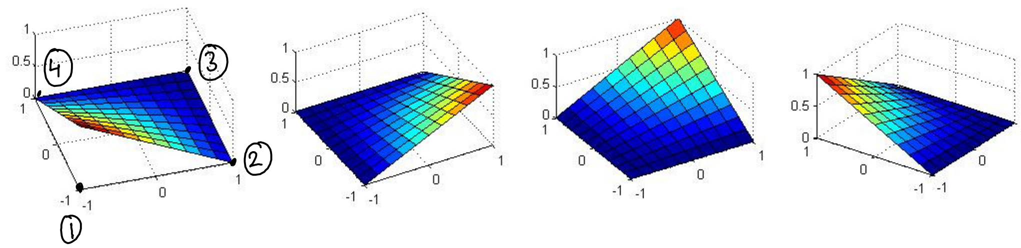
$$A_1 = x_2 y_3 - x_3 y_2 \qquad A_2 = x_3 y_1 - x_1 y_3 \qquad A_3 = x_1 y_2 - x_2 y_1$$

$$B_1 = y_2 - y_3 \qquad B_2 = y_3 - y_1 \qquad B_3 = y_1 - y_2$$

$$C_1 = x_3 - x_2 \qquad C_2 = x_1 - x_3 \qquad C_3 = x_2 - x_1$$

$$\boxed{N_\alpha(x,y) = \frac{1}{2\Delta}\left(A_\alpha + B_\alpha x + C_\alpha y\right)} \quad \alpha = 1,2,3 \qquad \text{where} \qquad 2\Delta = \det\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}$$

2D Quadrilaterals:

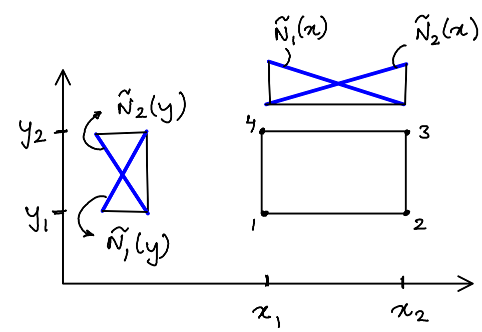


$$N_1^e(x,y) = \tilde{N}_1(x) * \tilde{N}_1(y)$$

$$N_2^e(x,y) = \tilde{N}_2(x) * \tilde{N}_1(y)$$

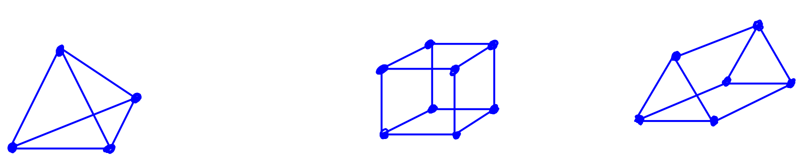
$$N_3^e(x,y) = \tilde{N}_2(x) * \tilde{N}_2(y)$$

$$N_4^e(x,y) = \tilde{N}_1(x) * \tilde{N}_2(y)$$



Finite element "**Element-wise**" approximation 3D

Element types:



Tetrahedron Hexahedron Prisms