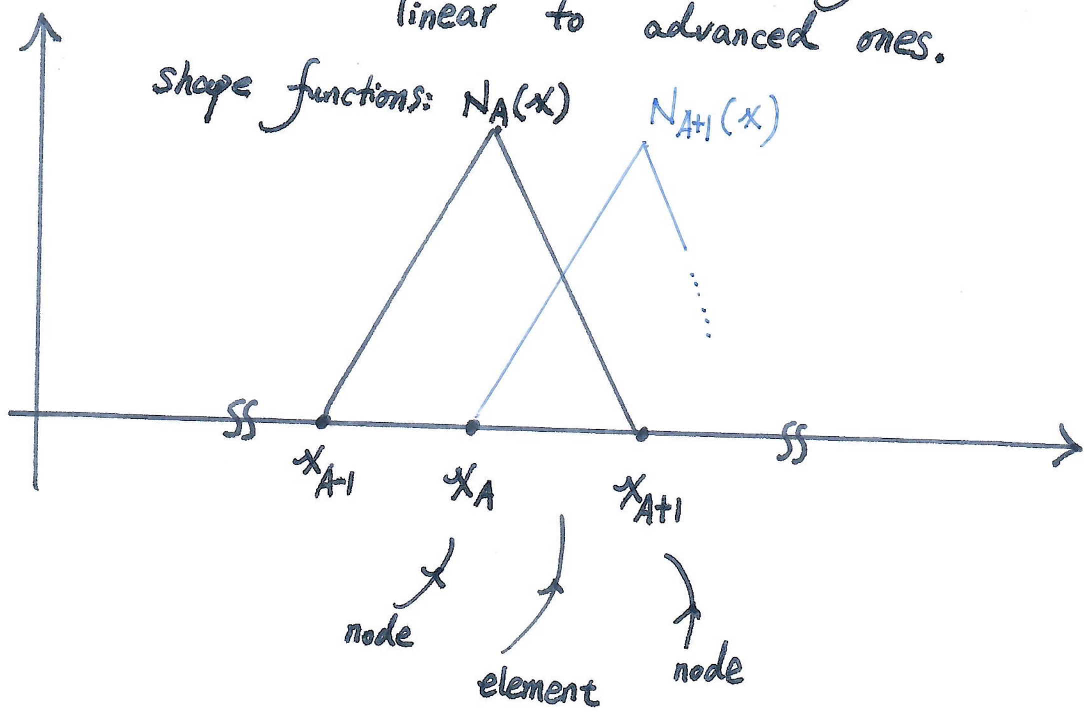
# The element point of view

Global point of view: $N_A$'s are defined as a function on $\Omega$.

useful in mathematical analysis.

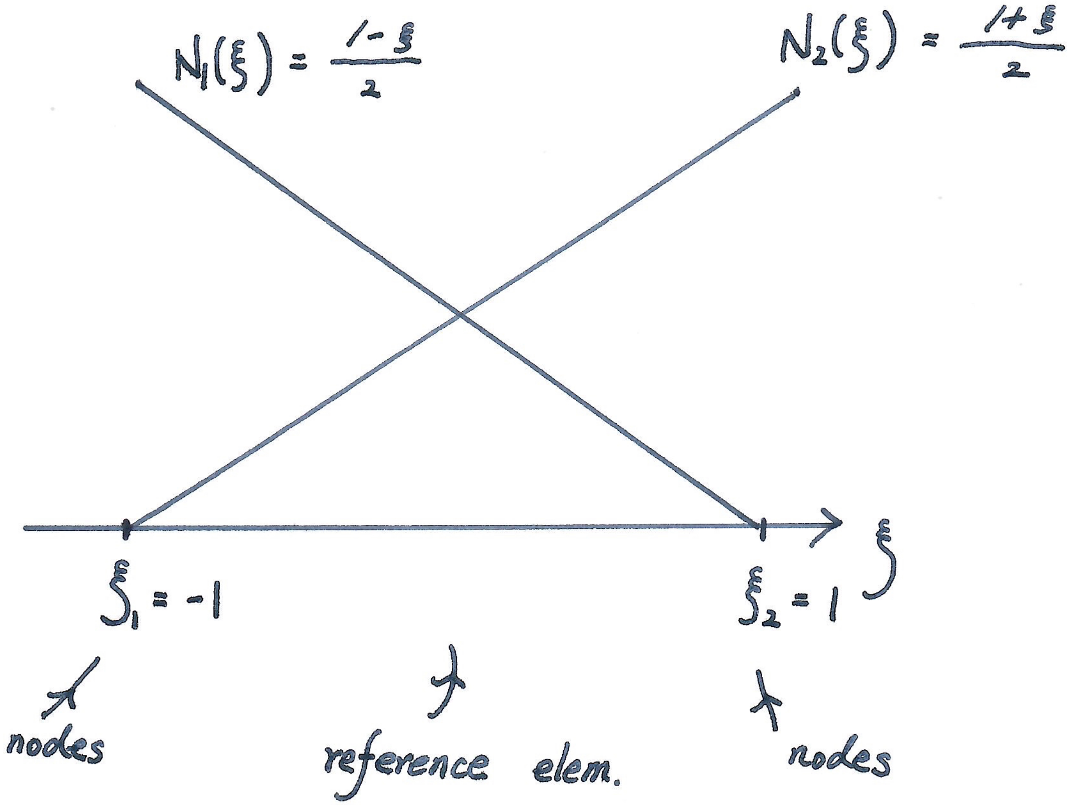
Local or element point of view: consider the problem based on the element.

- useful in programming
- useful for generalizing the element from P.W. linear to advanced ones.




Interpolation function: $u^h(x) = N_A(x)d_A + N_{A+1}(x)d_{A+1}$

We want to standardize the calculations over elements.




Interpolation function : $u^h(\xi) = N_1(\xi) d_1 + N_2(\xi) d_2$.

We need a mapping from the ref. elem. to the 'physical' elem :

$$\xi : [x_A, x_{A+1}] \longrightarrow [\xi_1, \xi_2]$$
$$x \longmapsto \xi$$

with $\xi(x_A) = \xi_1$, $\xi(x_{A+1}) = \xi_2$.

It is a standard partice that we choose $\xi_1 = -1$, $\xi_2 = 1$.

If we choose $\xi(x) = c_1 + c_2 x$.

$$\begin{cases} -1 = c_1 + c_2 x_A \\ 1 = c_1 + c_2 x_{A+1} \end{cases}$$

$$\Rightarrow \quad c_1 = -\frac{x_A + x_{A+1}}{h_A} \qquad\qquad h_A = x_{A+1} - x_A$$

$$c_2 = \frac{2}{h_A}$$

$$\Rightarrow \quad \xi(x) = \frac{2x - x_A - x_{A+1}}{h_A}$$

The inverse map is

$$x(\xi) = \frac{h_A \xi + x_A + x_{A+1}}{2} .$$

Remark: We may construct $x(\xi)$ as $x_A N_1(\xi) + x_{A+1} N_2(\xi)$.

If we use 'a' as the index for local objects, we have $N_a(\xi) = \frac{1}{2}(1 + \xi_a \xi)$ $a = 1, 2$.

If we use the superscript $e$ to identify the element $e$ that the local object belongs to, we have $d_a^e = d_A$,

$x^e : [\xi_1, \xi_2] \rightarrow [x_1^e, x_2^e] = [x_A, x_{A+1}]$.

$$x^e(\xi) = \sum_{a=1}^{2} N_a(\xi) x_a^e .$$

useful identities :

- $N_{a,\xi} = \frac{\xi_a}{2} = \frac{(-1)^a}{2}$
- $x^e_{,\xi} = \frac{x_2^e - x_1^e}{2} = \frac{h^e}{2}$
- $\xi_{,x} = (x_{,\xi})^{-1} = \frac{2}{h^e}$

Let $e$ be a variable index : $1 \leq e \leq n_{el}$ ← # of elements

We have $\Omega_1 = \bigcup_{e=1}^{n_{el}} \Omega_1^e$ $\quad \Omega_1^e = [x_1^e, x_2^e]$, and

$$\int_{\Omega_1} \cdots \, dx = \sum_{e=1}^{n_{el}} \int_{\Omega_1^e} \cdots \, dx .$$

$$K_{AB} = a(N_A, N_B) = \int_0^1 N_{A,x} N_{B,x} \, dx$$

$$= \sum_{e=1}^{n_{el}} \underbrace{\int_{x_1^e}^{x_2^e} N_{A,x} N_{B,x} \, dx}_{K_{AB}^e} = \sum_{e=1}^{n_{el}} K_{AB}^e .$$

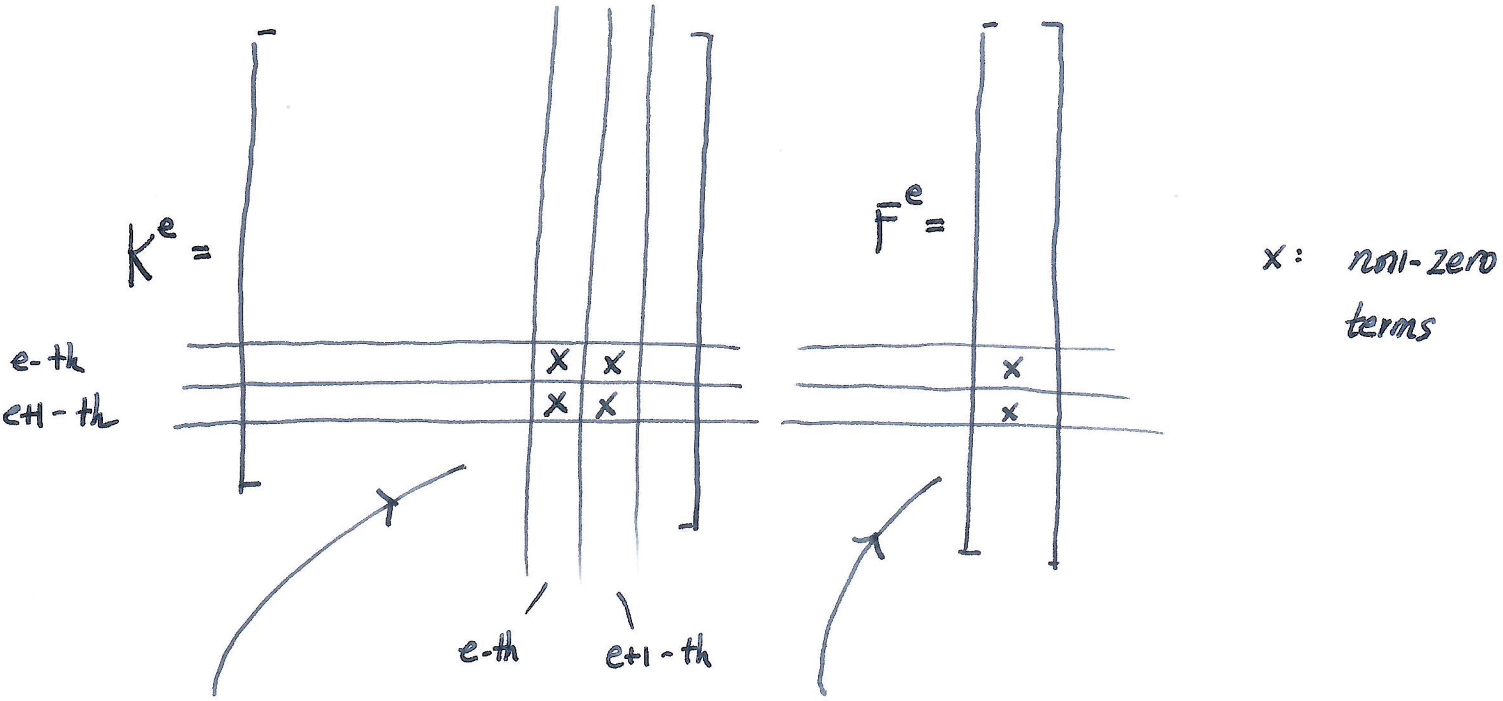
$$F_A = (N_A, f) + N_A(0) h - a(N_A, N_{n+1}) g$$

$$= \int_0^1 N_A f \, dx + \delta_{A1} h - \int_0^1 N_{A,x} N_{n+1,x} \, dx \, g$$

$$= \sum_{e=1}^{n_{el}} \left\{ \int_{x_1^e}^{x_2^e} N_A f \, dx + \delta_{e1} \delta_{A1} h - \int_{x_1^e}^{x_2^e} N_{A,x} N_{n+1,x} \, dx \, g \right\}$$

$$= \sum_{e=1}^{n_{el}} F_A^e$$

The above means $K$ & $F$ can be constructed by summing contributions from elements.




$$k^e = [k^e_{ab}], \quad 1 \leq a, b \leq 2; \qquad f^e = \{f^e_a\} \quad 1 \leq a \leq 2.$$

element stiffness matrix $\qquad$ element load vector

$$k^e_{ab} = \int_{\Omega^e} N_{a,x} N_{b,x}\, dx \qquad f^e_a = \int_{\Omega^e} N_a f\, dx + \begin{cases} \delta_{a1} h & e = 1 \\ 0 & e = 2, \cdots, n_{el}-1 \\ -k^e_{a2}\, g & e = n_{el}. \end{cases}$$

LM array : size is $n_{en} \times n_{el}$

$n_{en}$: # of element nodes (2 here)

$$A = LM(a, e) = \begin{cases} e & \text{if } a = 1 \\ e+1 & \text{if } a = 2 \end{cases}$$

| a \ e | 1 | 2 | 3 | ... | e | ... | $n_{el}-1$ | $n_{el}$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2 | 3 | | e | | n-1 | n |
| 2 | 2 | 3 | 4 | | e+1 | | n | 0 |

Notice!

Usage: Given $[k^e_{ab}]$, we put them into $K$ as

$$K_{ee} \mathrel{+}= k^e_{11} \qquad K_{e,e+1} \mathrel{+}= k^e_{12}$$

$$K_{e+1\,e} \mathrel{+}= k^e_{21} \qquad K_{e+1,e+1} \mathrel{+}= k^e_{22}$$

Given $[f^e_a]$, we assemble $F$ as

$$F_e \mathrel{+}= f^e_1 \qquad F_{e+1} \mathrel{+}= f^e_2$$

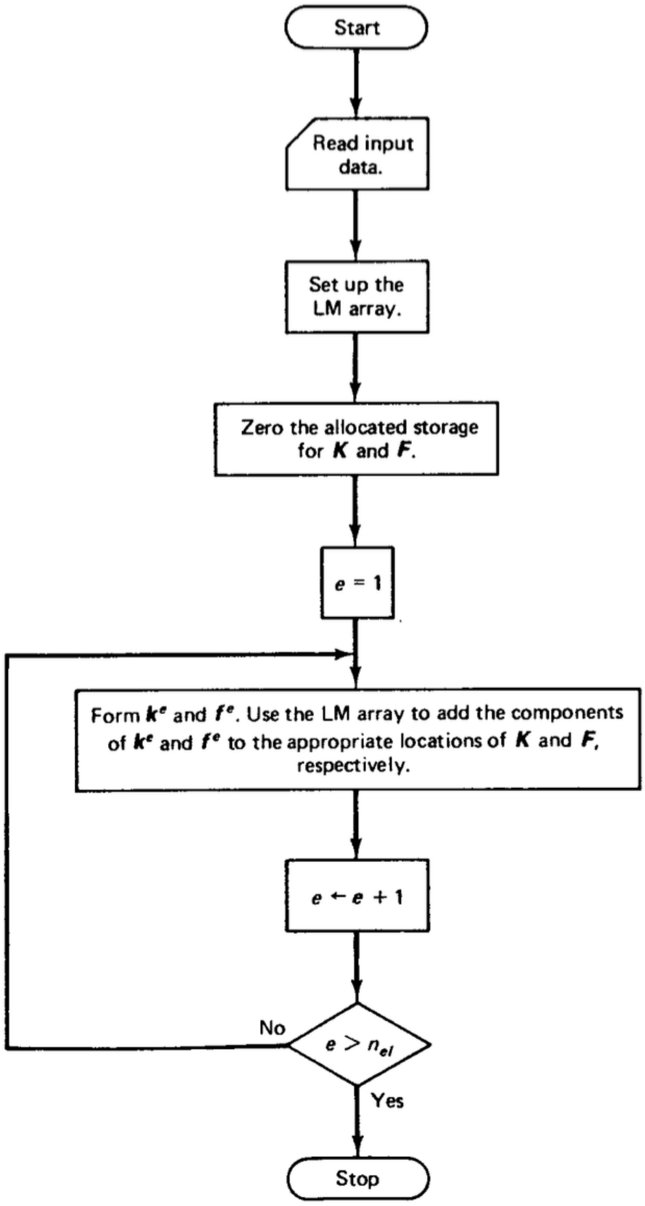
$LM(1,e)$ $\qquad$ $LM(2,e)$.

Remark: The 0-value in $LM$ array will be ignored. In some languages, negative index is ignored. Here, for $e = n_{el}$, we only perform

$$K_{nn} \mathrel{+}= f^{n_{el}}_{11} \qquad F_n \mathrel{+}= f^{n_{el}}_1 .$$

The action of the assembly algorithm is by $\mathbb{A}$, the assembly operator.

$$K = \mathop{\mathbb{A}}_{e=1}^{n_{el}} k^e \qquad F = \mathop{\mathbb{A}}_{e=1}^{n_{el}} f^e .$$


Start
Read input data.
Set up the LM array.
Zero the allocated storage for $\boldsymbol{K}$ and $\boldsymbol{F}$.
$e = 1$
Form $\boldsymbol{k}^e$ and $\boldsymbol{f}^e$. Use the LM array to add the components of $\boldsymbol{k}^e$ and $\boldsymbol{f}^e$ to the appropriate locations of $\boldsymbol{K}$ and $\boldsymbol{F}$, respectively.
$e \leftarrow e + 1$
$e > n_{el}$
No
Yes
Stop

## Change-of-variable formula

Let the mapping $x : [\xi_1, \xi_2] \to [x_1, x_2]$ be continuously differentiable, with $x(\xi_1) = x_1$, $x(\xi_2) = x_2$.

$$\int_{x_1}^{x_2} f(x)\, dx = \int_{\xi_1}^{\xi_2} f(x(\xi))\, x_{,\xi}(\xi)\, d\xi .$$

Now $k^e_{ab} = \int_{x^e_1}^{x^e_2} N_{a,x} N_{b,x}\, dx$

$$= \int_{-1}^{1} N_{a,x}(x(\xi))\, N_{b,x}(x(\xi))\, x_{,\xi}\, d\xi$$

$$= \int_{-1}^{1} N_{a,\xi}\, \xi_{,x}\, N_{b,\xi}\, \xi_{,x}\, x_{,\xi}\, d\xi$$

$$= \int_{-1}^{1} N_{a,\xi}\, N_{b,\xi}\, \xi_{,x}\, d\xi$$

$$= \int_{-1}^{1} \frac{(-1)^a}{2} \frac{(-1)^b}{2} \frac{2}{h^e}\, d\xi$$

$$= \frac{(-1)^{a+b}}{h^e}$$

comes from '$x_{,\xi}$', relies on the particular element data.

not dependent on the particular element data

$$\text{part of } f^e_a = \int_{x^e_1}^{x^e_2} N_a f \approx \int_{x^e_1}^{x^e_2} N_a f^h = \int_{x^e_1}^{x^e_2} N_a \sum_{b=1}^{2} f_b N_b$$

$$= \sum_{b=1}^{2} \int_{-1}^{1} N_a(\xi)\, N_b(\xi)\, x_{,\xi}\, d\xi \; f_b$$

$$= \frac{h^e}{2} \sum_{b=1}^{2} \int_{-1}^{1} N_a(\xi)\, N_b(\xi)\, d\xi \; f_b$$

$$= \frac{h^e}{6} \sum_{b=1}^{2} (1 + \delta_{ab})\, f_b$$

$$= \frac{h^e}{6} \begin{Bmatrix} 2f_1 + f_2 \\ f_1 + 2f_2 \end{Bmatrix} .$$

Alternatively, we may work on $f$ directly by invoking a quadrature rule:

$$f_a^e = \int_{x_1^e}^{x_2^e} N_a f = \int_{-1}^{1} N_a(\xi)\, f(x(\xi))\, x_{,\xi}\, d\xi$$

$$= \frac{h^e}{2} \int_{-1}^{1} N_a(\xi)\, f(x(\xi))\, d\xi$$

$$\approx \frac{h^e}{2} \sum_{l=1}^{n_{int}} W_l\, N_a(\xi_l)\, f(x(\xi_l)) \; * .$$

quadrature weights    quadrature points