Thus far we have viewed our shape functions defined over the entire domain.
In this we have

$$u^h = v^h + g^h = N_A d_A + N_B g_B$$

We ultimately need to have this decomposition. However, we want to get there from individual elements where we consider all possible d.o.f and figure out later which will go with d_A's and which are "taken up" by g's (essential B.C.)

For an element, e, we will have

$$u^{h^e} = \sum_{b=1}^{nee} N_b d_b$$

nee - all potential dof for the element

Lower case for elements

When we want to put things together the d_b's will be matched to the right d_B's and g_B's at the global level and we can work it out - Lets see that algebraically gives us the same thing that we were getting before

define - $\{D_B\}$ ← The list of all possible global dof (includes d_A's and g_B's) that all the d_a's match to

Assume that $\{D_B\}$ is partitioned into $\begin{Bmatrix} \{d_B\} \\ \{g_C\} \end{Bmatrix}$

Then all the element stiffness will go into the right place in the global system

$$\begin{bmatrix}[K_{dd}] & [K_{dg}]\\ [K_{gd}] & [K_{gg}]\end{bmatrix}\begin{Bmatrix}\{d_B\}\\ \{g_C\}\end{Bmatrix} = \begin{Bmatrix}\{F_B\}\\ \{R_C\}\end{Bmatrix}$$

body loads? + tractions

"reactions"

Top Partition

$$[K_{dd}]\{d_B\} + [K_{dg}]\{g_C\} = \{F_B\}$$

$$[K_{dd}]\{d_B\} = \{F_B\} - [K_{dg}]\{g_C\}$$

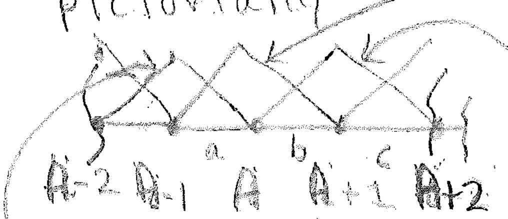
Which is infact exactly what we had from before $= a(w^h, v^h) = (w^h, f) + (w^h, h)_{\Gamma} - a(w^h, g)$

Section 1.8 begins to lay the ground work for making the relationship from element dof to global dof.

pictorially

node A shape function

node $A+1$ shape function

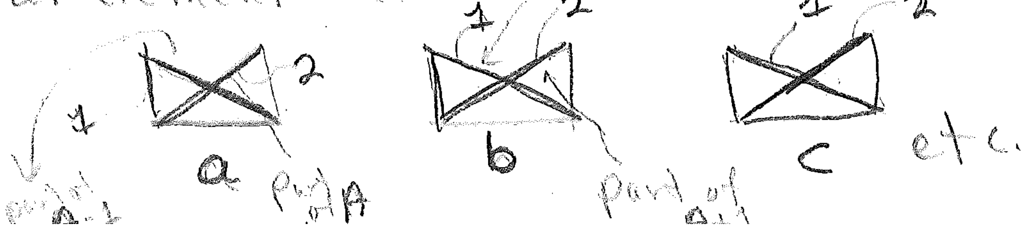


$N_i(x_j) = \delta_{ij}$

node $A-1$ shape function

other part of A

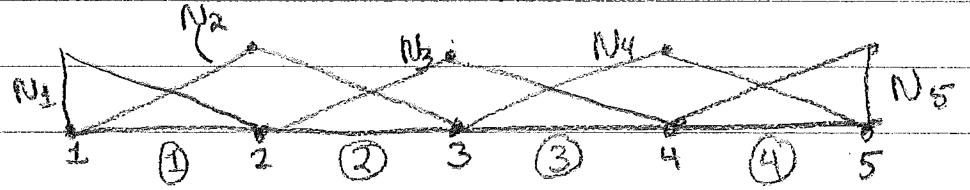
at element level



etc.

Global dof - sum element contributions

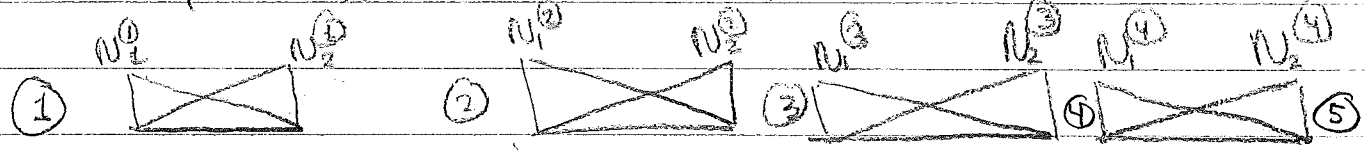
Lets Look at this for our specific case with 5 nodes



Global

$n=4$, with essential bc on right need one shape function for that $\rightarrow N_5$ $\leftarrow$ $\underline{N_{n+1}}$

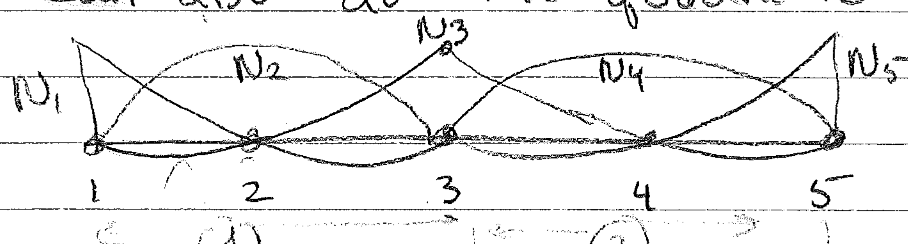
Locally - we have 4 elements



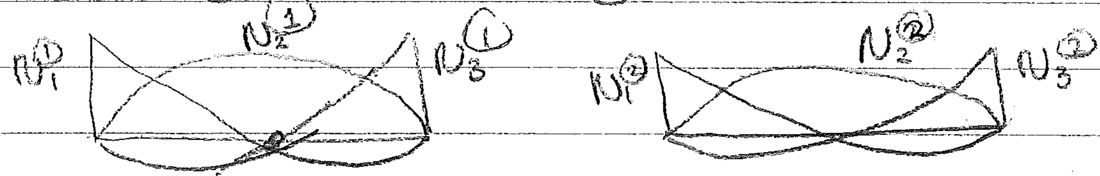
$N_1 = N_1^{(1)}$, $N_2 = N_2^{(1)} + N_1^{(2)}$, $N_3 = N_2^{(2)} + N_1^{(3)}$

$N_4 = N_2^{(3)} + N_1^{(4)}$, $N_5 = N_2^{(4)}$

Can also do the quadratic $\Leftarrow$ Now 2 - 3 noded elements



$N_e = 2$

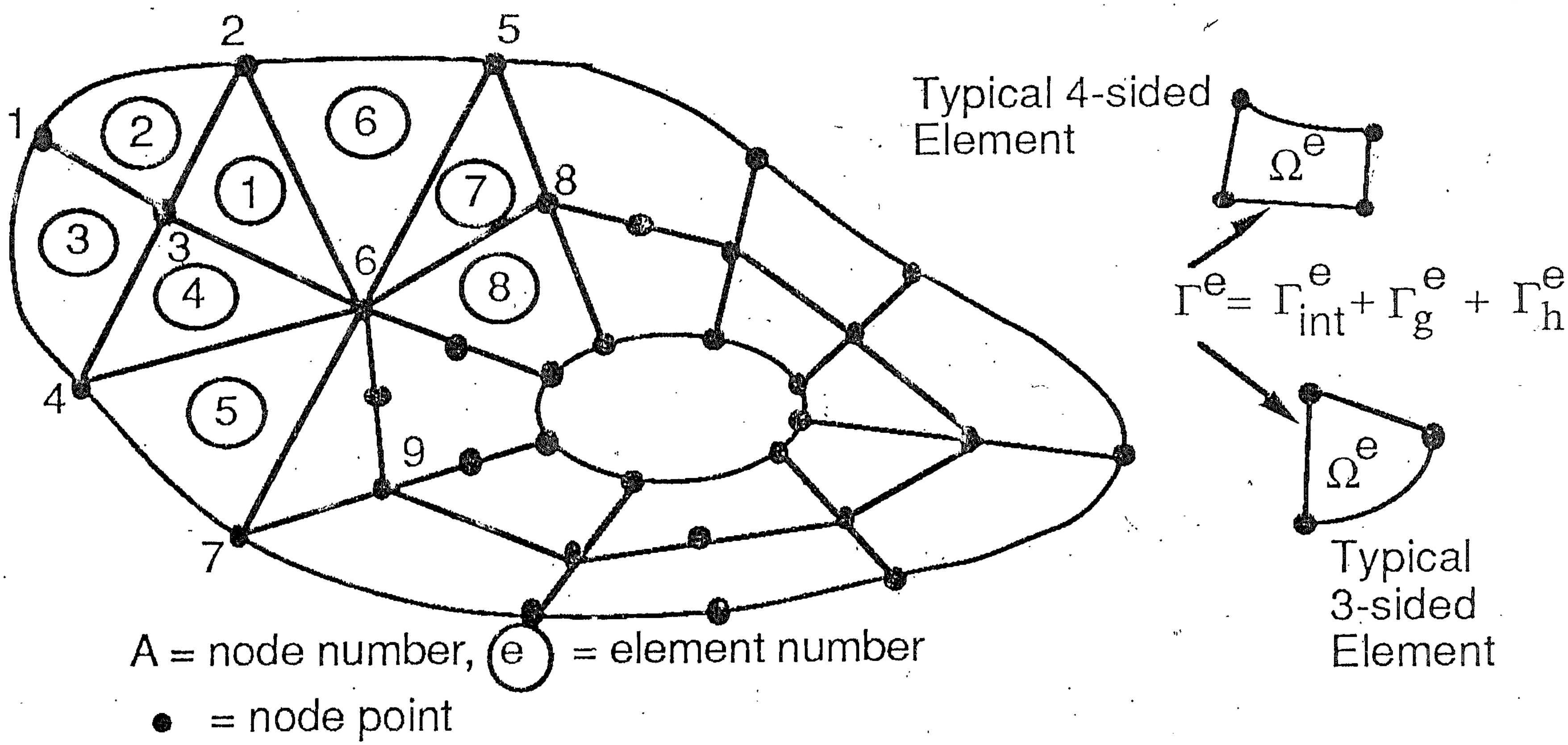


$N_1 = N_1^{(1)}$, $N_2 = N_2^{(1)}$, $N_3 = N_3^{(1)} + N_1^{(2)}$,

$N_4 = N_2^{(2)}$, $N_5 = N_3^{(2)}$

Note - each shape function meets $N_A(x_B) = \delta_{AB}$

shape functions are continuous in value, but not slope between elements

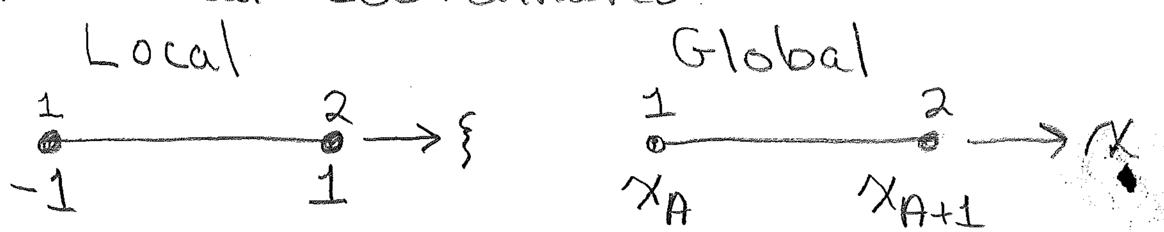


1
2
3
4
5
6
7
8
9
Typical 4-sided Element
Ω^e
$\Gamma^e = \Gamma^e_{int} + \Gamma^e_g + \Gamma^e_h$
Typical 3-sided Element
A = node number, e = element number
• = node point

Local Coordinates

Our example used $\Omega: 0<x<1$
In general our elements are of any size. However, when we get to 2D and 3D it will not be possible to write shape functions for any shape element. Instead we will write our shape functions over a regularly shaped element and construct a mapping to account for the actual element shape and size.

1D local coordinates



Note-conditions $\quad \xi(x_A) = \xi_1 = -1$
$\xi(x_{A+1}) = \xi_2 = 1$

Use 2 conditions to define a linear map

$$\xi(x) = C_1 + C_2 x$$

define $h_A = x_{A+1} - x_A$

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

$$-1 = C_1 + C_2 x_A$$

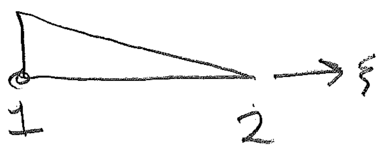
$$1 = C_1 + C_2 x_{A+1}$$

$$C_2 = 2/(x_{A+1} - x_A)$$

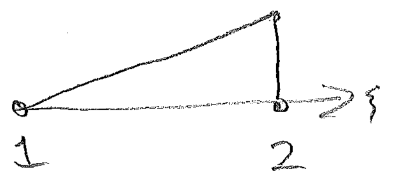
$$C_1 = \frac{-x_{A+1} - x_A}{(x_{A+1} - x_A)}$$

F

We really do not find it convenient to construct our mappings this way.
Instead we find it better to define our shape functions in the local coordinate system and use them to define our maps.



$$N_1 = \frac{1}{2}(1-\xi)$$



$$N_2 = \frac{1}{2}(1+\xi)$$

These allow us to directly define a linear map.

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

easy to take derivative wrt to ξ

$$x_{,\xi} = \sum_{a=1}^{2} N_{a,\xi} \, x_a^e = -\frac{1}{2}x_1 + \frac{1}{2}x_2$$

Problem is we need $N_{a,x}$
start with chain rule

$$\frac{\partial}{\partial \xi}\left(f(x(\xi))\right) = \frac{\partial f}{\partial x}(x(\xi)) \, \frac{\partial x}{\partial \xi}(\xi) = f_{,x}(x(\xi))\, x_{,\xi}(\xi)$$

$$f_{,x}(x(\xi)) = \left(x_{,\xi}\right)^{-1} \frac{\partial f}{\partial \xi}\left(f(x(\xi))\right)$$

We also have to integrate

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

Consider calculation of stiffness matrix

$$k^e_{ab} = \int_{\Omega^e} N_{a,x}(x)\, N_{b,x}(x)\, dx$$

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

note – $N_{a,x}(x(\xi)) = N_{a,\xi}\, x_{,\xi}(\xi)^{-1}$

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

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