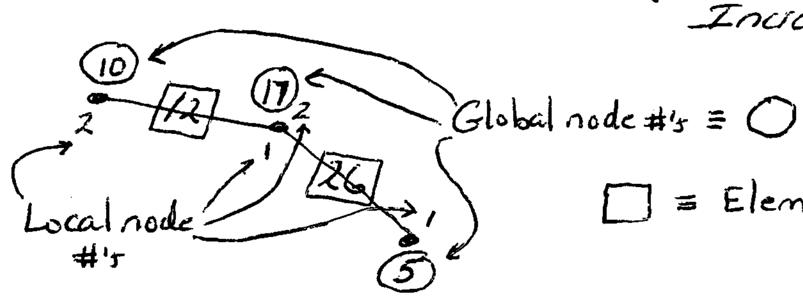
Linear Boundary Elements

2D Boundary element $\Rightarrow$ same as 1D finite element

- Can use same element structure (Global/Local node #'s Incidence list, etc.)



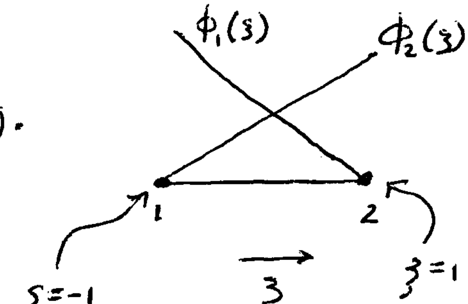
- each boundary node has unique Global node #
- each boundary element has unique element # (with 2 local node #s $\Rightarrow$ 1, 2)

Incidence list: $IN(12,1) = 17$, $IN(12,2) = 10$

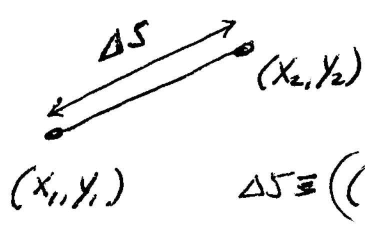
el # — local node # — global node #

- In general, G_1 is complicated so must do $\oint$ by quadrature... best to define element on local coordinate system and use isoparametric mapping

e.g.



$\xrightarrow[x(\xi)\; y(\xi)]{}$



$$\Delta S \equiv \left((x_2-x_1)^2+(y_2-y_1)^2\right)^{1/2}$$

$$\phi_1 = \frac{1-\xi}{2} \qquad x = \sum_{i=1}^{2} x_i\,\phi_i(\xi)$$

$$\phi_2 = \frac{1+\xi}{2} \qquad y = \sum_{i=1}^{2} y_i\,\phi_i(\xi)$$

So... $\int_{be} \phi_j(s)\, G_i(r(s))\, ds = \int_{-1}^{1} G_i(r(\xi))\, \phi_j(\xi)\, |J|\, d\xi$

on a linear line segment $|J| = \frac{\Delta s}{2}$

(Note: $\int_e ds = \Delta s = \int_{-1}^{1} |J|\, d\xi = 2|J|$)

then $\int_{-1}^{1} G_i(r(\xi))\, \phi_j(\xi)\, \frac{\Delta s}{2}\, d\xi \cong \underbrace{\frac{\Delta s}{2} \sum_{k=1}^{M} G_i(r(\xi_k))\, \phi_j(\xi_k)\, W_k}_{\text{Numerical (Gaussian) Quadrature}}$

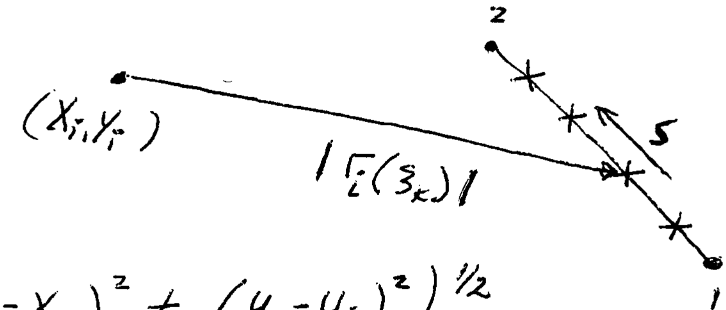
Similarly for $\int_{be} \frac{\partial G_i}{\partial n} \phi_j\, ds \cong \frac{\Delta s}{2} \sum_{k=1}^{M} \left.\frac{\partial G_i}{\partial n}\right|_{r(\xi_k)} \phi_j(\xi_k)\, W_k$

Therefore... all we have to do is evaluate these integrands at Gauss pts; ϕ_j is trivial since already expressed in ξ coordinate system, Δs is just the element length and W_k is dictated by the gauss pts being used

Only issues... how to evaluate $G_i(r(\xi_k))$

$\left.\frac{\partial G_i}{\partial n}\right|_{r(\xi_k)}$

What is $r(\xi_k)$??



$$r_i(s) = \left((x_s - x_i)^2 + (y_s - y_i)^2\right)^{1/2}$$

$$r_i(\xi_k) = \left((x(\xi_k) - x_i)^2 + (y(\xi_k) - y_i)^2 \right)^{1/2}$$

where $x(\xi_k) = x_1 \phi_1(\xi_k) + x_2 \phi_2(\xi_k) \Rightarrow \sum_{j=1}^{2} x_j \phi_j(\xi)$

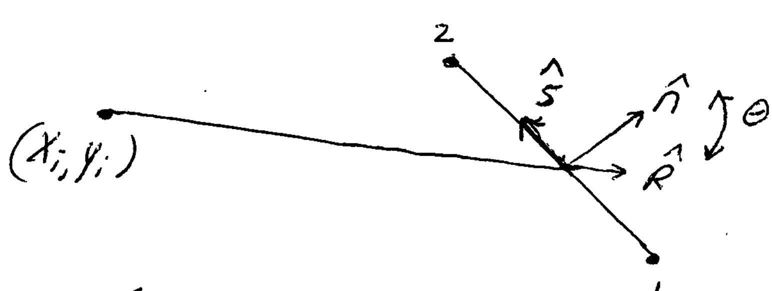
$y(\xi_k) = y_1 \phi_1(\xi_k) + y_2 \phi_2(\xi_k) \Rightarrow \sum_{j=1}^{2} y_j \phi_j(\xi)$

So to compute $G_i(r_i(\xi_k))$ ... first compute r_i at ξ_k
then evaluate G_i

What about $\frac{\partial G_i}{\partial n}(r_i(\xi_k))$??

$\frac{\partial G_i}{\partial n} = \hat{n} \cdot \nabla G_i$ But G_i only a function of radial distance away from $i \Rightarrow \nabla G_i = \frac{\partial G}{\partial R} \hat{R}$

$$\frac{\partial G_i}{\partial n} = \frac{\partial G}{\partial R} \hat{R} \cdot \hat{n}$$



Adopt the convention: $\hat{s}$ points from local node 1 to local node 2 <u>and</u> $\hat{n} \times \hat{s} = \hat{z}$
($\hat{n}$ is <u>outward</u> <u>normal</u>)

so $$\hat{R} = \frac{\underset{\sim}{R}}{|\underset{\sim}{R}|} = \frac{(x_s - x_i)\hat{x} + (y_s - y_i)\hat{y}}{\left((x_s - x_i)^2 + (y_s - y_i)^2\right)^{1/2}}$$

$$\hat{s} = \frac{(x_2 - x_1)\hat{x} + (y_2 - y_1)\hat{y}}{\Delta s}$$

$$\hat{n} = \frac{(y_2 - y_1)\hat{x} - (x_2 - x_1)\hat{y}}{\Delta s} \quad \left(\text{since } \begin{matrix} \hat{s}\cdot\hat{n} = 0 \\ \hat{n}\times\hat{s} = \hat{z} \end{matrix}\right)$$

$$\therefore \quad \hat{R}\cdot\hat{n} = \frac{(x_s - x_i)(y_2 - y_1) - (y_s - y_i)(x_2 - x_1)}{\Delta s\, |\underset{\sim}{R}|}$$

then $$\frac{\partial G_i}{\partial n}(r_i(\xi_k)) = \frac{\partial G_i}{\partial R}(r_i(\xi_k))\, \hat{R}\cdot\hat{n}\Big|_{r_i(\xi_k)}$$

$$= \frac{\partial G_i}{\partial R}\Big|_{r_i(\xi_k)} \left[\frac{(x(\xi_k) - x_i)(y_2 - y_1) - (y(\xi_k) - y_i)(x_2 - x_1)}{\Delta s \left((x(\xi_k) - x_i)^2 + (y(\xi_k) - y_i)^2 \right)^{1/2}} \right]$$

Rule of thumb...

if "i" is not in the element, use quadrature

<u>Otherwise</u>

do integration analytically

Look at the specifics for Laplace's equation

$$\left[\frac{\alpha_i}{2\pi} u_i = \oint \left(\frac{\partial u}{\partial n} G_i - \frac{\partial G_i}{\partial n} u\right) ds\right] * 2\pi \leftarrow \text{Note!}$$

$$\alpha_i u_i = -\sum_j \frac{\partial u}{\partial n_j} \oint \underbrace{\phi_j \ln R_i}_{2\pi * G_i} ds + \sum_j u_j \oint \underbrace{\phi_j \frac{1}{R_i}\frac{\partial R_i}{\partial n}}_{2\pi * \frac{\partial G_i}{\partial n}} ds$$

So $\displaystyle \oint \phi_j \ln R_i \, ds = \sum_e \int_e \phi_j \ln R_i \, ds_e$

$$\oint \phi_j \frac{1}{R_i}\frac{\partial R_i}{\partial n} ds = \sum_e \int_e \phi_j \left(\frac{1}{R_i}\frac{\partial R_i}{\partial n}\right) ds_e$$

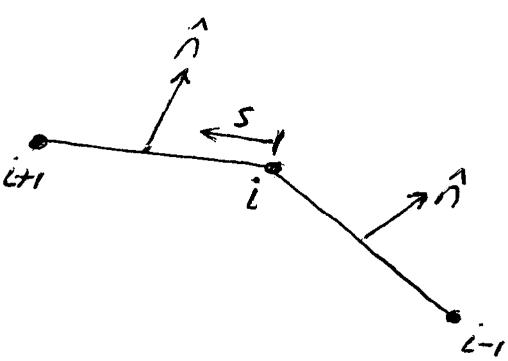
<u>when $i \notin e$</u>

$$\int_e \phi_j \ln R_i \, ds_e \cong \frac{\Delta s_e}{2} \sum_{k=1}^{M} \phi_j(\xi_k) \ln R_i(\xi_k) w_k$$

$$\int_e \phi_j \frac{1}{R_i}\frac{\partial R_i}{\partial n} ds_e \cong \frac{\Delta s_e}{2} \sum_{k=1}^{M} \phi_j(\xi_k) \frac{1}{R_i(\xi_k)} \left.\frac{\partial R_i}{\partial n}\right|_{\xi_k} w_k$$

where $R_i(\xi_k)$, $\left.\frac{\partial R_i}{\partial n}\right|_{\xi_k}$, $\phi_j(\xi_k)$ defined earlier

<u>when $i \in e$</u>



Note: $\frac{\partial G_i}{\partial n} = \frac{\partial G_i}{\partial R}\hat{R}\cdot\hat{n}$ but $\hat{R} = \pm\hat{s}$ and $\hat{s}\cdot\hat{n} = 0$

$\therefore \frac{\partial G_i}{\partial n} \equiv 0$ (on linear element containing i)

So $\underline{\underline{\int_e \frac{\partial G_i}{\partial n}\phi_j\, ds_e = 0}}$ when $i \in e$

Still need $\int_e \phi_j G_i\, ds = \int_e \phi_j \ln R_i\, ds$ for $j = 1, 2$

But $R = s$ and ϕ_j has form $a + bs$ (i.e. is linear)

So we need to compute integrals of the form

$$\int_0^{\Delta s} a \ln s = a\Delta s(\ln\Delta s - 1)$$

$$\int_0^{\Delta s} bs\ln s = b\frac{\Delta s^2}{2}\left(\ln\Delta s - \frac{1}{2}\right)$$

when $j = i$: $a = 1$, $b = -\frac{1}{\Delta s}$

$$\int_e \phi_i \ln R_i\, ds = \Delta s(\ln\Delta s - 1) - \frac{\Delta s}{2}(\ln\Delta s - 1/2)$$

$$= \underline{\underline{\frac{\Delta s}{2}\left(\ln\Delta s - \frac{3}{2}\right)}}$$

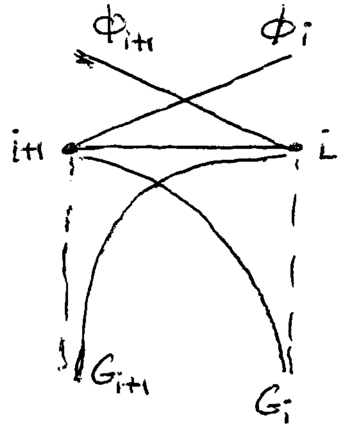
when $j \neq i$: $a = 0$, $b = \frac{1}{\Delta s}$

$$\int_e \phi_j \ln R_i = \underline{\underline{\frac{\Delta s}{2}(\ln\Delta s - 1/2)}}$$

By symmetry on any single element

$$\int_e \phi_i G_i = \int_e \phi_{i+1} G_{i+1}$$

$$\int_e \phi_{i+1} G_i = \int_e \phi_i G_{i+1}$$

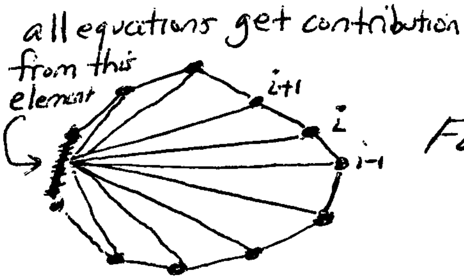


Building a Program ... Guts: Construction of [A], [B]

Two Strategies:

Method I $\Rightarrow$ a. Loop over all elements $\oint = \sum_e \int_e$

b. Loop over all nodes (all G_i)



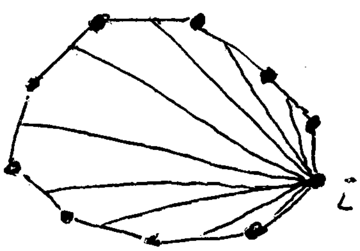
FEM-like ... go to an element and never return ... build all entries from that element ... build matrix "column by column"

$$\begin{bmatrix} & x & & x & \\ & x & & x & \\ & x & & x & \\ & x & & x & \\ & x & & x & \\ & x & & x & \\ & x & & x & \end{bmatrix}$$

global node #5 assoc. w/ local #1, #2

Method II $\Rightarrow$ a. Loop over all nodes (all G_i)

b. Loop over all elements $\Rightarrow \oint = \sum_e \int_e$



builds matrix "row by row"

$$\begin{bmatrix} x & x & x & x & x \end{bmatrix}$$

Matrix Assembly: [A]; [B] ... Method I

Loop over elements; $L = 1, NE$

Load $\left\{ \begin{array}{l} XL(I),\ YL(I) \\ JL(I) \end{array} \right\} I = 1,2$ — local node coordinates; global node #'s

$\Delta S_e = ((XL(2)-XL(1))**2 + (YL(2)-YL(1))**2)^{1/2}$

Loop over Gauss pts; $k = 1, m$

$Z = \xi(k)$ Gauss pt coordinate

$\phi(1) = \frac{1-Z}{2}$; $\phi(2) = \frac{1+Z}{2}$ Define basis

$XS = XL(1) * \phi(1) + XL(2) * \phi(2)$
$YS = YL(1) * \phi(1) + YL(2) * \phi(2)$ $(X,Y)_{\text{Gauss pt}}$

Loop over nodes: $I = 1, NN$

IF (JL(1).EQ.I.or.JL(2).EQ.I) go to next I (skip analytic integrations)

$r_i = ((XS - X(I))**2 + (YS - Y(I))**2)^{1/2}$

$$\frac{\partial r_i}{\partial n} = \frac{(YL(2)-YL(1)) * (XS - X(I)) - (XL(2)-XL(1)) * (YS - Y(I))}{\Delta S_e * r_i}$$

$G_i = -\mathrm{LN}\, r_i$

$$\frac{\partial G_i}{\partial n} = -\frac{1}{r_i} * \frac{\partial r_i}{\partial n}$$

Loop over columns: $J = 1,2$

$A(I, JL(J)) = A(I, JL(J)) + \phi(J) * \frac{\partial G_i}{\partial n} * \frac{\Delta S_e}{2} * W_k$

$B(I, JL(J)) = B(I, JL(J)) + \phi(J) * G_i * \frac{\Delta S_e}{2} * W_k$

END J Loop

END I Loop

END Gauss pt Loop

CALL analytic $(B, A, \Delta S_e, JL, \alpha)$

END element Loop

Matrix Assembly : [A] ; [B] . . . Method II

Loop over nodes ; $I = 1, NN$

Loop over elements ; $L = 1, NE$

Load $\left\{ \begin{matrix} XL(I),\ YL(I) \\ JL(I) \end{matrix} \right\}$ $I = 1,2$ — local node coordinates; global node #'s

$\Delta S_e = \left((XL(2) - XL(1))**2 + (YL(2) - YL(1))**2 \right)^{1/2}$

IF (I.EQ. JL(1).OR. I.EQ. JL(2)) then

call analytic (B, A, ΔS_e, JL, α) } analytic integrations

go to next element

END if

Loop over Gauss points ; $k = 1, m$

$z = \xi(k)$

$\phi(1) = \frac{1-z}{2}$; $\phi(2) = \frac{1+z}{2}$ Define Basis

$XS = XL(1) * \phi(1) + XL(2) * \phi(2)$

$YS = YL(1) * \phi(1) + YL(2) * \phi(2)$ $(x,y)_{\text{Gauss pt}}$

$r_i = \left((XS - X(I))**2 + (YS - Y(I))**2 \right)^{1/2}$

$$\frac{\partial r_i}{\partial n} = \frac{(YL(2) - YL(1)) * (XS - X(I)) - (XL(2) - XL(1)) * (YS - Y(I))}{\Delta S_e * r_i}$$

$G_i = - \ln r_i$

$\frac{\partial G_i}{\partial n} = - \frac{1}{r_i} * \frac{\partial r_i}{\partial n}$

Loop over columns ; $J = 1, 2$

$A(I, JL(J)) = A(I, JL(J)) + \phi(J) * \frac{\partial G_i}{\partial n} * \frac{\Delta S_e}{2} * W_k$

$B(I, JL(J)) = B(I, JL(J)) + \phi(J) * G_i * \frac{\Delta S_e}{2} * W_k$

END J Loop

END Gauss pt loop

END element Loop

$A(I, I) = A(I, I) + \alpha$ * if not in analytic subroutine

END Node Loop

Subroutine Analytic $(B, A, \Delta S_e, JL, \alpha)$

$$B(JL(1), JL(1)) = B(JL(1), JL(1)) + \frac{\Delta S_e}{2}\left(\frac{3}{2} - \ln \Delta S_e\right)$$

$$B(JL(1), JL(2)) = B(JL(1), JL(2)) + \frac{\Delta S_e}{2}\left(\frac{1}{2} - \ln \Delta S_e\right)$$

$$B(JL(2), JL(2)) = B(JL(2), JL(2)) + \frac{\Delta S_e}{2}\left(\frac{3}{2} - \ln \Delta S_e\right)$$

$$B(JL(2), JL(1) = B(JL(2), JL(1)) + \frac{\Delta S_e}{2}\left(\frac{1}{2} - \ln \Delta S_e\right)$$

$$A(JL(1), JL(1)) = A(JL(1), JL(1)) + \frac{\alpha(JL(1))}{2}$$

$$A(JL(2), JL(2)) = A(JL(2), JL(2)) + \frac{\alpha(JL(2))}{2}$$

Return

Left to do ... Apply BCs

Most general case: Type III $\quad c_j u_j + d_j \frac{\partial u}{\partial n_j} = e_j$

$$\begin{array}{cc} & \overset{2N}{\longleftrightarrow} \\ 2N & \left[\begin{array}{c|c} A & -B \\ \hline \ddots\ c\ \ddots & \ddots\ d\ \ddots \end{array}\right] \end{array} \left\{\begin{array}{c} u \\ \hline \frac{\partial u}{\partial n} \end{array}\right\} = \left\{\begin{array}{c} F \\ \hline e \end{array}\right\}$$

if we had Possion instead of Laplace

$2N \times 2N$ system: Storage 2^2 (relative to $N \times N$)
Run Time 2^3

Type I : d_j's $= 0$ $\quad \Rightarrow$ Pivoting issue w/ any Type 1 BCs
Type II : c_j's $= 0$ $\quad$ (or Type II if had written $\left[\begin{array}{c|c} \ddots c \ddots & \ddots d \ddots \\ \hline A & -B \end{array}\right] \left\{\begin{array}{c} u \\ \hline \frac{\partial u}{\partial n} \end{array}\right\}$)

Better to Collapse system to $N \times N$

$$\begin{array}{cc} \left[\begin{array}{c|c} A & -B \end{array}\right] & \left\{\begin{array}{c} u \\ \hline \frac{\partial u}{\partial n} \end{array}\right\} = \{F\} \\ A_j \quad B_j & \end{array}$$

Type I: $-A_j * \frac{e_j}{c_j}$ to RHS

$-B_j$ replaces A_j

$\frac{\partial u}{\partial n}{}_j$ replaces u_j

Type II: $B_j * \frac{e_j}{d_j}$ to RHS

Type III: $B_j * \frac{e_j}{d_j}$ to RHS (same as Type II)

$-B_j * \frac{c_j}{d_j}$ add to A_j

Single $N \times N$ system:

$$\left[\begin{array}{c} A_j - \frac{c_j}{d_j} B_j \quad (\text{Type II or III}) \\ \text{or} \quad -B_j \quad (\text{Type I}) \end{array}\right] \left\{\begin{array}{c} u \ (\text{Type II or III}) \\ \text{or} \\ \frac{\partial u}{\partial n} \ (\text{Type I}) \end{array}\right\} = \{F\} - \sum_{\text{Type I}} A_j \frac{e_j}{c_j} + \sum_{\text{Type II or III}} B_j \frac{e_j}{d_j}$$

have $[A']\{X\} = \{F'\}$

where

$$a'_{ij} = a_{ij} - \frac{c_j}{d_j} b_{ij} \qquad (\text{Type II or III})$$

$$= -b_{ij} \qquad (\text{Type I})$$

$$x_j = u_j \qquad (\text{Type II or III})$$

$$\frac{\partial U}{\partial n_j} \qquad (\text{Type I})$$

$$f_i' = f_i - \sum_{j=\text{Type I}} a_{ij}\left(\frac{e_j}{c_j}\right) + \sum_{j=\text{Type II or III}} b_{ij}\left(\frac{e_j}{d_j}\right)$$