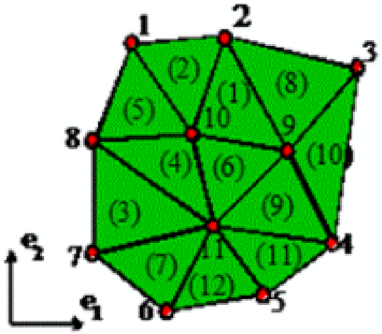
Review – General FEA for linear elasticity

- Goal: set up and solve system of linear equations for unknown displacements at nodes $K_{aibk}\Delta u_k^b = R_i^a + F_i^a$



$$K_{aibk} = \sum_{elements} \int_{\Omega_{el}} C_{ijkl} \frac{\partial N^a(\mathbf{x})}{\partial x_j} \frac{\partial N^b(\mathbf{x})}{\partial x_l} dV \quad R_i^a = -\sum_{elements} \int_{\Omega_{el}} \sigma_{ij}^0 \frac{\partial N^a(\mathbf{x})}{\partial x_j} dV \quad F_i^a = \sum_{faces} \int_{S_j} t_i^* N^a(\mathbf{x}) dA$$

To assemble $[K], \underline{R}$: Loop over elements

2. For a generic element:
   - Initialize integration points and weights w_i, ξ_i ;
   - Loop over integration points: for a generic integration point:
     - Calculate shape function derivatives $\dfrac{\partial N^a}{\partial \xi_i}$
     - Calculate the Jacobian matrix $\dfrac{\partial x_i}{\partial \xi_j} = x_i^a \dfrac{\partial N^a}{\partial \xi_j}$
     - Calculate the spatial shape function derivatives $\dfrac{\partial N^a}{\partial x_i} = \dfrac{\partial N^a}{\partial \xi_j}\dfrac{\partial \xi_j}{\partial x_i}$
     - Assemble [B] matrix
     - Add contribution to $[k^{el}] = \sum_{i=1}^{NINTP} w_i [B(\xi_i)]^T [D][B(\xi_i)]\eta(\xi_i)$

$$\underline{R}^{el} = -\sum_{i=1}^{NINTP} w_i [B(\xi_i)]^T \underline{\sigma}\eta(\xi_i)$$

   - Add element stiffness to global stiffness

Solve $[K]\Delta\underline{u} = \underline{R} + \underline{F}$

Topics for todays class

- Illustrate 3D linear elastic FEA as an ABAQUS UEL
- Brief look at HW3
- Accuracy and convergence of FEA for linear elasticity
  - FEA as a Galerkin method
  - Example of Galerkin method for solving beam equations

Coding an element as an ABAQUS UEL

```
      SUBROUTINE UEL(RHS,AMATRX,SVARS,ENERGY,NDOFEL,NRHS,NSVARS,
     1     PROPS,NPROPS,COORDS,MCRD,NNODE,U,DU,V,A,JTYPE,TIME,DTIME,
     2     KSTEP,KINC,JELEM,PARAMS,NDLOAD,JDLTYP,ADLMAG,PREDEF,NPREDF,
     3     LFLAGS,MLVARX,DDLMAG,MDLOAD,PNEWDT,JPROPS,NJPROP,PERIOD)
!
      INCLUDE 'ABA_PARAM.INC'
!
!
      DIMENSION RHS(MLVARX,*),AMATRX(NDOFEL,NDOFEL),PROPS(*),
     1   SVARS(*),ENERGY(8),COORDS(MCRD,NNODE),U(NDOFEL),
     2   DU(MLVARX,*),V(NDOFEL),A(NDOFEL),TIME(2),PARAMS(*),
     3   JDLTYP(MDLOAD,*),ADLMAG(MDLOAD,*),DDLMAG(MDLOAD,*),
     4   PREDEF(2,NPREDF,NNODE),LFLAGS(*),JPROPS(*)
```

Output Variables: RHS(I,1) $= R_{ai}$

AMATRX(I,J) $= K_{aibk}$

→ Optional: SVARS(I) – store int pt data

ENERGY – various energy measures

PNEWDT – Controls time steps

Everything else is input - eg U (I) displacement @ end of inc

DU(I) change in displacement

other vars available for info

Sample files are provided on EN234_FEA Github repository.

You can test a UEL using EN234_FEA codes and then run with ABAQUS

See course website for tutorials etc

```
!      --  Loop over integration points
  do kint = 1, n_points
    call abq_UEL_3D_shapefunctions(xi(1:3,kint),NNODE,N,dNdxi)
    dxdxi = matmul(coords(1:3,1:NNODE),dNdxi(1:NNODE,1:3))
    call abq_UEL_invert3d(dxdxi,dxidx,determinant)
    dNdx(1:NNODE,1:3) = matmul(dNdxi(1:NNODE,1:3),dxidx)
    B = 0.d0
    B(1,1:3*NNODE-2:3) = dNdx(1:NNODE,1)
    B(2,2:3*NNODE-1:3) = dNdx(1:NNODE,2)
    B(3,3:3*NNODE:3)   = dNdx(1:NNODE,3)
    B(4,1:3*NNODE-2:3) = dNdx(1:NNODE,2)
    B(4,2:3*NNODE-1:3) = dNdx(1:NNODE,1)
    B(5,1:3*NNODE-2:3) = dNdx(1:NNODE,3)
    B(5,3:3*NNODE:3)   = dNdx(1:NNODE,1)
    B(6,2:3*NNODE-1:3) = dNdx(1:NNODE,3)
    B(6,3:3*NNODE:3)   = dNdx(1:NNODE,2)

    strain = matmul(B(1:6,1:3*NNODE),U(1:3*NNODE))

    stress = matmul(D,strain)
    RHS(1:3*NNODE,1) = RHS(1:3*NNODE,1)
     - matmul(transpose(B(1:6,1:3*NNODE)),stress(1:6))*
                                          w(kint)*determinant

    AMATRX(1:3*NNODE,1:3*NNODE) = AMATRX(1:3*NNODE,1:3*NNODE)
    + matmul(transpose(B(1:6,1:3*NNODE)),matmul(D,B(1:6,1:3*NNODE)))
                                             *w(kint)*determinant

    ENERGY(2) = ENERGY(2)
     + 0.5D0*dot_product(stress,strain)*w(kint)*determinant           ! Store the elastic strain energy

    if (NSVARS>=n_points*6) then    ! Store stress at each integration point (if space was allocated to do so)
        SVARS(6*kint-5:6*kint) = stress(1:6)
    endif
  end do
```

get N^a, $dN^a/d\xi_i$

$[\partial x_i/\partial \xi_j]$

$dN^a/dx_i = dN^a/d\xi_k \; \partial\xi_k/\partial x_i$

$[B]$

$\underline{\varepsilon} = [B]\,\underline{u}$

$\underline{R} = \sum_{i=1}^{NINTP} [B]^T \underline{\sigma}\, w_i\, \eta$

$[K] = \sum [B]^T [D] [B]\, w_i\, \eta$

These are defined for plotting

5) Perspectives on FEA for linear elasticity

5.1) FEA as a Galerkin method

PDE $\quad \dfrac{\partial}{\partial x_j}\left\{ C_{ijkl} \dfrac{\partial u_k}{\partial x_l} \right\} = 0 \quad$ (equilibrium)

$$t_i^* - C_{ijkl} \frac{\partial u_k}{\partial x_l} n_j = 0 \quad \text{(traction BC) on } S_2$$

$$u_i = u_i^* \quad \text{on } S_1$$

① "Weak form" of PDE: Let η_i be a test function with $\eta_i = 0$ on S_1. Then

$$\int_{R} \frac{\partial}{\partial x_j}\left\{ C_{ijkl} \frac{\partial u_k}{\partial x_l} \right\} \eta_i \, dV + \int_{S_2} \left\{ t_i^* - C_{ijkl} \frac{\partial u_k}{\partial x_l} n_j \right\} \eta_i = 0 \quad *$$

Integrate first term by parts:

Note $\frac{\partial}{\partial x_j}\left\{ C_{ijkl}\frac{\partial u_k}{\partial x_l}\right\}\eta_i = \frac{\partial}{\partial x_j}\left\{ C_{ijkl}\frac{\partial u_k}{\partial x_l}\eta_i\right\}$

$$- C_{ijkl}\frac{\partial u_k}{\partial x_l}\frac{\partial \eta_i}{\partial x_j}$$

Recall $\int_R \frac{\partial}{\partial x_j}\phi_j \, dV = \int_S \phi_j n_j \, dA$

(Note also $\eta_i = 0$ on S_1)

Hence * becomes

$$\boxed{-\int_R C_{ijkl}\frac{\partial u_k}{\partial x_l}\frac{\partial \eta_i}{\partial x_j}\, dV + \int_{S_2} t_i^* \eta_i \, dA = 0}$$

Weak form: must hold $\forall$ admiss η_i

Introduce interpolation:

$$u_i = N^a u_i^a \qquad \eta_i = N^a \eta_i^a$$

Put into weak form

$$\left(-K_{aibk} u_k^b + f_{ai}\right)\eta_i^a = 0 \quad \forall \text{ admis } \eta_i$$

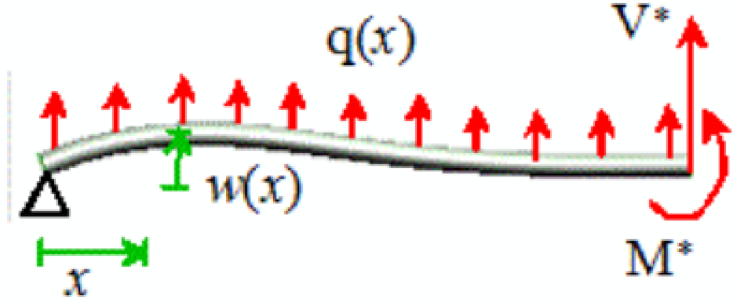
K_{aibk}, f_{ia} are usual FEA stiffness and ext force

Usual argument leads to FEA equations

We can use this to solve many PDEs or ODEs

Example: Euler-Bernoulli beam

Equations: $EI\dfrac{d^4w}{dx^4} - q(x) = 0$



$$\left.\begin{aligned} M^* - EI\frac{d^2w}{dx^2} &= 0 \\ V^* + EI\frac{d^3w}{dx^3} &= 0 \end{aligned}\right\} \text{ Loaded ends}$$

$$\left.\begin{aligned} w &= w^* \\ \frac{dw}{dx} &= \theta^* \end{aligned}\right\} \text{ constrained ends}$$

Galerkin: Let $\eta(x)$ be a test function with $\eta = 0$ $\dfrac{d\eta}{dx} = 0$ on constrained ends

$$\int_0^L \left(\underbrace{EI \frac{d^4 w}{dx^4}}_{\text{Integrate by parts twice}} - q(x) \right) \eta \, dx - \left[\left(M^* - EI \frac{d^2 w}{dx^2} \right) \frac{d\eta}{dx} \right]_0^L - \left[\left(V^* + EI \frac{d^3 w}{dx^3} \right) \eta \right]_0^L = 0$$

$$\int_0^L u \, dv = [uv]_0^L - \int v \, du$$

$$\boxed{\Rightarrow \int_0^L \left(EI \frac{d^2 w}{dx^2} \frac{d^2 \eta}{dx^2} - q\eta \right) dx - \left[M^* \frac{d\eta}{dx} \right]_0^L - \left[V^* \eta \right]_0^L = 0 \quad \forall \text{ admiss } \eta}$$

Introduce FE interpolation

$$w = N^b w^b \qquad \eta = N^a \eta^a$$

$$\left[K_{ab} w^b - f_a \right] \eta^a = 0 \quad \forall \; \eta^a$$

$$K_{ab} = \int_0^L EI \frac{d^2 N^b}{dx^2} \frac{d^2 N^a}{dx^2} \, dx$$

$$f_a = \int_0^L q N^a \, dx + \left[M^* \frac{dN^a}{dx} \right]_0^L + \left[V^* N^a \right]_0^L$$

With constraints $\left. \begin{array}{l} N^a w^a = w^* \\ \dfrac{dN^a}{dx} w^a = \theta^* \end{array} \right\}$ on constrained ends

we have $[K] \underline{w} = \underline{f}$ ⇐ FE equations