ME535

Finite Element Analysis of Structures

FEM theory and computations

2023

Instructor: Phill-Seung Lee, KAIST

Textbook: KJ Bathe, Finite element procedures, 1996, 2014

Contents

Appendix

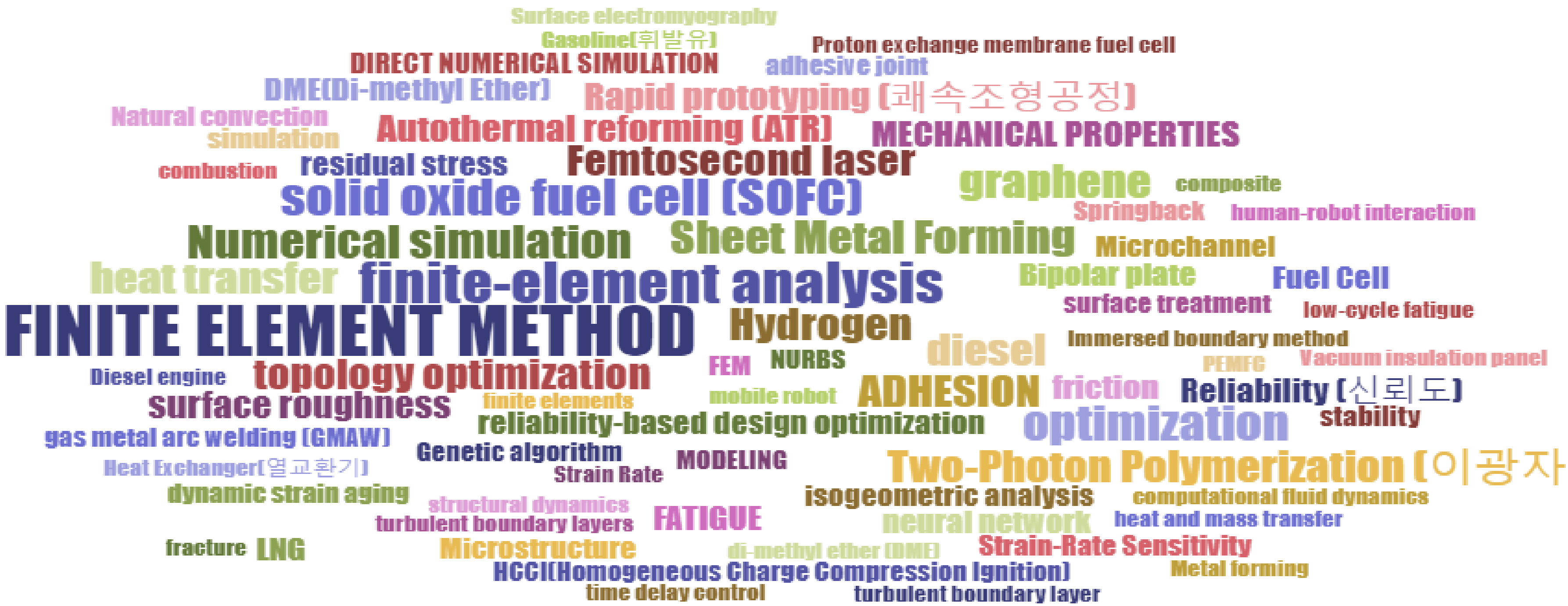
A1. Research Area Map

A2. Implementation of FE Code

A3. ADINA Sessions

A4. Problem Sets

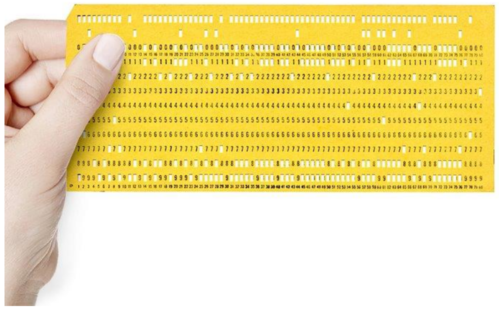
Introduction



Keyword trend analysis result in mechanical engineering (1970-2020)

History of Finite Element Method

- Matrix Algebra, 1858, invented by Cayley… also by Grassmann
- Matrix Structural Analysis, 1930 (World war I), by Collar ad Duncan
- First electronic commercial computer (UNIVAC I), 1951



- Delta Wing Challenge, early 1950s, by Turner & Clough

  → "Finite element" was first named.

- Energy theorems and structural analysis: Basic concepts of FEM, early 1950s, by JH Argyris
- First book on FEM, 1967, by Zienkiewicz
- Mathematical foundation, 1973, by Strang → generalized and applied to various engineering fields
- Most commercial FEM software were originated in the 1970s

  → ABAQUS, ADINA, ANSYS ......

What is Finite Element Method?

- Physical problems in engineering and applied science
- Mathematical model (PDE, Partial Differential Equations)
- Simple problems → Exact closed-form solutions

- Most real-world problems: Very Complex (geometry, properties, boundary conditions)
- Exact solutions cannot be obtained.
- Approximate solutions are necessary!

- Finite Element Method (FEM)
  - ✓ One of approximate solution techniques. (actually, most successful in solid mechanics and structural engineering)
  - ✓ One of numerical solution methods (ex. FDM, FVM, BEM……)

- FEM can easily handle very complex geometry (key to success), boundary conditions, linear/nonlinear and static/dynamic problems.

- "Commercialized"

Applications of FEM

- Stress and thermal analyses of industrial parts such as electronic chips, electric devices, valves, pipes, pressure vessels, automotive engines and aircraft
- Seismic analysis of dams, power plants, cities and high-rise buildings
- Crash analysis of cars, trains, ships and aircraft
- Fluid flow analysis of coolant ponds, pollutants and contaminants, and air in ventilation systems
- Electromagnetic analysis of antennas, transistors and aircraft signature
- Analysis of surgical procedures such as plastic surgery, jaw reconstruction, correction of scoliosis and many others
- Fluid-structure interactions of wind turbines, floating structures, and many complicated mechanical systems

1. Tensors and Solid Mechanics

Vector, Matrix and Tensor

- Scalar α : A physical quantity that can be completely described by a real number.

  (Ex) length, temperature, …

- Vector $\mathbf{a}$: A physical quantity that can be completely described by an array of numbers.

  (Ex) force, velocity, moment, …

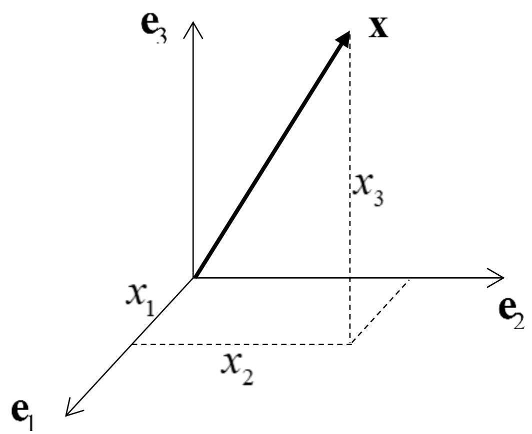
- Matrix $\mathbf{A}$: A physical quantity that can be completely described by an array of ordered numbers.

  (Ex) stress, strain, curvature, …

What is Tensor? → "Generalized quantity"

Tensor = Scalar ∪ Vector ∪ Matrix ∪ more … etc

(1) Vector algebra



(Cartesian coordinate system)

Geometric vector in 3D (position vector)

$\mathbf{x} = x_1\mathbf{e}_1 + x_2\mathbf{e}_2 + x_3\mathbf{e}_3$ with $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$: base vectors of the Cartesian coordinate system

Note that the geometric representation of $\mathbf{x}$ depends completely on the coordinate system chosen.

In general, the geometric vector is given by components and base vectors

$$\mathbf{x} = \sum_{i=1}^{3} x_i \mathbf{q}_i \quad \rightarrow \quad \mathbf{x} = x_1 \mathbf{q}_1 + x_2 \mathbf{q}_2 + x_3 \mathbf{q}_3$$

Summation convention (Einstein convention)

$$\mathbf{x} = \sum_{i=1}^{3} x_i \mathbf{q}_i = x_i \mathbf{q}_i$$

Tensorial representation of a vector

$$\mathbf{x} = x_i \mathbf{q}_i$$

In the Cartesian coordinate system given,

$$\mathbf{x} = x_i \mathbf{e}_i \quad \rightarrow \quad x_i$$

$$\mathbf{u} = u_i \mathbf{e}_i \quad \rightarrow \quad u_i$$

$$\mathbf{v} = v_i \mathbf{e}_i \quad \rightarrow \quad v_i$$

- Vector sum

$$\mathbf{c} = \mathbf{a} + \mathbf{b} \quad \rightarrow \quad c_i = a_i + b_i$$

- Scalar multiplication

$$\mathbf{b} = \alpha\mathbf{a} \quad \rightarrow \quad b_i = \alpha a_i$$

- Dot product

$$\alpha = \mathbf{a}\cdot\mathbf{b} = a_1b_1 + a_2b_2 + a_3b_3 \quad \rightarrow \quad \alpha = a_ib_i$$

- Cross product

$$\mathbf{c} = \mathbf{a}\times\mathbf{b} = (a_2b_3 - a_3b_2)\mathbf{e}_1 + (a_3b_1 - a_1b_3)\mathbf{e}_2 + (a_1b_2 - a_2b_1)\mathbf{e}_3 \quad \rightarrow \quad c_i = \varepsilon_{ijk}a_jb_k$$

ε_{ijk} is "permutation symbol" defined as $\varepsilon_{ijk} = \begin{cases} 0 & \text{for } i = j,\ j = k \text{ or } k = i \\ 1 & \text{for } i, j, k \in \{(1,2,3),(2,3,1),(3,1,2)\} \\ -1 & \text{for } i, j, k \in \{(1,3,2),(3,2,1),(2,1,3)\} \end{cases}$

(2) Matrix algebra

A set of linear equations

$$3x_1 + 3x_2 - x_3 = 1$$
$$2x_1 + 7x_2 + 3x_3 = -3$$
$$x_1 - x_2 - x_3 = 4$$

In matrix form,

$$\begin{bmatrix} 3 & 3 & -1 \\ 2 & 7 & 3 \\ 1 & -1 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -3 \\ 4 \end{bmatrix}, \qquad \mathbf{Ax} = \mathbf{b}$$

In 3D geometry,

$$\mathbf{A} = \sum_{i=1}^{3} \sum_{j=1}^{3} a_{ij} (\mathbf{e}_i \otimes \mathbf{e}_j) = a_{ij} (\mathbf{e}_i \otimes \mathbf{e}_j)$$ (a_{ij} : Components of matrix $\mathbf{A}$)

(Note) $\otimes$: Tensor product

$$\mathbf{e}_1 \otimes \mathbf{e}_2 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{A} = a_{11}(\mathbf{e}_1 \otimes \mathbf{e}_1) + a_{12}(\mathbf{e}_1 \otimes \mathbf{e}_2) + \cdots\cdots\cdots + a_{33}(\mathbf{e}_3 \otimes \mathbf{e}_3)$$

$$= a_{11} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + a_{12} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \cdots\cdots\cdots + a_{33} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Matrix sum

$$\mathbf{C} = \mathbf{A} + \mathbf{B} \rightarrow \qquad c_{ij} = a_{ij} + b_{ij}$$

- Scalar multiplication

$$\mathbf{B} = \alpha \mathbf{A} \quad \rightarrow \qquad b_{ij} = \alpha a_{ij}$$

- Matrix multiplication

$$\mathbf{A} = \mathbf{B}\mathbf{C} \quad \rightarrow \qquad a_{ij} = b_{ik} c_{kj}$$

- Identity matrix

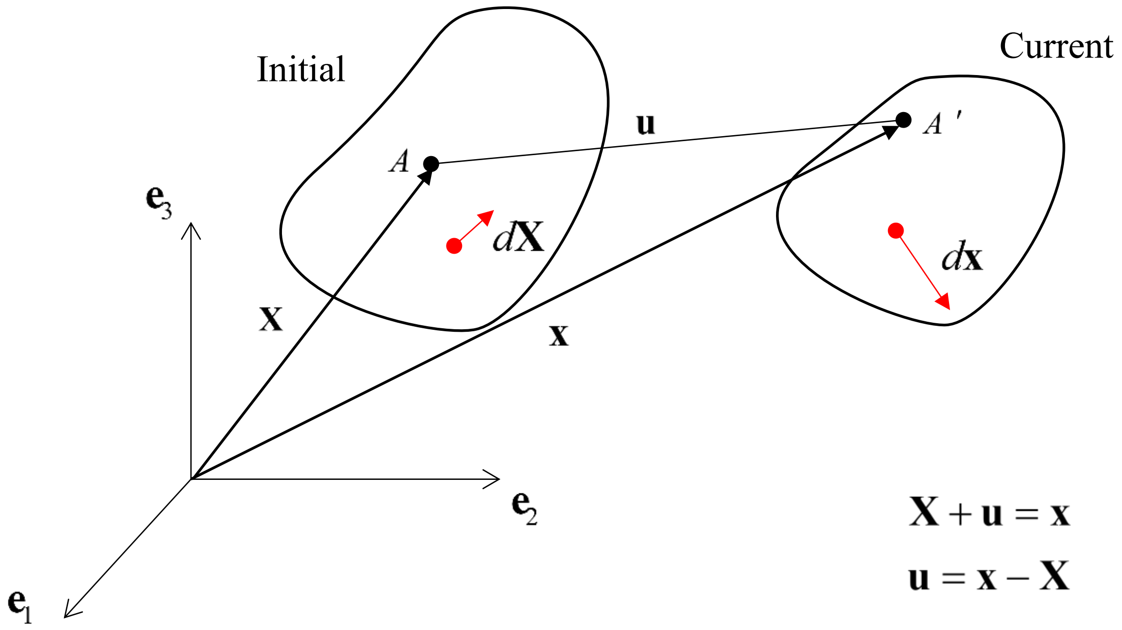
$$\mathbf{I} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \rightarrow \qquad \delta_{ij} = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases}$$ (Kronecker delta)

- A set of linear equations

$$\mathbf{A}\mathbf{x} = \mathbf{b} \quad \rightarrow \qquad a_{ij} x_j = b_i$$

Review of Solid Mechanics

(1) Deformation and strain



$$\mathbf{X} + \mathbf{u} = \mathbf{x}$$

$$\mathbf{u} = \mathbf{x} - \mathbf{X}$$

$$x_1 = x_1(X_1, X_2, X_3)$$

$$x_2 = x_2(X_1, X_2, X_3)$$

$$x_3 = x_3(X_1, X_2, X_3)$$

Let us consider material vectors $d\mathbf{X}$ and $d\mathbf{x}$ in the initial and current configurations.

$$d\mathbf{X} = \begin{bmatrix} dX_1 \\ dX_2 \\ dX_3 \end{bmatrix}, \quad d\mathbf{x} = \begin{bmatrix} dx_1 \\ dx_2 \\ dx_3 \end{bmatrix}$$

Then, the following relation is given

$$d\mathbf{x} = \mathbf{F} d\mathbf{X},$$

in which $\mathbf{F}$ is “deformation tensor” (deformation gradient).

$$\mathbf{F} = \frac{\partial x_i}{\partial X_j}(\mathbf{e}_i \otimes \mathbf{e}_j) = \begin{bmatrix} \frac{\partial x_1}{\partial X_1} & \frac{\partial x_1}{\partial X_2} & \frac{\partial x_1}{\partial X_3} \\ \frac{\partial x_2}{\partial X_1} & \frac{\partial x_2}{\partial X_2} & \frac{\partial x_2}{\partial X_3} \\ \frac{\partial x_3}{\partial X_1} & \frac{\partial x_3}{\partial X_2} & \frac{\partial x_3}{\partial X_3} \end{bmatrix}$$

Strain is the geometrical measure of deformation representing the "relative displacements" between material points.

Green-Lagrange strain tensor

$$\mathbf{E} = \frac{1}{2}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) = \begin{bmatrix} E_{11} & E_{12} & E_{13} \\ E_{21} & E_{22} & E_{23} \\ E_{31} & E_{32} & E_{33} \end{bmatrix}$$

When deformation is "small", the small strain tensor $\boldsymbol{\varepsilon}$ is defined as

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) = \begin{bmatrix} \dfrac{\partial u_1}{\partial x_1} & \dfrac{1}{2}\left(\dfrac{\partial u_1}{\partial x_2} + \dfrac{\partial u_2}{\partial x_1}\right) & \dfrac{1}{2}\left(\dfrac{\partial u_1}{\partial x_3} + \dfrac{\partial u_3}{\partial x_1}\right) \\ \dfrac{1}{2}\left(\dfrac{\partial u_2}{\partial x_1} + \dfrac{\partial u_1}{\partial x_2}\right) & \dfrac{\partial u_2}{\partial x_2} & \dfrac{1}{2}\left(\dfrac{\partial u_2}{\partial x_3} + \dfrac{\partial u_3}{\partial x_2}\right) \\ \dfrac{1}{2}\left(\dfrac{\partial u_3}{\partial x_1} + \dfrac{\partial u_1}{\partial x_3}\right) & \dfrac{1}{2}\left(\dfrac{\partial u_3}{\partial x_2} + \dfrac{\partial u_2}{\partial x_3}\right) & \dfrac{\partial u_3}{\partial x_3} \end{bmatrix}$$

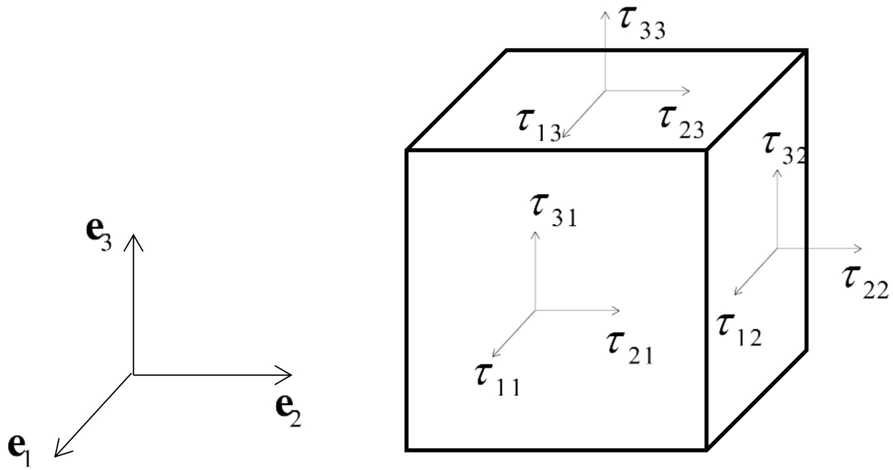
(2) Momentum balance and stress

(Newton's law) → (Momentum balance)

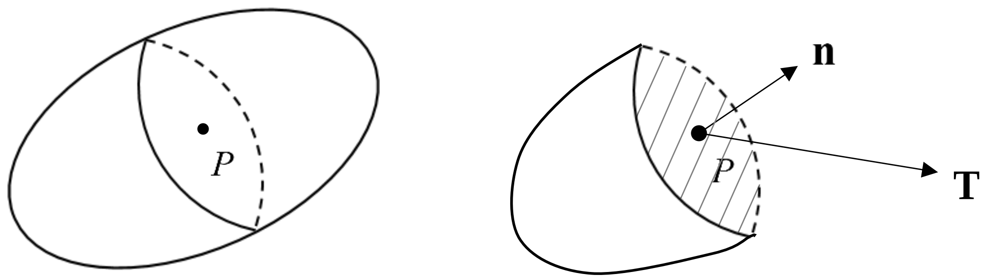
$\mathbf{F} = m\mathbf{a}$

Linear momentum : $m\mathbf{V}$

Angular momentum : $\mathbf{x} \times m\mathbf{V}$



$$\boldsymbol{\tau} = \tau_{ij}(\mathbf{e}_i \otimes \mathbf{e}_j) = \begin{bmatrix} \tau_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \tau_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \tau_{33} \end{bmatrix}$$ (Cauchy stress tensor)



(a) Traction

$$\tau_{ij} n_j = T_i \qquad \text{or} \qquad \boldsymbol{\tau}\mathbf{n} = \mathbf{T}$$

(b) Local equilibrium

$$\tau_{ij,j} + f_i^{\,B} = 0 \qquad \text{or} \qquad div(\boldsymbol{\tau}) + \mathbf{f}^{\,B} = 0 \qquad \leftarrow \; \tau_{ij,j} = \frac{\partial \tau_{ij}}{\partial x_j}$$

$$\frac{\partial \tau_{11}}{\partial x_1} + \frac{\partial \tau_{12}}{\partial x_2} + \frac{\partial \tau_{13}}{\partial x_3} + f_1^{\,B} = 0$$

$$\frac{\partial \tau_{21}}{\partial x_1} + \frac{\partial \tau_{22}}{\partial x_2} + \frac{\partial \tau_{23}}{\partial x_3} + f_2^{\,B} = 0$$

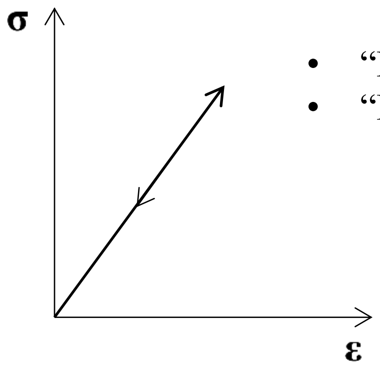
$$\frac{\partial \tau_{31}}{\partial x_1} + \frac{\partial \tau_{32}}{\partial x_2} + \frac{\partial \tau_{33}}{\partial x_3} + f_3^{\,B} = 0$$

(3) Elasticity and material law

$\boldsymbol{\tau} = \boldsymbol{\tau}(\boldsymbol{\varepsilon})$ or $\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\boldsymbol{\tau})$

For linear 3D elastic material: $\boldsymbol{\tau} = \mathbf{C} : \boldsymbol{\varepsilon}$ → $\tau_{ij} = C_{ijkl}\varepsilon_{kl}$ with C_{ijkl}: 4th order tensor

Isotropic linear elastic material



- "Linear"
- "Elastic": loading path = unloading path

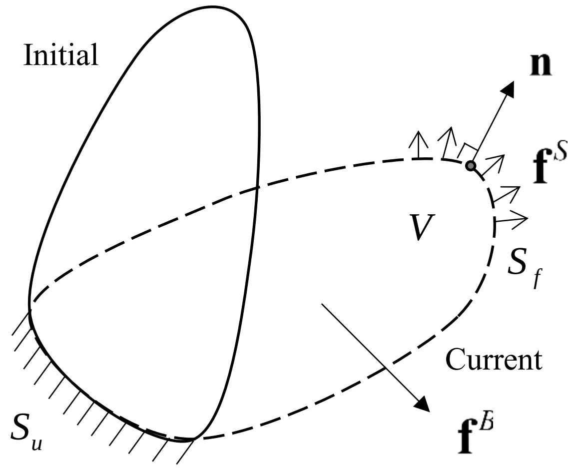
$$\tau_{ij} = \lambda\varepsilon_{kk}\delta_{ij} + 2G\varepsilon_{ij}$$ with Lame's constants λ and G

(4) Governing equations

- Equilibrium in V and on S_f

$$\tau_{ij,j} + f_i^B = 0 \qquad \text{in } V \quad \text{(Local equilibrium)}$$

$$\tau_{ij} n_j = f_i^S \qquad \text{on } S_f \quad \text{(Force BC)}$$



- Compatibility

$$u_i = 0 \quad \text{on } S_u \quad \text{(Displacement BC)}$$

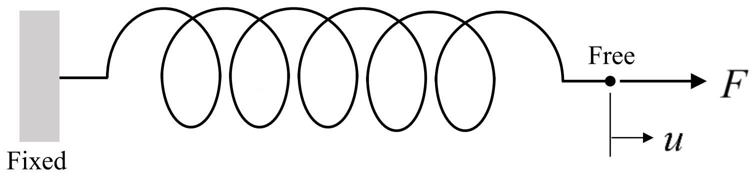
- Material law (stress-strain law)

$$\tau_{ij} = \tau_{ij}(\varepsilon_{kl})$$

2. Matrix Structural Analysis

Hooke's law

Single DOF spring system:



force

$$ku = F$$

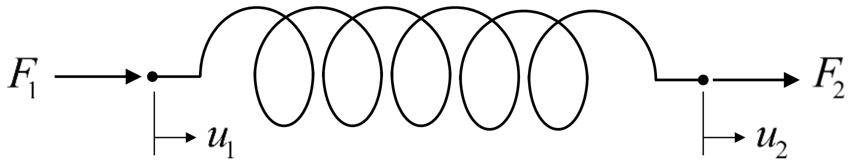
stiffness of spring

displacement

Degrees of freedom (DOFs):

DOFs are the set of independent variables to describe a system. In mechanics, DOFs are the set of independent displacements and/or rotations.

A spring with free boundary



$$\begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$; 2 DOFs system

Stiffness matrix

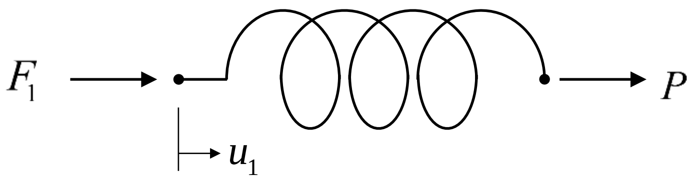
Displacement vector

Force vector

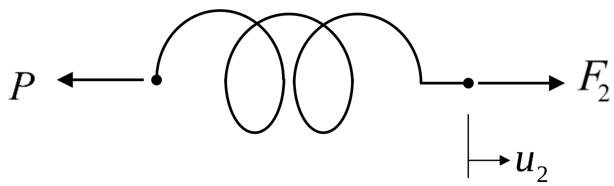
$\mathbf{KU} = \mathbf{F}$; "Equilibrium eqn of a spring"

Let us find the stiffness matrix $\mathbf{K}$ of the single spring.

FBD 1



FBD 2



P : internal force (tension +, compression -)

FBD : free body diagram

Equilibrium

$$F_1 + P = 0 \quad \text{for FBD 1}$$

$$-P + F_2 = 0 \quad \text{for FBD 2} \qquad \rightarrow \ \therefore F_1 = -P, \quad F_2 = P$$

Stretch kinematics

$$u_2 - u_1 = \Delta u$$

Hooke's law

$$k\Delta u = P$$

$$k\left(u_2 - u_1\right) = P$$

A set of 2 linear equations

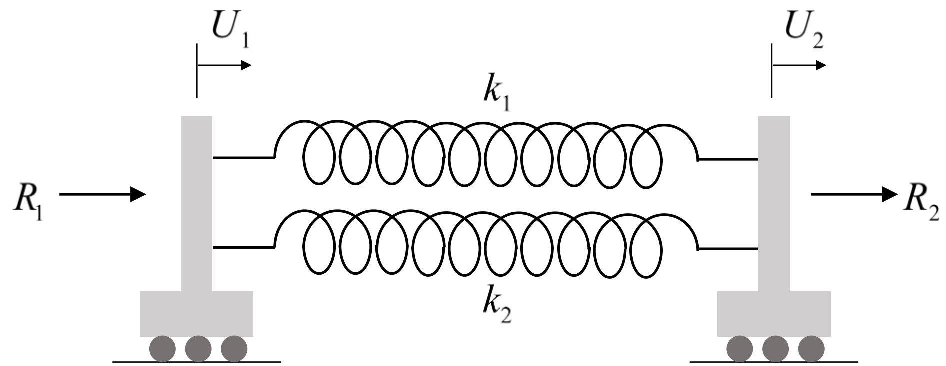
$$-k\left(u_2 - u_1\right) = F_1$$

$$k\left(u_2 - u_1\right) = F_2$$

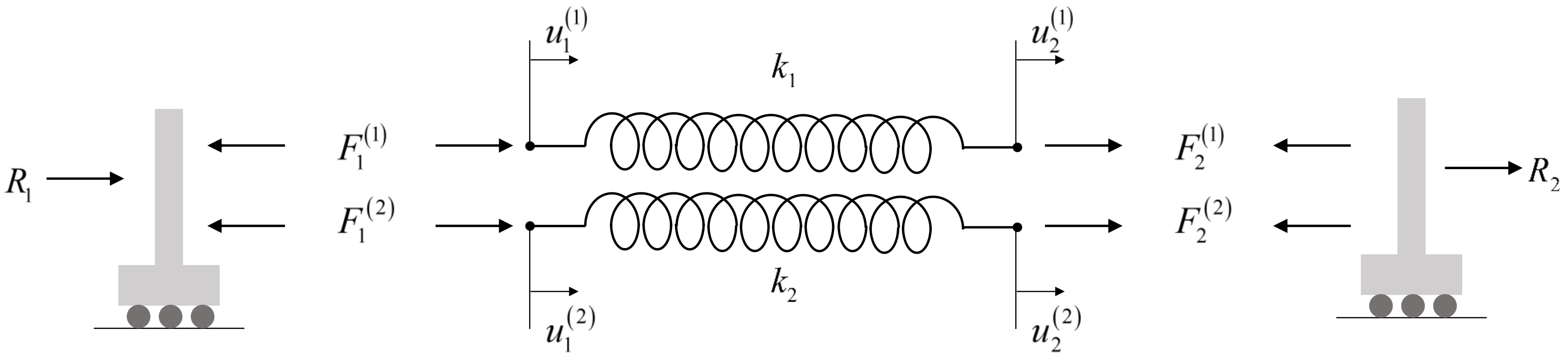
In matrix form

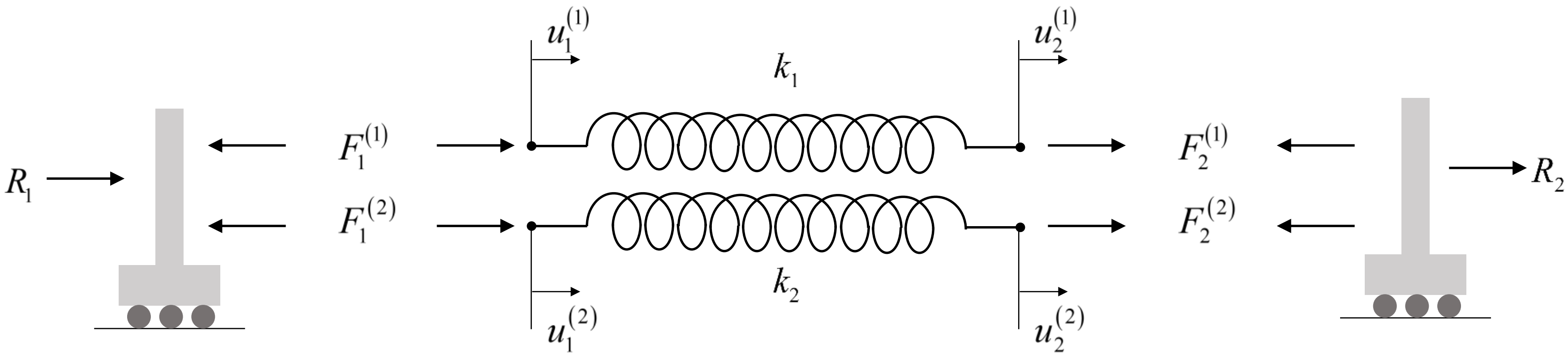
$$\begin{bmatrix} k & -k \\ -k & k \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$; symmetric $\mathbf{K} = \mathbf{K}^T$

Two spring system - Ⅰ (Parallel connection)



Free body diagram





Compatibility of displacement

$$u_1^{(1)} = u_1^{(2)} = U_1$$
$$u_2^{(1)} = u_2^{(2)} = U_2 \quad \cdots \text{ (eq. 2.1)}$$

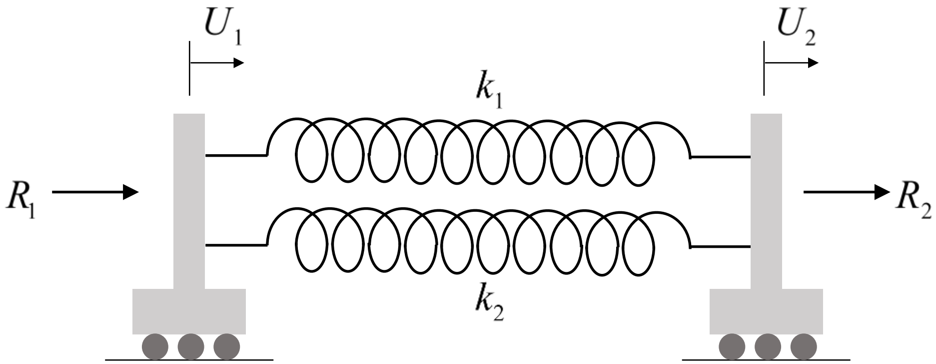
Equilibrium of force

$$F_1^{(1)} + F_1^{(2)} = R_1$$
$$F_2^{(1)} + F_2^{(2)} = R_2$$
$$\rightarrow \begin{bmatrix} F_1^{(1)} \\ F_2^{(1)} \end{bmatrix} + \begin{bmatrix} F_1^{(2)} \\ F_2^{(2)} \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \quad \text{... (eq. 2.2)}$$

Stiffness matrices of each springs

$$\begin{bmatrix} k_1 & -k_1 \\ -k_1 & k_1 \end{bmatrix} \begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \end{bmatrix} = \begin{bmatrix} F_1^{(1)} \\ F_2^{(1)} \end{bmatrix} \text{ for spring 1}$$

$$\begin{bmatrix} k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{bmatrix} u_1^{(2)} \\ u_2^{(2)} \end{bmatrix} = \begin{bmatrix} F_1^{(2)} \\ F_2^{(2)} \end{bmatrix} \text{ for spring 2} \qquad \text{... (eq. 2.3)}$$



Applying the compatibility and equilibrium

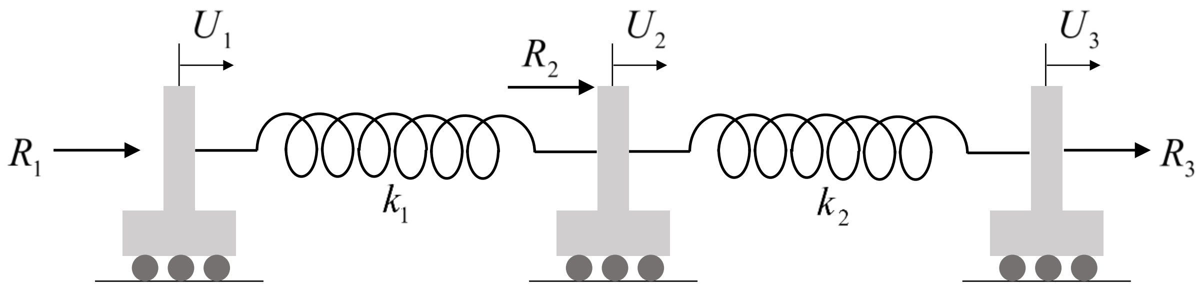
$$\begin{bmatrix} k_1 & -k_1 \\ -k_1 & k_1 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} + \begin{bmatrix} k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}$$

Finally

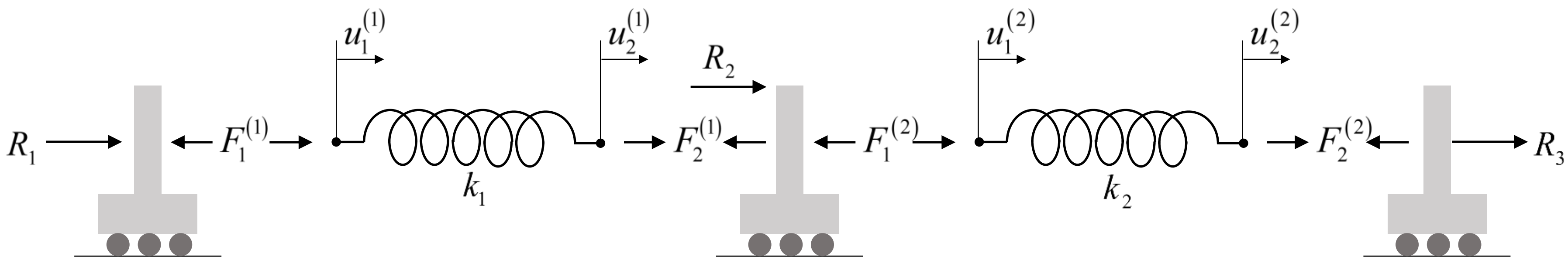
$$\begin{bmatrix} k_1 + k_2 & -k_1 - k_2 \\ -k_1 - k_2 & k_1 + k_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \end{bmatrix}$$

Symmetric : $\mathbf{K} = \mathbf{K}^T$

Two spring system - Ⅱ (Serial connection)



Free body diagram



Compatibility of displacement

$$u_1^{(1)} = U_1$$
$$u_2^{(1)} = u_1^{(2)} = U_2$$
$$u_2^{(2)} = U_3$$

Equilibrium of force

$$F_1^{(1)} = R_1$$
$$F_2^{(1)} + F_1^{(2)} = R_2$$
$$F_2^{(2)} = R_3$$

$$\rightarrow \begin{bmatrix} F_1^{(1)} \\ F_2^{(1)} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ F_1^{(2)} \\ F_2^{(2)} \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}$$

A set of linear equations

$$\begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} F_1^{(1)} \\ F_2^{(1)} \\ 0 \end{bmatrix} \quad \text{for spring 1}$$

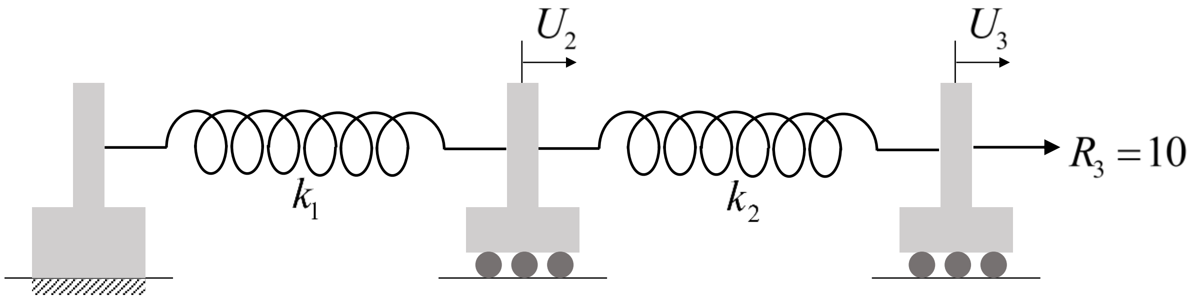
$$\begin{bmatrix} 0 & 0 & 0 \\ 0 & k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} 0 \\ F_1^{(2)} \\ F_2^{(2)} \end{bmatrix} \quad \text{for spring 2}$$

Finally,

$$\begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix}$$

Symmetric & Singular

When $U_1 = 0,\ R_2 = 0,\ R_3 = 10$



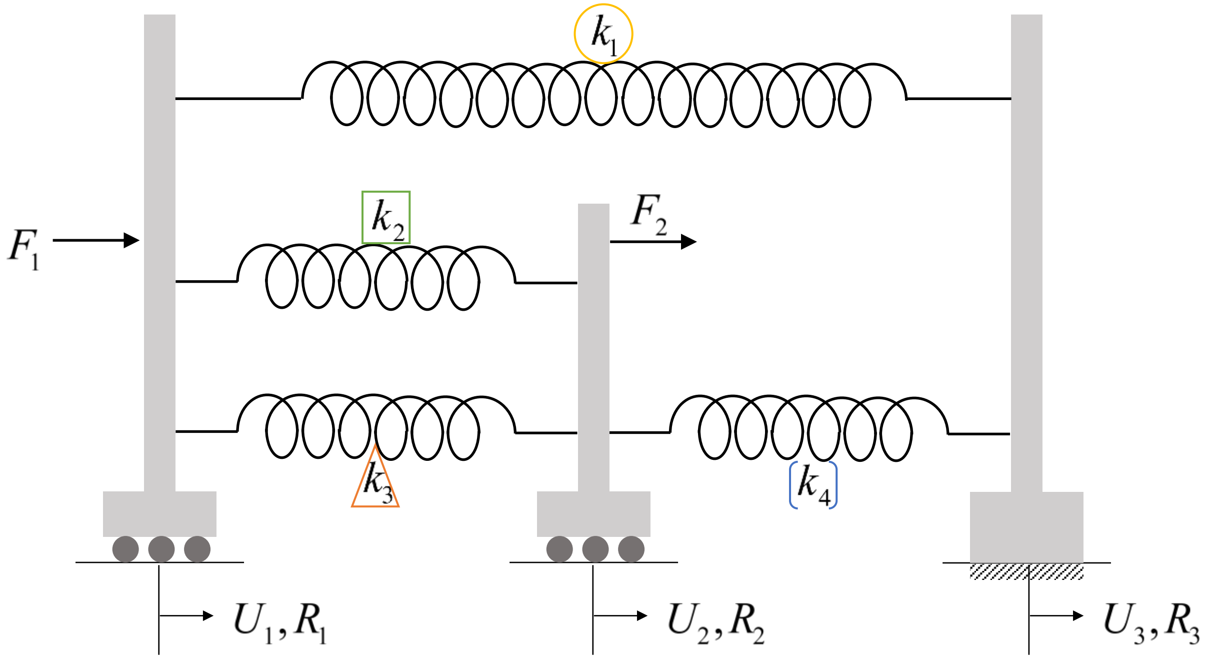
$$\begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1 + k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} \begin{bmatrix} 0 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} R_1 \\ 0 \\ R_3 \end{bmatrix}$$

→ The first row gives the reaction $R_1 = -k_1 U_2$

→ $\begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{bmatrix} U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} 0 \\ R_3 \end{bmatrix}$ → $\begin{bmatrix} U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ R_3 \end{bmatrix}$

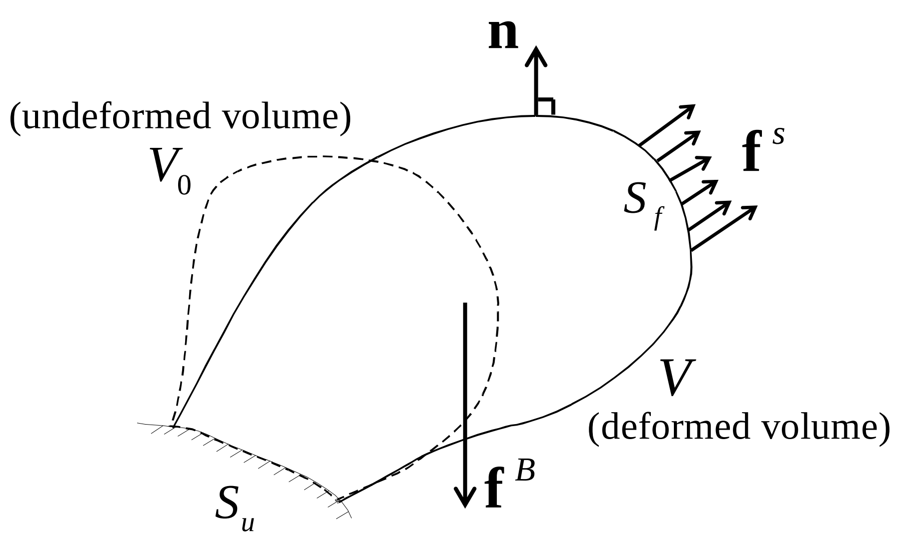
In this study, we found that the equilibrium of the entire system can be obtained from assembly of the equilibrium eqns of individual springs.

Direct stiffness method



$$\begin{bmatrix} k_1 + k_2 + k_3 & -k_2 - k_3 & -k_1 \\ -k_2 - k_3 & k_2 + k_3 + k_4 & -k_4 \\ -k_1 & -k_4 & k_1 + k_4 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \end{bmatrix} \rightarrow \begin{bmatrix} k_1 + k_2 + k_3 & -k_2 - k_3 \\ -k_2 - k_3 & k_2 + k_3 + k_4 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}$$

3. Principle of Virtual Work (PVW)



External surface

S_u : Displacement boundary

S_f : Force boundary

$S = S_u \bigcup S_f$, $S_u \bigcap S_f = \varnothing$

$$\mathbf{f}^s = \begin{bmatrix} f_1^s \\ f_2^s \\ f_3^s \end{bmatrix}:$$ Surface force [force/unit area]

$$\mathbf{f}^B = \begin{bmatrix} f_1^B \\ f_2^B \\ f_3^B \end{bmatrix}:$$ Body force [force/unit volume]

$$\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}:$$ Displacement vector

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{11} & \tau_{12} & \tau_{13} \\ & \tau_{22} & \tau_{23} \\ \text{sym} & & \tau_{33} \end{bmatrix}:$$ Cauchy stress tensor [force/unit deformed area]

$$\boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{11} & \varepsilon_{12} & \varepsilon_{13} \\ & \varepsilon_{22} & \varepsilon_{23} \\ \text{sym} & & \varepsilon_{33} \end{bmatrix}:$$ Small strain tensor, $\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$

The solutions should satisfy:

(1) Equilibrium in V and on S_f

$\tau_{ij,j} + f_i^B = 0$ in V (eq. 3.1)

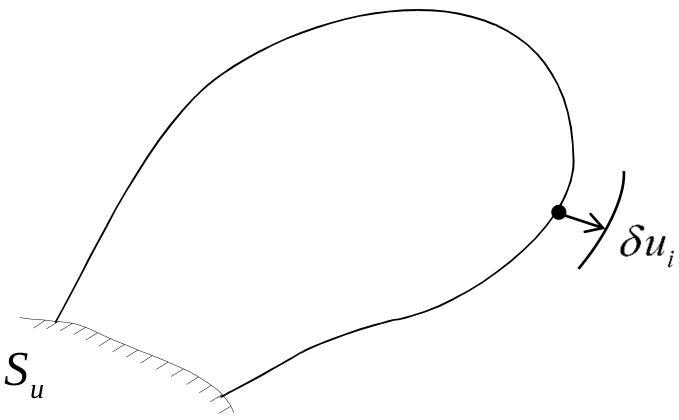
$\tau_{ij} n_j = f_i^s$ on S_f

(2) Compatibility

$u_i = 0$ on S_u

(3) Stress–strain law

$\tau_{ij} = \tau_{ij}\left(\varepsilon_{kl}\right)$



δu_i: virtual displacement → "zero on S_u, arbitrary otherwise, but continuous"

(1) Multiplying δu_i to the equilibrium equation (eq. 3.1)

$$\left(\tau_{ij,j} + f_i^B\right)\delta u_i = 0$$

(2) Integration w.r.t the total volume V ($\approx V_0$ when displacement is small)

$$\int_V \left(\tau_{ij,j} + f_i^B\right)\delta u_i \, dV = 0$$

(3) Integration by parts

$$\int_V (\tau_{ij,j}\delta u_i + f_i^B \delta u_i)dV = 0 \quad \leftarrow \quad \tau_{ij,j}\delta u_i = \left(\tau_{ij}\delta u_i\right)_{,j} - \tau_{ij}\delta u_{i,j}$$

$$\rightarrow \boxed{\int_V \left(\tau_{ij}\delta u_i\right)_{,j} dV} - \boxed{\int_V \left(\tau_{ij}\delta u_{i,j}\right)dV} + \int_V \left(f_i^B \delta u_i\right)dV = 0 \quad \text{(eq. 3.2)}$$

(4) Divergence theorem and small virtual strain

(a) Divergence theorem is used for the first term of (eq. 3.2)

$$\boxed{\int_V \left(\tau_{ij}\delta u_i\right)_{,j} dV} = \int_S \left(\tau_{ij}\delta u_i\right) n_j dS \quad \leftarrow \quad S = S_u \cup S_f$$

$$= \int_{S_f} \left(\tau_{ij} n_j\right)\delta u_i dS + \cancel{\int_{S_u} \left(\tau_{ij} n_j\right)\delta u_i dS} \quad \leftarrow \quad \tau_{ij} n_j = f_i^S \text{ on } S_f \quad \text{and} \quad \delta u_i = 0 \text{ on } S_u$$

$$= \int_{S_f} f_i^S \delta u_i dS$$

(b) Small virtual strain is defined for the second term of (eq. 3.2)

$$\int_V \left(\tau_{ij}\delta u_{i,j}\right)dV = \frac{1}{2}\int_V \tau_{ij}\left(\delta u_{i,j} + \delta u_{j,i}\right)dV \quad \leftarrow \quad \frac{1}{2}\left(\delta u_{i,j} + \delta u_{j,i}\right) = \delta\varepsilon_{ij}$$

: virtual strain tensor (sym.)

$$= \int_V \tau_{ij}\delta\varepsilon_{ij}dV$$

(5) Substituting the results of (a) and (b) into (eq. 3.2), PVW is obtained

$$\int_V \left(\tau_{ij}\delta\varepsilon_{ij}\right)dV = \int_V \left(f_i^B \delta u_i\right)dV + \int_{S_f}\left(f_i^S \delta u_i\right)dS$$

(Real terms: τ_{ij}, f_i^S, f_i^B ←→ Virtual terms: δu_i, $\delta\varepsilon_{ij}$)

- So called "weak form"
- "Internal virtual work" (LHS) is the same as "external virtual work" (RHS).

- Each term of PVW produces “scalar value” (scalar equation).

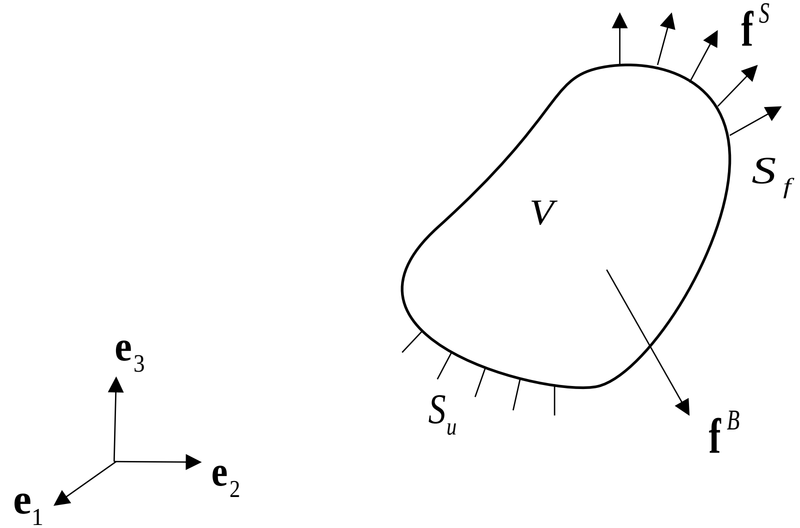
PVW for linear elastic materials

$$\int_V \left(\tau_{ij}\delta\varepsilon_{ij}\right)dV = \int_V \left(f_i^B \delta u_i\right)dV + \int_{S_f} \left(f_i^S \delta u_i\right)dS \qquad \leftarrow \; \tau_{ij} = C_{ijkl}\varepsilon_{kl}$$

$$\boxed{\int_V \delta\varepsilon_{ij} C_{ijkl} \varepsilon_{kl} dV = \int_V f_i^B \delta u_i dV + \int_{S_f} f_i^S \delta u_i dS},$$

Solution procedure with strong form	**Solution procedure with weak form**
Find the solution which satisfies: (1) Equilibrium in V and on S_f (2) Compatibility (3) Stress–Strain law	Find $\mathbf{u}$ such that "Principle of virtual work" is satisfied for all possible $\delta\mathbf{u}$

4. Finite Element Formulation



PVW: $\int_V \tau_{ij}\delta\varepsilon_{ij}dV = \int_V f_i^B \delta u_i dV + \int_{S_f} f_i^S \delta u_i dS$

Let us assume

① Small displacement: $V_o \approx V$

② Linear elastic material: $\tau_{ij} = C_{ijkl}\varepsilon_{kl}$

In the finite element formulation,

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{11} \\ \tau_{22} \\ \tau_{33} \\ \tau_{12} \\ \tau_{23} \\ \tau_{31} \end{bmatrix}, \qquad \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{11} \\ \varepsilon_{22} \\ \varepsilon_{33} \\ \gamma_{12} \\ \gamma_{23} \\ \gamma_{31} \end{bmatrix},$$

where $\gamma_{ij} = 2\varepsilon_{ij} \ (i \neq j)$: engineering shear strain

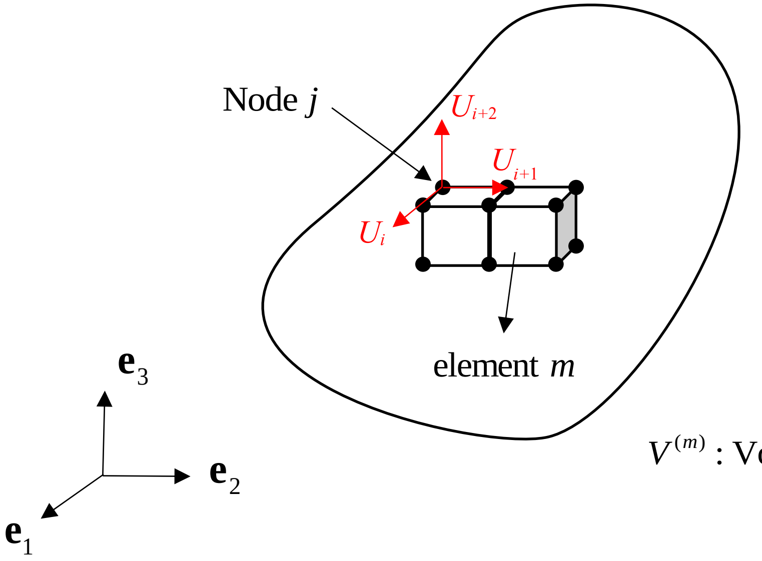
$$\tau_{ij}\varepsilon_{ij} = \sum_{i=1}^{3}\sum_{j=1}^{3}\tau_{ij}\varepsilon_{ij} = \boldsymbol{\varepsilon}^{\mathrm{T}}\boldsymbol{\tau} = \boldsymbol{\varepsilon}\cdot\boldsymbol{\tau}$$ (note: $\tau_{12}\varepsilon_{12} + \tau_{21}\varepsilon_{21} = 2\tau_{12}\varepsilon_{12} = \tau_{12}\gamma_{12}$)

Material Law: $\boldsymbol{\tau} = \mathbf{C}\boldsymbol{\varepsilon}$

PVW in vector/matrix form:

$$\boxed{\int_V \delta\boldsymbol{\varepsilon}^{\mathrm{T}}\mathbf{C}\boldsymbol{\varepsilon}\,dV = \int_V \delta\mathbf{u}^{\mathrm{T}}\mathbf{f}^{B}\,dV + \int_{S_f} \delta\mathbf{u}^{\mathrm{T}}\mathbf{f}^{S}\,dS}$$

Finite Element Discretization



$V^{(m)}$: Volume of element m, $V \approx \sum_m V^{(m)}$

Let us assume that $S_f = S$ and $S_u = \phi$ → "PVW still works."

Nodal displacement at node j → $\begin{bmatrix} U_i \\ U_{i+1} \\ U_{i+2} \end{bmatrix}$

Nodal displacement vector (nodal DOFs vector),

$$\mathbf{U} = \begin{bmatrix} U_1 \\ \vdots \\ U_i \\ U_{i+1} \\ \vdots \\ U_N \end{bmatrix}, \qquad N: \text{number of the total DOFs}$$

PVW: $$\sum_m \int_{V^{(m)}} \delta \boldsymbol{\varepsilon}^{(m)^{\mathrm{T}}} \mathbf{C}^{(m)} \boldsymbol{\varepsilon}^{(m)} dV^{(m)} = \sum_m \int_{V^{(m)}} \delta \mathbf{u}^{(m)^{\mathrm{T}}} \mathbf{f}^{B(m)} dV^{(m)} + \sum_m \int_{S_1^{(m)} \cdots S_q^{(m)}} \delta \mathbf{u}^{s(m)^{\mathrm{T}}} \mathbf{f}^{S(m)} dS^{(m)},$$

where $S_1^{(m)} \cdots S_q^{(m)}$ are surfaces of “element m” on boundary.

Displacement Interpolations

$\mathbf{u}^{(m)} = \mathbf{H}^{(m)}\mathbf{U}$ (interpolation of displacement)

$\delta\mathbf{u}^{(m)} = \mathbf{H}^{(m)}\delta\mathbf{U}$ (interpolation of virtual displacement)

where

$\mathbf{u}^{(m)}$: displacement field of element m

$\delta\mathbf{u}^{(m)}$: virtual displacement field of element m

$\mathbf{U}$: nodal displacement vector

$\delta\mathbf{U}$: virtual nodal displacement vector

$\mathbf{H}^{(m)}$: displacement interpolation matrix for element m

(Note) The same interpolation is used for real and virtual displacements.

→ "symmetric stiffness matrix"

From the strain-displacement relation,

$$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \qquad \rightarrow \qquad \boldsymbol{\varepsilon}^{(m)} = \mathbf{B}^{(m)}\mathbf{U}$$

$$\delta\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial \delta u_i}{\partial x_j} + \frac{\partial \delta u_j}{\partial x_i}\right) \qquad \rightarrow \qquad \delta\boldsymbol{\varepsilon}^{(m)} = \mathbf{B}^{(m)}\delta\mathbf{U}$$

$\boldsymbol{\varepsilon}^{(m)}$: strain field of element m

$\delta\boldsymbol{\varepsilon}^{(m)}$: virtual strain field of element m

$\mathbf{B}^{(m)}$: strain interpolation matrix for element m

Using the displacement and strain interpolations in PVW, the following equation is obtained

$$\delta\mathbf{U}^{\mathrm{T}}\left[\sum_m \int_{V^{(m)}} \mathbf{B}^{(m)^{\mathrm{T}}}\mathbf{C}^{(m)}\mathbf{B}^{(m)} dV^{(m)}\right]\mathbf{U} = \delta\mathbf{U}^{\mathrm{T}}\left[\sum_m \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}}\mathbf{f}^{B(m)} dV^{(m)} + \sum_m \int_{S_1^{(m)}\cdots S_q^{(m)}} \mathbf{H}_s^{(m)^{\mathrm{T}}}\mathbf{f}^{S(m)} dS^{(m)}\right]$$

(eq. 3.1)

Substituting $\delta \mathbf{U} = [1\ 0\ \cdots\ 0]^{\mathrm{T}}$ ($\delta U_1 = 1,\ \text{others} = 0$) into (eq. 3.1), we obtain a linear equation

$$K_{11}U_1 + K_{12}U_2 + \cdots + K_{1N}U_N = R_1$$

Substituting $\delta \mathbf{U} = [0\ 1\ 0\ \cdots\ 0]^{\mathrm{T}}$ ($\delta U_2 = 1,\ \text{others} = 0$) into (eq. 3.1), we obtain another equation

$$K_{21}U_1 + K_{22}U_2 + \cdots + K_{2N}U_N = R_2$$

We can do this task N times.

This process to apply “virtual displacement vectors” is the same to

$$\cancel{\mathbf{I}^{\mathrm{T}}}\left[\sum_m \int_{V^{(m)}} \mathbf{B}^{(m)^{\mathrm{T}}} \mathbf{C}^{(m)} \mathbf{B}^{(m)} dV^{(m)}\right]\mathbf{U} = \cancel{\mathbf{I}^{\mathrm{T}}}\left[\sum_m \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \mathbf{f}^{B(m)} dV^{(m)} + \sum_m \int_{S_1^{(m)} \cdots S_q^{(m)}} \mathbf{H}_S^{(m)^{\mathrm{T}}} \mathbf{f}^{S(m)} dS^{(m)}\right]$$

Finally, we obtain a set of N linear equations. → "static equilibrium equations"

$$\boxed{\mathbf{KU} = \mathbf{R}},$$

where $\mathbf{K} = \sum_m \mathbf{K}^{(m)}$ with $\mathbf{K}^{(m)} = \int_{V^{(m)}} \mathbf{B}^{(m)^{\mathrm{T}}} \mathbf{C}^{(m)} \mathbf{B}^{(m)} dV^{(m)}$

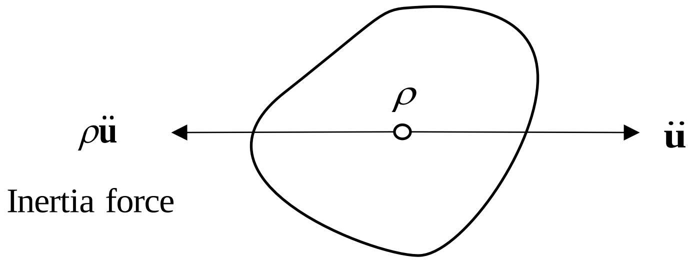
$$\mathbf{R} = \mathbf{R}_B + \mathbf{R}_S \qquad \begin{cases} \mathbf{R}_B = \sum_m \mathbf{R}_B^{(m)} \quad \text{with } \mathbf{R}_B^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \mathbf{f}^{B(m)} dV^{(m)} \\ \mathbf{R}_S = \sum_m \mathbf{R}_s^{(m)} \quad \text{with } \mathbf{R}_s^{(m)} = \int_{S_1^{(m)} \cdots S_q^{(m)}} \mathbf{H}_s^{(m)^{\mathrm{T}}} \mathbf{f}^{s(m)} dS^{(m)} \end{cases}$$

$\mathbf{K}$: Stiffness matrix

$\mathbf{U}$: Nodal displacement vector

$\mathbf{R}$: Nodal force vector

Dynamic Equilibrium Equations (Equations of motion)



$$\mathbf{R}^{B(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \mathbf{f}^{B(m)} dV^{(m)}$$

$$\mathbf{f}^{B(m)} = \overline{\mathbf{f}}^{B(m)} + (-\rho \ddot{\mathbf{u}}^{(m)}) \quad \leftarrow \quad \mathbf{u}^{(m)} = \mathbf{H}^{(m)}\mathbf{U} \text{ and } \ddot{\mathbf{u}}^{(m)} = \mathbf{H}^{(m)}\ddot{\mathbf{U}}$$

$$\mathbf{R}^{B(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \overline{\mathbf{f}}^{B(m)} dV^{(m)} - \left[\int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \rho^{(m)} \mathbf{H}^{(m)} dV^{(m)} \right] \ddot{\mathbf{U}}$$

$\mathbf{KU} = \mathbf{R} - \mathbf{M}\ddot{\mathbf{U}}$ with "mass matrix" $\mathbf{M} = \sum_m \mathbf{M}^{(m)}$, $\mathbf{M}^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \rho^{(m)} \mathbf{H}^{(m)} dV^{(m)}$

$$\boxed{\mathbf{M}\ddot{\mathbf{U}} + \mathbf{KU} = \mathbf{R}}$$

Imposition of Zero-Displacement BC

Note that $\mathbf{K}$ is singular because $S_u = \phi$ is assumed.

The displacement BC is imposed by simply getting rid of the columns corresponding to zero-displacements ($U_i = 0$) and the rows corresponding to zero-virtual displacements ($\delta U_i = 0$) in the stiffness matrix $\mathbf{K}$.

Ex) When $U_3 = U_4 = 0$,

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & K_{15} \\ & K_{22} & K_{23} & K_{24} & K_{25} \\ & & K_{33} & K_{34} & K_{35} \\ & sym. & & K_{44} & K_{45} \\ & & & & K_{55} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} \rightarrow \begin{bmatrix} K_{11} & K_{12} & K_{15} \\ & K_{22} & K_{25} \\ sym. & & K_{55} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_5 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_5 \end{bmatrix}$$

Then, $\mathbf{K}_{N \times N}$ is reduced to $\tilde{\mathbf{K}}_{\tilde{N} \times \tilde{N}}$ ($\tilde{\mathbf{N}} = \mathbf{N} -$(# of prescribed DOFs)) and the displacement and force vectors are also reduced into $\tilde{\mathbf{U}}$ and $\tilde{\mathbf{R}}$. Finally, we get $\tilde{\mathbf{K}}\tilde{\mathbf{U}} = \tilde{\mathbf{R}}$. When the displacement BC is properly applied, the equilibrium equation can be solved.

Imposition of General Displacement BC

In general, if $U_k = U_k^*$ is specified, we have

$$\bar{R}_k = K_{kk} U_k^*,$$

$$\bar{R}_i = R_i - K_{ik} U_k^*, \quad K_{ik} = K_{tk} = 0 \qquad \text{with } i = 1, 2, \cdots, k-1, k+1, \cdots, n \ \ (i \neq k).$$

This procedure is repeated for every specified displacement.

Ex) When $U_3 = U_3^*$ is specified,

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & K_{15} \\ K_{21} & K_{22} & K_{23} & K_{24} & K_{25} \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} \\ K_{41} & K_{42} & K_{43} & K_{44} & K_{45} \\ K_{51} & K_{52} & K_{53} & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{bmatrix} = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \end{bmatrix} \rightarrow \begin{bmatrix} K_{11} & K_{12} & 0 & K_{14} & K_{15} \\ K_{21} & K_{22} & 0 & K_{24} & K_{25} \\ 0 & 0 & K_{33} & 0 & 0 \\ K_{41} & K_{42} & 0 & K_{44} & K_{45} \\ K_{51} & K_{52} & 0 & K_{54} & K_{55} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{bmatrix} = \begin{bmatrix} \bar{R}_1 \\ \bar{R}_2 \\ K_{33} U_3^* \\ \bar{R}_4 \\ \bar{R}_5 \end{bmatrix}$$

Then, $\mathbf{K}_{N \times N}$ is modified to $\bar{\mathbf{K}}_{N \times N}$ and the displacement and force vectors are also modified into $\bar{\mathbf{U}}$ and $\bar{\mathbf{R}}$. Finally, we get $\bar{\mathbf{K}}\bar{\mathbf{U}} = \bar{\mathbf{R}}$.

Properties of Stiffness Matrix

① The i^th^ column of $\mathbf{K}$ represents the forces needed at the nodes to impose the unit displacement corresponding to the i^th^ displacement DOF ($U_i = 1$) and keep all other DOFs equal to zero ($U_j = 0,\ i \neq j$).

② $\mathbf{K}$ is a symmetric matrix, i.e. $\mathbf{K}^{\mathrm{T}} = \mathbf{K}$ or $K_{ij} = K_{ji}$.

"Betti's reciprocal theorem" and "Maxwell's theorem"

③ $\mathbf{KU} = \mathbf{R}$

External work $= \frac{1}{2}\mathbf{U}^{\mathrm{T}}\mathbf{R} = \frac{1}{2}\mathbf{U}^{\mathrm{T}}\mathbf{KU} \geq 0$ (strain energy stored)

When $\mathbf{U} = \mathbf{0}$ or $\mathbf{U}$ is the displacement vector corresponding to rigid body motions, $\frac{1}{2}\mathbf{U}^{\mathrm{T}}\mathbf{KU} = 0$.

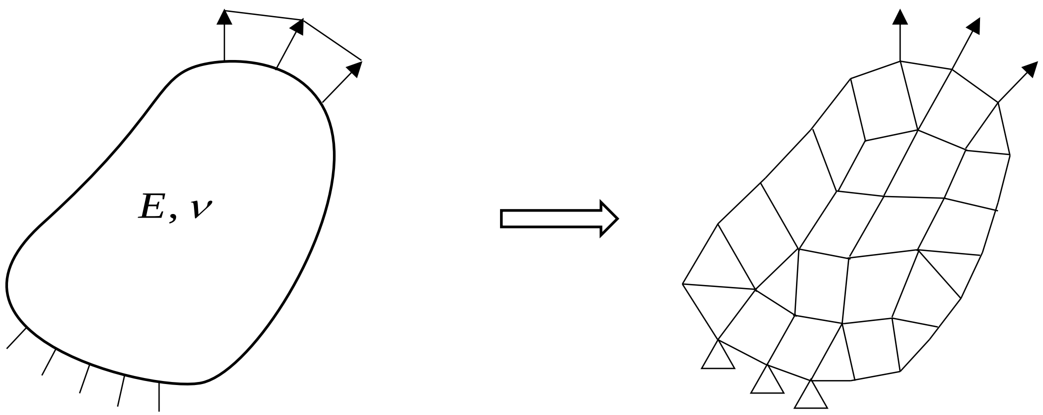
When $\mathbf{U} \neq \mathbf{0}$, $\frac{1}{2}\tilde{\mathbf{U}}^{\mathrm{T}}\tilde{\mathbf{K}}\tilde{\mathbf{U}} > 0$ for all $\tilde{\mathbf{U}}$. → $\tilde{\mathbf{K}}$: "positive definite" matrix

FE Solution Procedure

Principal unknown: $\mathbf{U}$

Step 1) Geometry, material properties, applied load and displacement BC are given.

→ Construct "FE model".



FE model

FE model has information on

- nodal positions
- element connectivity (a set of nodes to construct the element)
- material properties (E, ν)
- BCs (force & displacement) are only applied at nodes

Step 2) Calculate $\mathbf{K}^{(m)}$ and $\mathbf{R}^{(m)}$ of each finite element (element matrix)

Step 3) Assemble $\mathbf{K}$ and $\mathbf{R}$ (total matrix).

$$\mathbf{K} = \sum_m \mathbf{K}^{(m)}, \quad \mathbf{R} = \sum_m \mathbf{R}^{(m)}$$

Step 4) Apply the displacement BC.

$$\mathbf{K} \rightarrow \tilde{\mathbf{K}}$$

Step 5) Solve the linear system.

$\tilde{\mathbf{K}}\tilde{\mathbf{U}} = \tilde{\mathbf{R}} \rightarrow \tilde{\mathbf{U}} = \tilde{\mathbf{K}}^{-1}\tilde{\mathbf{R}}$ → $\mathbf{U}$ is found.

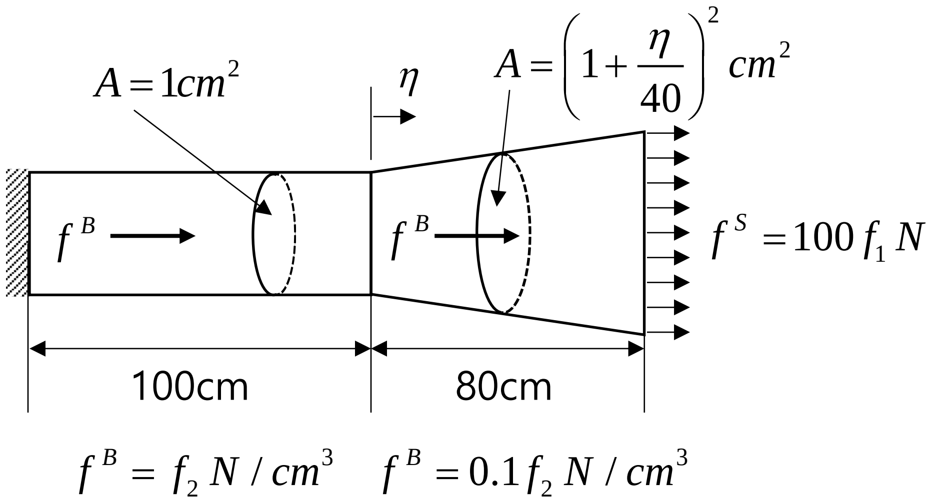
Step 6) Calculate solutions

→ Displacement field of element m: $\mathbf{u}^{(m)} = \mathbf{H}^{(m)}\mathbf{U}$

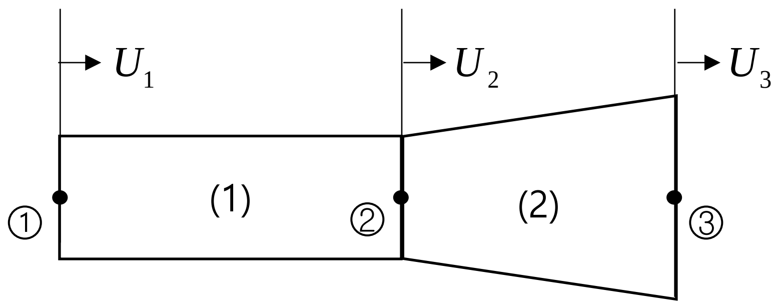
→ Strain field of element m: $\boldsymbol{\varepsilon}^{(m)} = \mathbf{B}^{(m)}\mathbf{U}$

→ Stress field of element m: $\boldsymbol{\tau}^{(m)} = \mathbf{C}^{(m)}\boldsymbol{\varepsilon}^{(m)} = \mathbf{C}^{(m)}\mathbf{B}^{(m)}\mathbf{U}$

Example – 1D bar problem

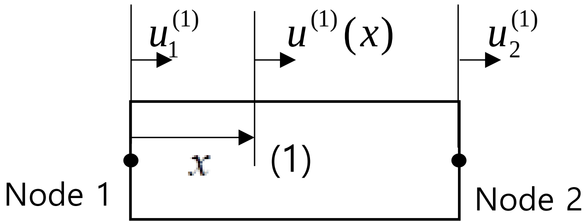


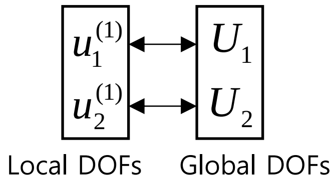
FE model



$$\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}$$

Element (1)





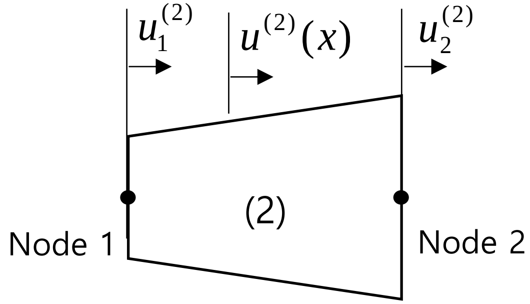
x (locally defined coordinate)

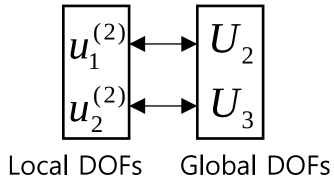
$$u^{(1)}(x) = \left(1 - \frac{x}{100}\right) u_1^{(1)} + \frac{x}{100} u_2^{(1)}$$

$$u^{(1)}(x) = \begin{bmatrix} 1 - \dfrac{x}{100} & \dfrac{x}{100} \end{bmatrix} \begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \end{bmatrix} = \begin{bmatrix} 1 - \dfrac{x}{100} & \dfrac{x}{100} & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \mathbf{H}^{(1)} \mathbf{U}$$

$$\varepsilon_{xx}^{(1)} = \frac{\partial u^{(1)}}{\partial x} = \begin{bmatrix} -\dfrac{1}{100} & \dfrac{1}{100} \end{bmatrix} \begin{bmatrix} u_1^{(1)} \\ u_2^{(1)} \end{bmatrix} = \begin{bmatrix} -\dfrac{1}{100} & \dfrac{1}{100} & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \mathbf{B}^{(1)} \mathbf{U}$$

Element (2)





$$u^{(2)}(x) = \left(1 - \frac{x}{80}\right) u_1^{(2)} + \frac{x}{80} u_2^{(2)}$$

$$u^{(2)}(x) = \begin{bmatrix} 1 - \frac{x}{80} & \frac{x}{80} \end{bmatrix} \begin{bmatrix} u_1^{(2)} \\ u_2^{(2)} \end{bmatrix} = \begin{bmatrix} 0 & 1 - \frac{x}{80} & \frac{x}{80} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \mathbf{H}^{(2)} \mathbf{U}$$

$$\varepsilon_{xx}^{(2)} = \frac{\partial u^{(2)}}{\partial x} = \begin{bmatrix} -\frac{1}{80} & \frac{1}{80} \end{bmatrix} \begin{bmatrix} u_1^{(2)} \\ u_2^{(2)} \end{bmatrix} = \begin{bmatrix} 0 & -\frac{1}{80} & \frac{1}{80} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \mathbf{B}^{(2)} \mathbf{U}$$

Stiffness matrix

$$\mathbf{KU} = \mathbf{R} \quad \text{with } \mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}$$

$$\mathbf{K} = \sum_{m=1}^{2} \mathbf{K}^{(m)} = \mathbf{K}^{(1)} + \mathbf{K}^{(2)}$$

$$= \int_{V^{(1)}} \mathbf{B}^{(1)^T} E \mathbf{B}^{(1)} dV + \int_{V^{(2)}} \mathbf{B}^{(2)^T} E \mathbf{B}^{(2)} dV^{(2)}$$

$$\mathbf{K} = \int_0^{100} 1 \times \begin{bmatrix} -\frac{1}{100} \\ \frac{1}{100} \\ 0 \end{bmatrix} E \begin{bmatrix} -\frac{1}{100} & \frac{1}{100} & 0 \end{bmatrix} dx + \int_0^{80} \left(1 + \frac{x}{40}\right)^2 \begin{bmatrix} 0 \\ -\frac{1}{80} \\ \frac{1}{80} \end{bmatrix} E \begin{bmatrix} 0 & -\frac{1}{80} & \frac{1}{80} \end{bmatrix} dx$$

$$= \frac{E}{100}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \frac{13E}{240}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix} = \frac{E}{240}\begin{bmatrix} 2.4 & -2.4 & 0 \\ -2.4 & 15.4 & -13 \\ 0 & -13 & 13 \end{bmatrix}$$

Direct stiffness method

$$\mathbf{K}^{(1)} = \int_0^{100} 1 \times \begin{bmatrix} -\dfrac{1}{100} \\ \dfrac{1}{100} \end{bmatrix} E \begin{bmatrix} -\dfrac{1}{100} & \dfrac{1}{100} \end{bmatrix} dx = \frac{E}{100} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$\mathbf{K}^{(2)} = \int_0^{80} \left(1 + \frac{x}{40}\right)^2 \times \begin{bmatrix} -\dfrac{1}{80} \\ \dfrac{1}{80} \end{bmatrix} E \begin{bmatrix} -\dfrac{1}{80} & \dfrac{1}{80} \end{bmatrix} dx = \frac{13E}{240} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$\mathbf{K} = \begin{bmatrix} \dfrac{E}{100} & -\dfrac{E}{100} & 0 \\ -\dfrac{E}{100} & \dfrac{E}{100} + \dfrac{13E}{240} & \dfrac{-13E}{240} \\ 0 & \dfrac{-13E}{240} & \dfrac{13E}{240} \end{bmatrix} = \frac{E}{240} \begin{bmatrix} 2.4 & -2.4 & 0 \\ -2.4 & 15.4 & -13 \\ 0 & -13 & 13 \end{bmatrix}$$

Load vector

$$\mathbf{R} = \mathbf{R}_B + \mathbf{R}_S$$

$$\mathbf{R}_B = \sum_m \mathbf{R}_B^{(m)} \quad \text{with} \quad \mathbf{R}_B^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)T} \mathbf{f}^{(m)} dV^{(m)}$$

$$\mathbf{R}_S = \sum_m \mathbf{R}_S^{(m)} \quad \text{with} \quad \mathbf{R}_S^{(m)} = \int_{S_f^{(m)}} \mathbf{H}_S^{(m)T} \mathbf{f}^{S(m)} dS^{(m)}$$

$$\mathbf{R}_B = \int_0^{100} 1 \times \begin{bmatrix} 1 - \frac{x}{100} \\ \frac{x}{100} \\ 0 \end{bmatrix} f_2 \, dx + \int_0^{80} \left(1 + \frac{x}{40}\right)^2 \begin{bmatrix} 0 \\ 1 - \frac{x}{80} \\ \frac{x}{80} \end{bmatrix} 0.1 f_2 \, dx = \frac{1}{3} \begin{bmatrix} 150 \\ 186 \\ 68 \end{bmatrix} f_2 \qquad \text{(Body force)}$$

$$\mathbf{R}_S = \mathbf{R}_S^{(1)} + \mathbf{R}_S^{(2)} = \int_{S_2} \mathbf{H}^{(2)T}\Big|_{x=80} \frac{100 f_1}{S_2} dS = \int_{S_2} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \frac{100 f_1}{S_2} dS = \begin{bmatrix} 0 \\ 0 \\ 100 f_1 \end{bmatrix}$$ (Surface force)

$$\mathbf{R} = \mathbf{R}_B + \mathbf{R}_S = \begin{bmatrix} 50 f_2 \\ 62 f_2 \\ \frac{68}{3} f_2 + 100 f_1 \end{bmatrix}$$

Equilibrium equation

$$\frac{E}{240} \begin{bmatrix} 2.4 & -2.4 & 0 \\ -2.4 & 15.4 & -13 \\ 0 & -13 & 13 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} 50 f_2 \\ 62 f_2 \\ \frac{68}{3} f_2 + 100 f_1 \end{bmatrix}$$ ($\mathbf{KU} = \mathbf{R}$, $\mathbf{K}$ is singular.)

Imposition of displacement BC

$U_1 = 0$ (and $\delta U_1 = 0$)

$$\frac{E}{240}\begin{bmatrix} 15.4 & -13 \\ -13 & 13 \end{bmatrix}\begin{bmatrix} U_2 \\ U_3 \end{bmatrix} = \begin{bmatrix} 62 f_2 \\ \frac{68}{3} f_2 + 100 f_1 \end{bmatrix} \qquad (\tilde{\mathbf{K}}\tilde{\mathbf{U}} = \tilde{\mathbf{R}})$$

Nodal displacement vector is found: $\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \end{bmatrix}$

→ Displacement field: $\mathbf{u}^{(m)} = \mathbf{H}^{(m)}\mathbf{U}$

→ Strain field: $\varepsilon_{xx}^{(m)} = \mathbf{B}^{(m)}\mathbf{U}$

→ Stress field: $\tau_{xx}^{(m)} = E\mathbf{B}^{(m)}\mathbf{U}$

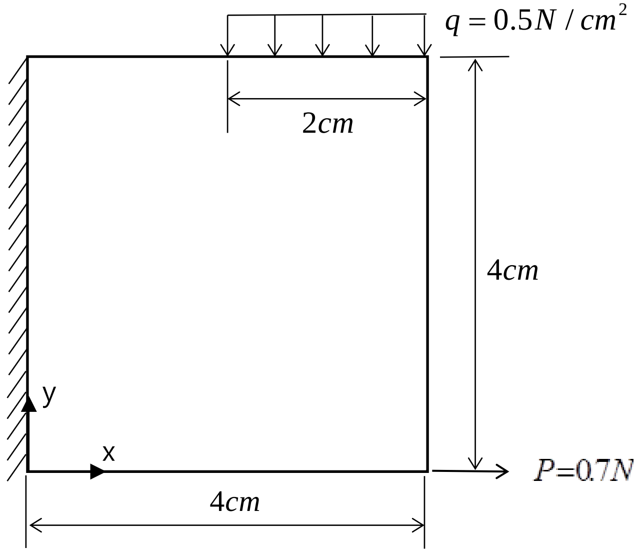
Mass matrix for dynamic analysis

$$\mathbf{M} = \sum_{m=1}^{2} \int_{V^{(m)}} \mathbf{H}^{T(m)} \rho^{(m)} \mathbf{H}^{(m)} \, dV^{(m)}$$

$$\mathbf{M} = \int_0^{100} 1 \times \begin{bmatrix} 1-\dfrac{x}{100} \\ \dfrac{x}{100} \\ 0 \end{bmatrix} \rho \begin{bmatrix} 1-\dfrac{x}{100} & \dfrac{x}{100} & 0 \end{bmatrix} dx + \int_0^{80} \left(1+\frac{x}{40}\right)^2 \begin{bmatrix} 0 \\ 1-\dfrac{x}{80} \\ \dfrac{x}{80} \end{bmatrix} \rho \begin{bmatrix} 0 & 1-\dfrac{x}{80} & \dfrac{x}{80} \end{bmatrix} dx$$

$$\mathbf{M} = \frac{\rho}{6} \begin{bmatrix} 200 & 100 & 0 \\ & 584 & 336 \\ \text{sym.} & & 1024 \end{bmatrix}$$

Example – 2D plane stress problem

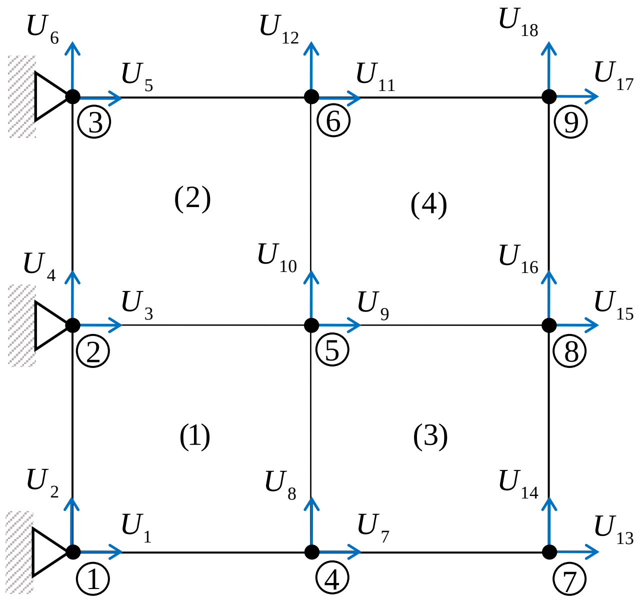


Thickness =1 , E : Young's modulus, υ : Poisson's ratio

Plane stress condition

$$\boldsymbol{\tau} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}\boldsymbol{\varepsilon} \qquad \text{with} \quad \boldsymbol{\tau} = \begin{bmatrix} \tau_{xx} \\ \tau_{yy} \\ \tau_{xy} \end{bmatrix}, \ \boldsymbol{\varepsilon} = \begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix}.$$

Finite element model



$$\mathbf{U} = \begin{bmatrix} U_1 \\ U_2 \\ \vdots \\ U_{18} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} R_1 \\ R_2 \\ \vdots \\ R_{18} \end{bmatrix}$$

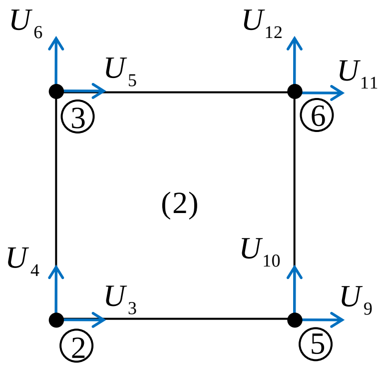
Number of nodes : 9

Number of elements : 4

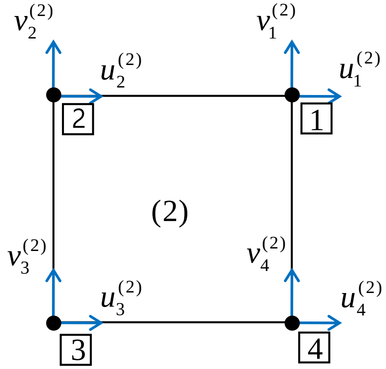
Number of total DOFs : 18 (9x2)

Displacement BC : $U_1 = U_2 = U_3 = U_4 = U_5 = U_6 = 0.$

Element stiffness matrices, $\mathbf{K}^{(m)}$



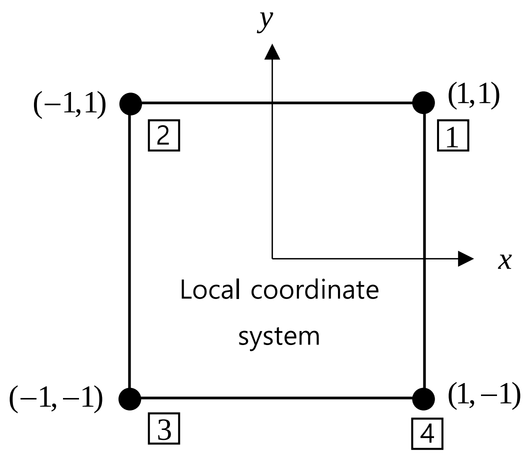
<Global DOFs>



<Local DOFs>

$$\mathbf{u}^{(2)} = \begin{bmatrix} u_1^{(2)} \\ u_2^{(2)} \\ u_3^{(2)} \\ u_4^{(2)} \\ v_1^{(2)} \\ v_2^{(2)} \\ v_3^{(2)} \\ v_4^{(2)} \end{bmatrix} \begin{matrix} \leftrightarrow U_{11} \\ \leftrightarrow U_5 \\ \leftrightarrow U_3 \\ \leftrightarrow U_9 \\ \leftrightarrow U_{12} \\ \leftrightarrow U_6 \\ \leftrightarrow U_4 \\ \leftrightarrow U_{10} \end{matrix}$$

$\mathbf{u}^{(2)}$: Nodal displacement vector of element (2)



Displacement interpolation:

$$\mathbf{u}(x,y)=\begin{bmatrix}u^{(2)}(x,y)\\ v^{(2)}(x,y)\end{bmatrix}$$

$$u^{(2)}(x,y)=\alpha_0+\alpha_1 x+\alpha_2 y+\alpha_3 xy$$

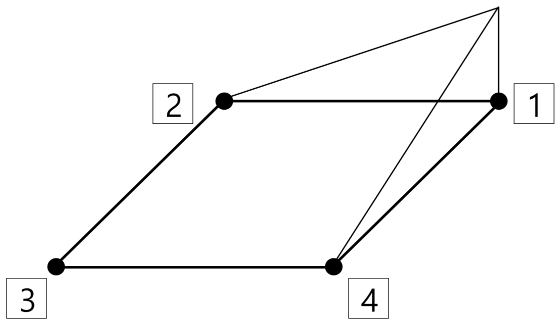
$$u^{(2)}(1,1)=u_1^{(2)},\quad u^{(2)}(-1,1)=u_2^{(2)},\quad u^{(2)}(-1,-1)=u_3^{(2)},\quad u^{(2)}(1,-1)=u_4^{(2)}$$

$$u^{(2)}(x,y)=\sum_{i}^{4} h_i(x,y)u_i^{(2)} = h_1(x,y)u_1^{(2)} + h_2(x,y)u_2^{(2)} + h_3(x,y)u_3^{(2)} + h_4(x,y)u_4^{(2)}$$

with "shape functions"

$$h_1 = \frac{1}{4}(1+x)(1+y),\ \ h_2 = \frac{1}{4}(1-x)(1+y),\ \ h_3 = \frac{1}{4}(1-x)(1-y),\ \ h_4 = \frac{1}{4}(1+x)(1-y)$$

(Note) $h_i = 1$ at node i, and $h_i = 0$ at other nodes.



$$v^{(2)} = \sum_{i}^{4} h_i(x,y)v_i^{(2)} = h_1 v_1^{(2)} + h_2 v_2^{(2)} + h_3 v_3^{(2)} + h_4 v_4^{(2)}$$

$$\begin{bmatrix} u^{(2)} \\ v^{(2)} \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 & h_4 \end{bmatrix} \begin{bmatrix} u_1^{(2)} \\ u_2^{(2)} \\ u_3^{(2)} \\ u_4^{(2)} \\ v_1^{(2)} \\ v_2^{(2)} \\ v_3^{(2)} \\ v_4^{(2)} \end{bmatrix} = \mathbf{H}^{(2)}\mathbf{u}^{(2)}$$

$$\varepsilon_{xx}^{(2)} = \frac{\partial u^{(2)}}{\partial x} = \begin{bmatrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_2}{\partial x} & \frac{\partial h_3}{\partial x} & \frac{\partial h_4}{\partial x} & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{u}^{(2)}$$

$$\varepsilon_{yy}^{(2)} = \frac{\partial v^{(2)}}{\partial y} = \begin{bmatrix} 0 & 0 & 0 & 0 & \frac{\partial h_1}{\partial y} & \frac{\partial h_2}{\partial y} & \frac{\partial h_3}{\partial y} & \frac{\partial h_4}{\partial y} \end{bmatrix} \mathbf{u}^{(2)}$$

$$\gamma_{xy}^{(2)} = \frac{\partial v^{(2)}}{\partial x} + \frac{\partial u^{(2)}}{\partial y} = \begin{bmatrix} \frac{\partial h_1}{\partial y} & \frac{\partial h_2}{\partial y} & \frac{\partial h_3}{\partial y} & \frac{\partial h_4}{\partial y} & \frac{\partial h_1}{\partial x} & \frac{\partial h_2}{\partial x} & \frac{\partial h_3}{\partial x} & \frac{\partial h_4}{\partial x} \end{bmatrix} \mathbf{u}^{(2)}$$

$$\boldsymbol{\varepsilon}^{(2)} = \begin{bmatrix} \varepsilon_{xx}^{(2)} \\ \varepsilon_{yy}^{(2)} \\ \gamma_{xy}^{(2)} \end{bmatrix} = \begin{bmatrix} \frac{\partial h_1}{\partial x} & \frac{\partial h_2}{\partial x} & \frac{\partial h_3}{\partial x} & \frac{\partial h_4}{\partial x} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\partial h_1}{\partial y} & \frac{\partial h_2}{\partial y} & \frac{\partial h_3}{\partial y} & \frac{\partial h_4}{\partial y} \\ \frac{\partial h_1}{\partial y} & \frac{\partial h_2}{\partial y} & \frac{\partial h_3}{\partial y} & \frac{\partial h_4}{\partial y} & \frac{\partial h_1}{\partial x} & \frac{\partial h_2}{\partial x} & \frac{\partial h_3}{\partial x} & \frac{\partial h_4}{\partial x} \end{bmatrix} \begin{bmatrix} u_1^{(2)} \\ \vdots \\ u_4^{(2)} \\ v_1^{(2)} \\ \vdots \\ v_4^{(2)} \end{bmatrix} = \mathbf{B}^{(2)}\mathbf{u}^{(2)}$$

$$\mathbf{K}^{(2)} = \int_{V^{(2)}} \mathbf{B}^{(2)T}\mathbf{C}^{(2)}\mathbf{B}^{(2)} dV^{(2)} = t\int_{-1}^{1}\int_{-1}^{1} \mathbf{B}^{(2)T}(x,y)\mathbf{C}^{(2)}\mathbf{B}^{(2)}(x,y)dxdy$$

with $\mathbf{C}^{(2)} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$

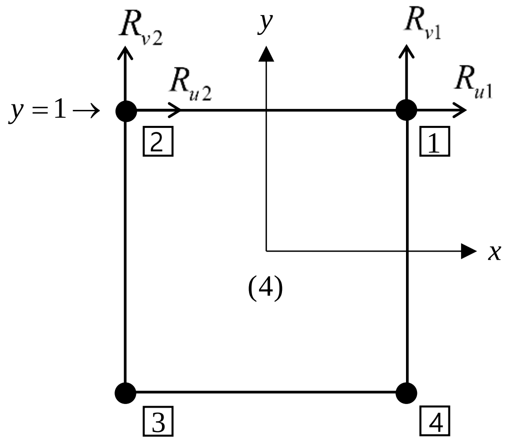
$$\mathbf{K}^{(2)} = \mathbf{K}^{(1)} = \mathbf{K}^{(3)} = \mathbf{K}^{(4)}$$

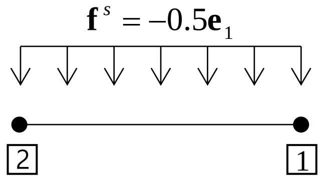
Stiffness matrix, $\mathbf{K}$: $\underset{(8\times 8)}{\mathbf{K}^{(1)}},\ \underset{(8\times 8)}{\mathbf{K}^{(2)}},\ \underset{(8\times 8)}{\mathbf{K}^{(3)}},\ \underset{(8\times 8)}{\mathbf{K}^{(4)}} \rightarrow \underset{(18\times 18)}{\mathbf{K}}$

$$\mathbf{K}_{8\times 8}^{(2)} = \begin{array}{c} \\ \delta u_1^{(2)} \\ \delta u_2^{(2)} \\ \vdots \end{array} \begin{array}{c} \begin{array}{cc} u_1^{(2)} & u_2^{(2)} \end{array} \\ \begin{bmatrix} K_{11}^{(2)} & K_{12}^{(2)} & \cdots \\ K_{21}^{(2)} & K_{22}^{(2)} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \end{array}$$

$$\mathbf{K}_{18\times 18} = \begin{array}{c} \\ \delta U_1 \\ \delta U_2 \\ \vdots \\ \delta U_{11} \\ \vdots \\ \delta U_{18} \end{array} \begin{array}{c} \begin{array}{ccccccc} U_1 & U_2 & \cdots & U_5 & \cdots & U_{11} & \cdots & U_{18} \end{array} \\ \begin{bmatrix} & & & | & & | & & \\ & & & | & & | & & \\ & & & | & & | & & \\ - & - & - & K_{12}^{(2)} & - & K_{11}^{(2)} & - & - & - \\ & & & | & & | & & \\ & & & | & & | & & \end{bmatrix} \end{array}$$

Load vector: $\mathbf{R}$





$$\mathbf{R}=\mathbf{R}_B+\mathbf{R}_S$$

$$\mathbf{R}_B=\sum_m \mathbf{R}_B^{(m)} \quad \text{with} \quad \mathbf{R}_B^{(m)}=\int_{V^{(m)}}\mathbf{H}^{(m)T}\mathbf{f}^{(m)}dV^{(m)}$$

$$\mathbf{R}_S=\sum_m \mathbf{R}_S^{(m)} \quad \text{with} \quad \mathbf{R}_S^{(m)}=\int_{S_f^{(m)}}\mathbf{H}_S^{(m)T}\mathbf{f}^{S(m)}dS^{(m)}$$

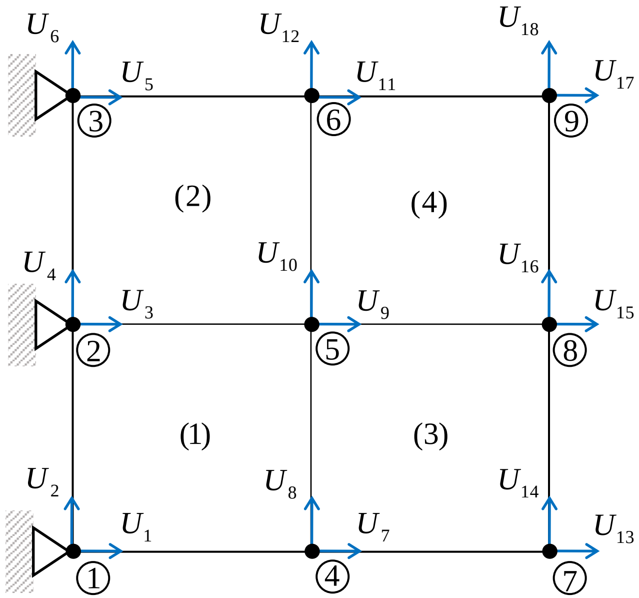
$$\begin{bmatrix} u^{(4)} \\ v^{(4)} \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 & h_4 \end{bmatrix} \mathbf{u}^{(4)} = \mathbf{H}^{(4)} \mathbf{u}^{(4)}$$

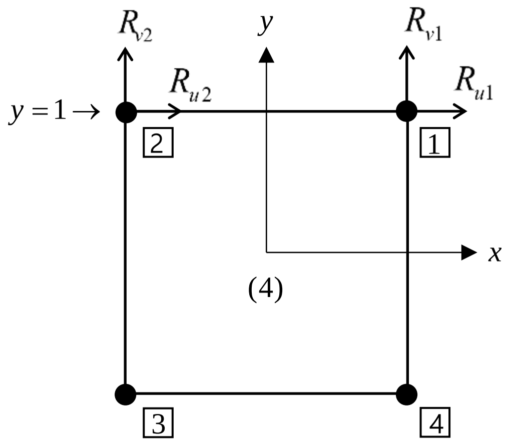
$$\mathbf{H}_S^{(4)} = \begin{bmatrix} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 & h_4 \end{bmatrix} \text{ at } y = 1$$

$$= \begin{bmatrix} (1+x)/2 & (1-x)/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & (1+x)/2 & (1-x)/2 & 0 & 0 \end{bmatrix}$$

$$\mathbf{f}^{S(4)} = \begin{bmatrix} 0 \\ -0.5 \end{bmatrix}$$

$$\mathbf{R}^{(4)} = \mathbf{R}_S^{(4)} = \int_{S_f^{(4)}} \mathbf{H}_S^{(4)T} \mathbf{f}^{S(4)} dS^{(4)} = 1 \times \int_{-1}^{1} \mathbf{H}_S^{(4)T} \mathbf{f}^{S(4)} dx = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -0.5 \\ -0.5 \\ 0 \\ 0 \end{bmatrix}$$





$$\mathbf{R} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -0.5 \\ 0.7 \\ \vdots \\ 0 \\ -0.5 \end{bmatrix} \begin{matrix} \\ \\ \\ \leftarrow R_{12} \\ \leftarrow R_{13} \\ \\ \\ \leftarrow R_{18} \end{matrix}$$

We obtain the equilibrium equation.

$$\mathbf{KU} = \mathbf{R}$$

Imposition of displacement BC

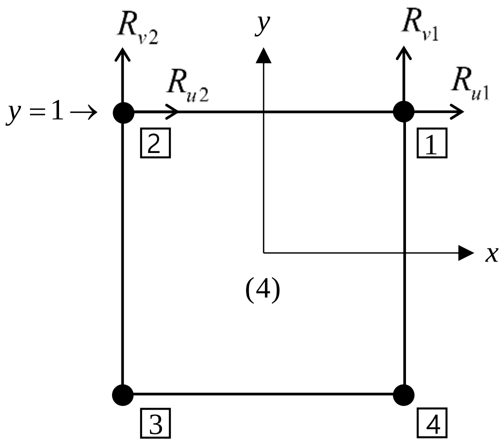
Displacement BC : $U_1 = U_2 = U_3 = U_4 = U_5 = U_6 = 0$

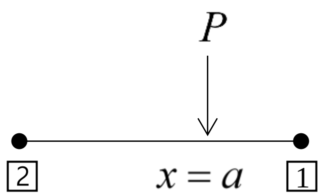
$$\underset{12\times 12}{\tilde{\mathbf{K}}}\,\tilde{\mathbf{U}} = \tilde{\mathbf{R}}$$

Strain and stress

$$\begin{cases} \boldsymbol{\varepsilon}^{(m)} = \mathbf{B}^{(m)}\mathbf{u}^{(m)} \\ \boldsymbol{\tau}^{(m)} = \mathbf{C}^{(m)}\mathbf{B}^{(m)}\mathbf{u}^{(m)} \end{cases}$$

Consideration of a point load





$$\mathbf{f}^{S(4)}=\begin{bmatrix}0\\-P\delta(x-a)\end{bmatrix}$$ with Dirac's delta function δ

$\delta(x)=+\infty$ at $x=0$ and $\delta(x)=0$ at $x\neq 0$

$$\int_{-\infty}^{\infty}\delta(x)dx=1,\quad \int_{-\infty}^{\infty}f(x)\delta(x-a)dx=f(a)$$

$$\mathbf{R}^{(4)} = \mathbf{R}_S^{(4)} = \int_{S_f^{(4)}} \mathbf{H}_S^{(4)T} \mathbf{f}^{S(4)} dS^{(4)}$$

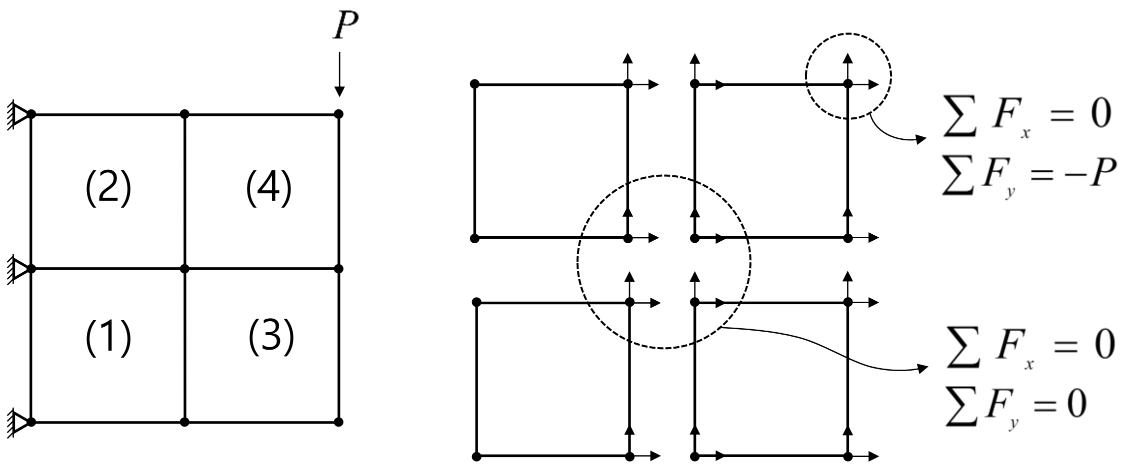
$$= 1 \times \int_{-1}^{1} \mathbf{H}_S^{(4)T} \mathbf{f}^{S(4)} dx = 1 \times \begin{bmatrix} (1+a)/2 & 0 \\ (1-a)/2 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & (1+a)/2 \\ 0 & (1-a)/2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ -P \end{bmatrix} = -P \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ (1+a)/2 \\ (1-a)/2 \\ 0 \\ 0 \end{bmatrix}$$

5. FE solutions and Convergence

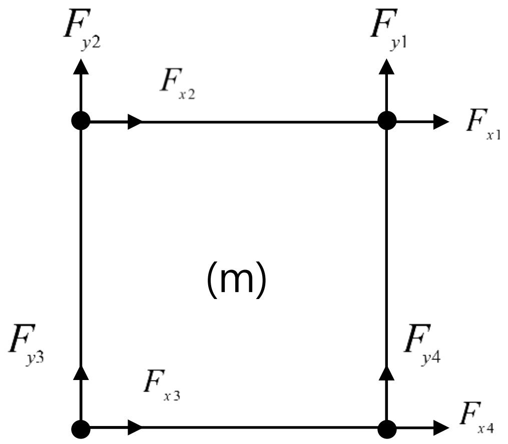
Global equilibrium at nodes

The internal forces acting at nodes are in equilibrium with the external forces.

Ex)



Let us consider the internal forces of element m



$\mathbf{F}^{(m)} = [F_{x1} \quad F_{x2} \quad F_{x3} \quad F_{x4} \quad F_{y1} \quad F_{y2} \quad F_{y3} \quad F_{y4}]^T$; the vector of element nodal point forces

PVW is satisfied in element m

$$\int_{V^{(m)}} (\delta \boldsymbol{\varepsilon}^{(m)})^T \boldsymbol{\tau}^{(m)} dV = \delta \mathbf{u}^T \mathbf{F}^{(m)} \quad \rightarrow \quad \cancel{\delta \mathbf{u}^T} \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV = \cancel{\delta \mathbf{u}^T} \mathbf{F}^{(m)}$$

$$\mathbf{F}^{(m)} = \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV^{(m)}$$

$$\sum_m \mathbf{F}^{(m)} = \sum_m \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV^{(m)} \qquad \leftarrow \quad \boldsymbol{\tau}^{(m)} = \mathbf{C}^{(m)}\mathbf{B}^{(m)}\mathbf{U}$$

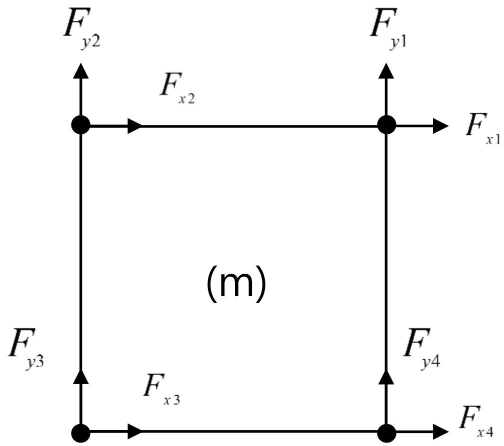
$$= \sum_m \left[\int_{V^{(m)}} (\mathbf{B}^{(m)})^T \mathbf{C}^{(m)} \mathbf{B}^{(m)} dV^{(m)} \right] \mathbf{U}$$

$$= \mathbf{K}\mathbf{U} = \mathbf{R}$$

$$\rightarrow \quad \sum_m \mathbf{F}^{(m)} = \mathbf{R}$$

Global equilibrium in each element

Each element is in equilibrium under its element forces, $\mathbf{F}^{(m)}$.



$$\sum F_x = 0$$
$$\sum F_y = 0$$
$$\sum M_z = 0$$

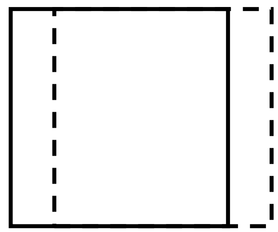
$$\mathbf{F}^{(m)} = \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV^{(m)}$$

$$\delta \mathbf{u}_R^T \mathbf{F}^{(m)} = \delta \mathbf{u}_R^T \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV^{(m)}$$

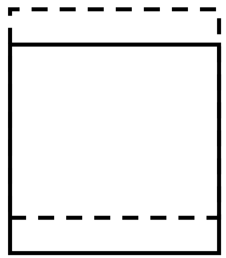
where $\delta \mathbf{u}_R$ is a virtual nodal displacement vector corresponding to "rigid body motion."

Rigid body motions in 2D: two translations and one rotation

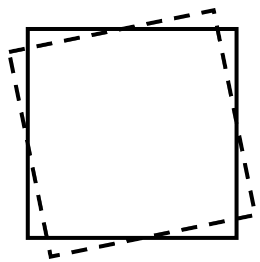
$$\delta\mathbf{u}_{Rx} = \begin{bmatrix}1 & 1 & 1 & 1 & 0 & 0 & 0 & 0\end{bmatrix}^T$$



$$\delta\mathbf{u}_{Ry} = \begin{bmatrix}0 & 0 & 0 & 0 & 1 & 1 & 1 & 1\end{bmatrix}^T$$



$$\delta\mathbf{u}_{R\theta} = \begin{bmatrix}-1 & -1 & 1 & 1 & 1 & -1 & -1 & 1\end{bmatrix}^T$$



$$\delta \mathbf{u}_R^T \mathbf{F}^{(m)} = \delta \mathbf{u}_R^T \int_{V^{(m)}} (\mathbf{B}^{(m)})^T \boldsymbol{\tau}^{(m)} dV^{(m)}$$

$$= \int_{V^{(m)}} (\underline{\mathbf{B}^{(m)} \delta \mathbf{u}_R})^T \boldsymbol{\tau}^{(m)} dV^{(m)}$$

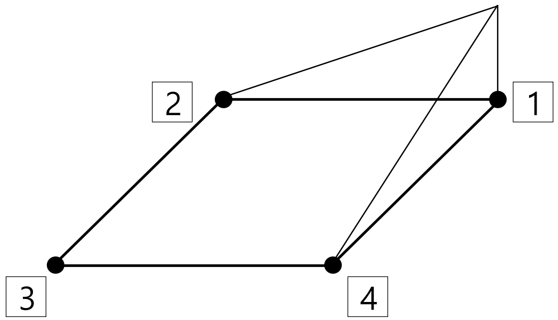
$\rightarrow \delta \boldsymbol{\varepsilon}_R = 0$: Virtual strain in rigid body motions

$$= 0$$

$$\delta \mathbf{u}_R^T \mathbf{F}^{(m)} = 0 \quad \rightarrow \quad \delta \mathbf{u}_{Rx}^T \mathbf{F}^{(m)} = \sum \mathbf{F}_x = 0, \quad \delta \mathbf{u}_{Ry}^T \mathbf{F}^{(m)} = \sum \mathbf{F}_y = 0, \quad \delta \mathbf{u}_{R\theta}^T \mathbf{F}^{(m)} = \sum \mathbf{M}_z = 0$$

Ex) $\delta \mathbf{u}_R = \delta \mathbf{u}_{Rx} \quad \rightarrow \quad \delta \mathbf{u}_{Rx}^T \mathbf{F}^{(m)} = F_{x1} + F_{x2} + F_{x3} + F_{x4} = 0$

To satisfy this property, rigid body motions should be contained in the element displacement interpolation. That is, the displacement interpolation function should represent rigid body motions.



$$h_1 = \frac{1}{4}(1+x)(1+y)$$

$$h_2 = \frac{1}{4}(1-x)(1+y)$$

$$h_3 = \frac{1}{4}(1-x)(1-y)$$

$$h_4 = \frac{1}{4}(1+x)(1-y)$$

$$u = \sum_i h_i u_i$$

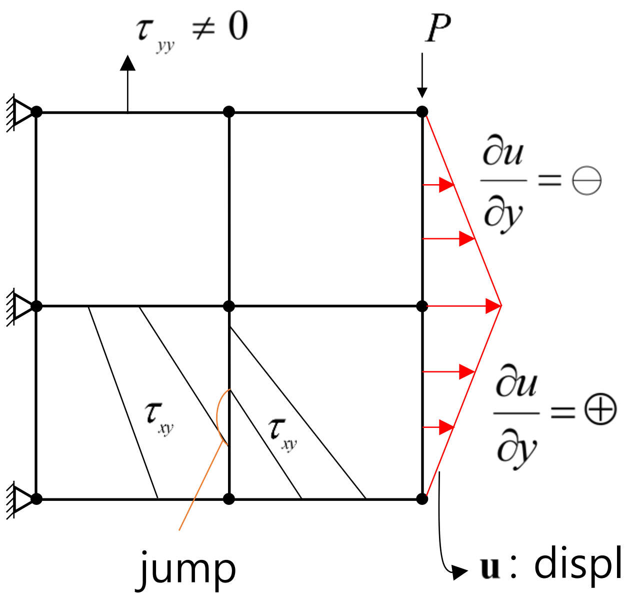
$$v = \sum_i h_i v_i$$

Let us assume $u_i = \Delta$ (rigid body translation in the x-direction)

$$u = \sum_i h_i \Delta = \Delta \sum_i h_i = \Delta \quad \rightarrow \quad \sum_i h_i = 1$$

$$\varepsilon_{xx} = \frac{\partial u}{\partial x} = \sum_i \frac{\partial h_i}{\partial x}\Delta = \Delta \sum_i \frac{\partial h_i}{\partial x} = 0 \quad \rightarrow \quad \sum_i \frac{\partial h_i}{\partial x} = 0, \quad \sum_i \frac{\partial h_i}{\partial y} = 0$$

→ "Basic requirements for shape functions"



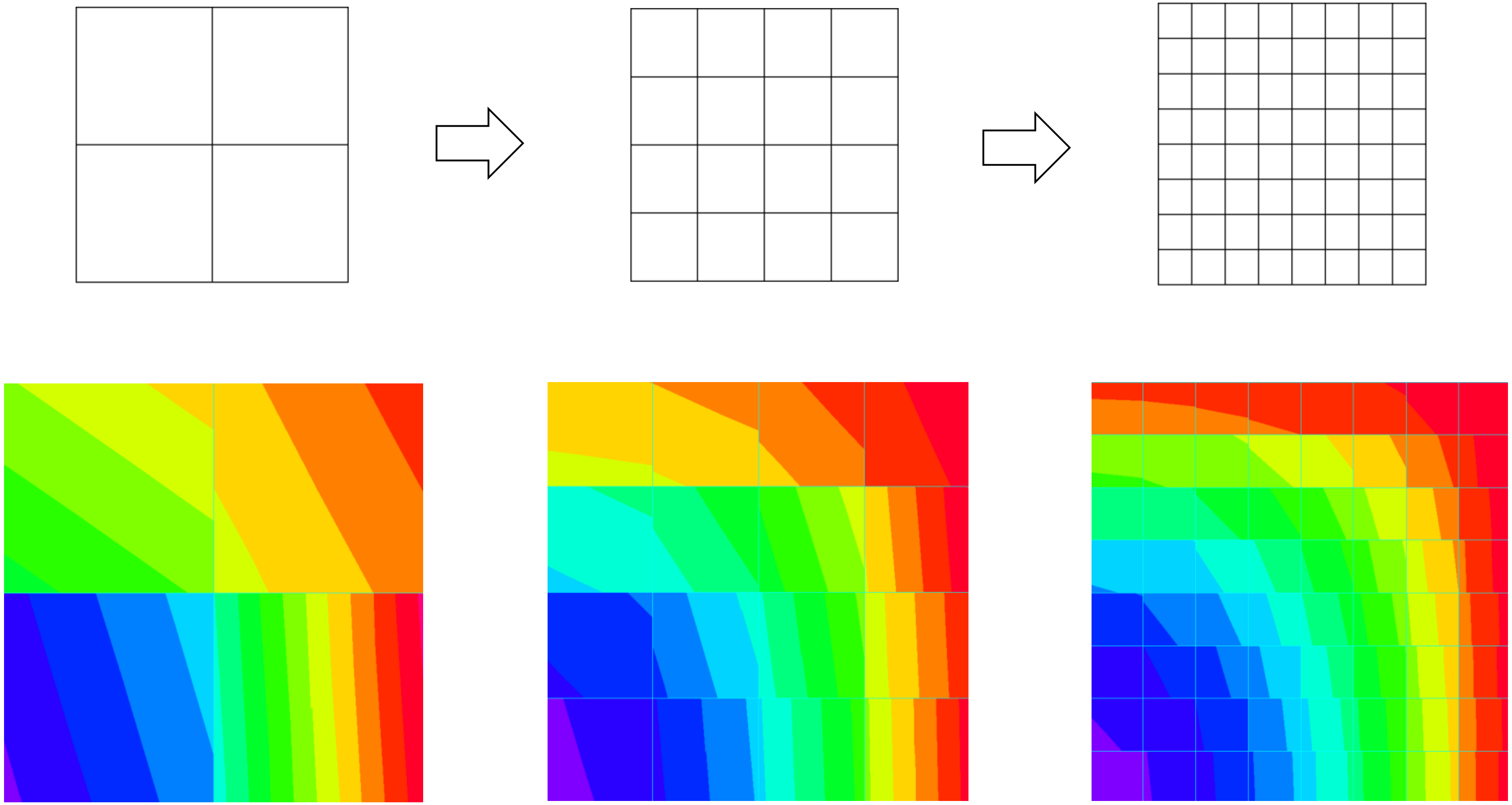
① Compatibility
② Stress-strain law

} are satisfied exactly.

③ Global equilibrium at nodes and in each element is satisfied.

④ <u>Equilibrium on differential level is not satisfied.</u>

As we take a finer and finer mesh, the stress/strain discontinuity becomes smaller and smaller.



Principle of Virtual Work

PVW : $\int_V \delta\boldsymbol{\varepsilon}^T \boldsymbol{\tau}\, dV = \int_V \delta\mathbf{u}^T \mathbf{f}^B dV + \int_{S_f} \delta\mathbf{u}^T \mathbf{f}^S dS$

$\delta\boldsymbol{\varepsilon}(\mathbf{u})$: virtual strain, $\boldsymbol{\tau}(\mathbf{u})$: exact stress (solution)

$\delta\mathbf{u}$: virtual displacement, $\mathbf{u}$: exact displacement (solution)

③ Space of function

$\Rightarrow$ Find $\mathbf{u} \in \vec{V}$ such that $\int_V \delta\boldsymbol{\varepsilon}^T \boldsymbol{\tau}\, dV = \int_V \delta\mathbf{u}^T \mathbf{f}^B dV + \int_{S_f} \delta\mathbf{u}^T \mathbf{f}^S dS$ for $\forall\, \delta\mathbf{u} \in \vec{V}$

$\Rightarrow$ Find $\mathbf{u} \in \vec{V}$ such that $a(\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v})$ for $\forall\, \mathbf{v} \in \vec{V}$

① Linear form

② Bilinear form

① Linear form $(\mathbf{f}, \cdot\,)$

- Linear form is linear in the second term.
- Linearity : $(\mathbf{f}, \gamma_1 \mathbf{v}_1 + \gamma_2 \mathbf{v}_2) = \gamma_1 (\mathbf{f}, \mathbf{v}_1) + \gamma_2 (\mathbf{f}, \mathbf{v}_2)$

② Bilinear form $a(\,\cdot,\,\cdot\,)$

- Bilinear form is linear in the first and second terms including all derivatives, material law and integration.
- Bilinearity :

$$a(\gamma_1\mathbf{u}_1+\gamma_2\mathbf{u}_2,\mathbf{v})=\gamma_1 a(\mathbf{u}_1,\mathbf{v})+\gamma_2 a(\mathbf{u}_2,\mathbf{v})$$
$$a(\mathbf{u},\gamma_1\mathbf{v}_1+\gamma_2\mathbf{v}_2)=\gamma_1 a(\mathbf{u},\mathbf{v}_1)+\gamma_2 a(\mathbf{u},\mathbf{v}_2)$$

③ Solution space of function $\vec{V}$

- $\vec{V}=\left\{\mathbf{v}\,|\,\mathbf{v}\in L^2(Vol);\;\; \dfrac{\partial v_i}{\partial x_j}\in L^2(Vol),\, i,\, j=1,2,3;\;\; v_i\big|_{S_u}=0,\, i=1,2,3\right\},$

  where $L^2(Vol)=\left\{\mathbf{w}\,|\,\mathbf{w}\text{ is defined in }Vol\text{ and }\|\mathbf{w}\|^2_{L^2(Vol)}=\int_{Vol}\left(\sum_{i=1}^{3}(w_i)^2\right)dVol<+\infty\right\}$

- $L^2(Vol)$ is the space of square integrable functions in the volume.

Ellipticity

If a body is well supported (no rigid body motion is allowed),

$\exists \alpha > 0$ such that $a(\mathbf{w},\mathbf{w}) \geq \alpha \|\mathbf{w}\|_1^2$, $\forall \mathbf{w} \in \vec{V}$,

where $\|\mathbf{w}\|_1^2 = \int_{Vol} \left(\sum_{i=1}^{3} (w_i)^2 \right) dVol + \int_{Vol} \left(\sum_{i=1,\, j=1}^{3} \left(\frac{\partial w_i}{\partial x_j} \right)^2 \right) dVol$.

- This is an important property of $a(\,\cdot, \cdot\,)$ and $\exists$ denotes "There exists".
- $a(\mathbf{w},\mathbf{w})$ is twice the strain energy stored in the body when the body is subjected to the displacement field $\mathbf{w}\left(\in \vec{V}\right)$.
- Strain energy is greater than zero for any $\mathbf{w}$ different from zero displacement.

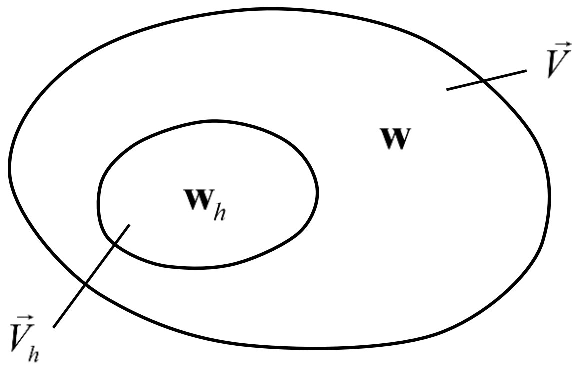
Convergence of FE Solutions

Let $\mathbf{u}_h$ be the FE (approximated) solution obtained from the following procedure.

Find $\mathbf{u}_h \in \vec{V}_h$ such that $a(\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h)$ for $\forall\, \mathbf{v}_h \in \vec{V}_h$,

where $\vec{V}_h = \left\{ \mathbf{v}_h \mid \mathbf{v}_h \in L^2(Vol);\ \ \dfrac{\partial (v_h)_i}{\partial x_j} \in L^2(Vol),\ i, j = 1,2,3;\ \ (v_h)_i\Big|_{S_u} = 0,\ i = 1,2,3 \right\}$.

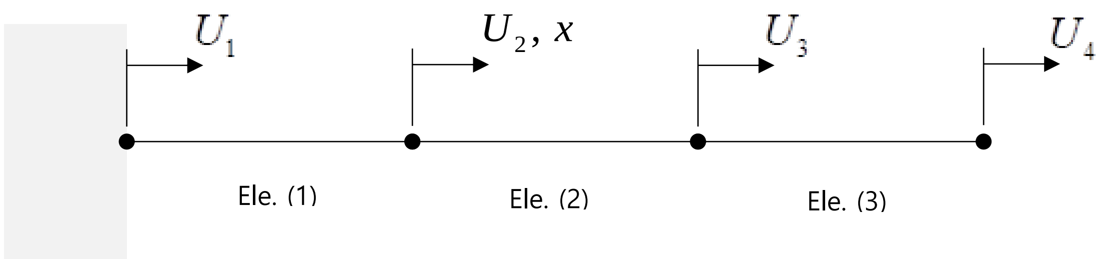
$\vec{V}_h$ is the function space of the FE solutions and $\vec{V}_h \subset \vec{V}$.



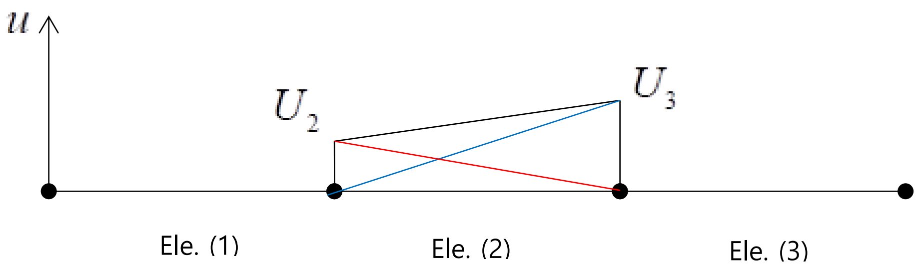
(Note) Function space for the FE solutions $\vec{V}_h$

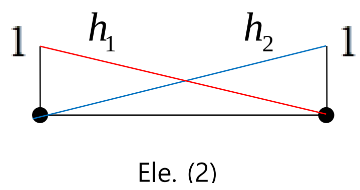
A set of functions that can be represented by FE basis functions.

Ex) Bar problem modeled by 3 elements

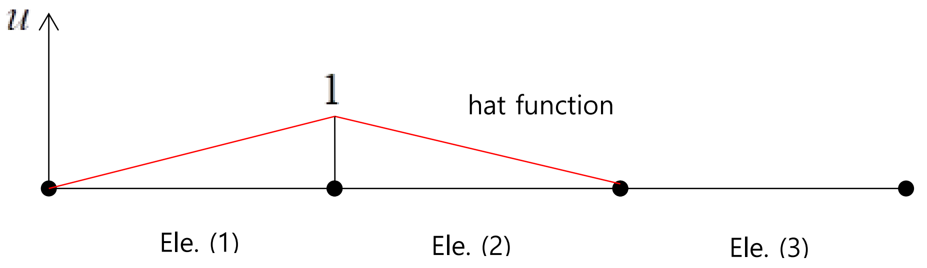


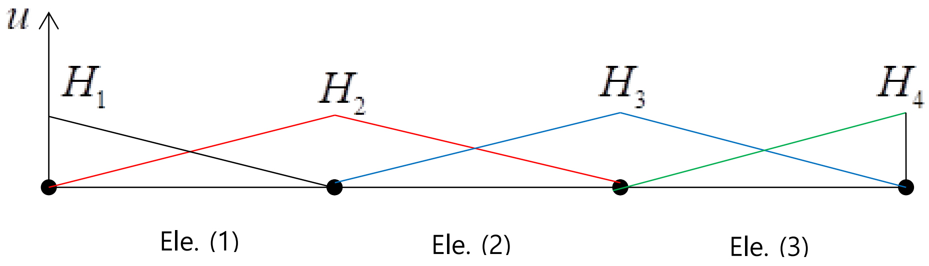
$$u^{(2)}(x) = \left(1 - \frac{x}{L}\right)U_2 + \frac{x}{L}U_3 = h_1(x)U_2 + h_2(x)U_3$$





When $U_2 = 1$ and other nodal displacements = 0, a FE basis function H_2 is defined.





Here, the function space of the FE solutions contains all the functions represented by

$$u = H_1(x)U_1 + H_2(x)U_2 + H_3(x)U_3 + H_4(x)U_4.$$

Considering the FE solution $\mathbf{u}_h$ and the exact solution $\mathbf{u}$, we have the following important properties.

Property 1.

Let the error $\mathbf{e}_h$ between the exact solution $\mathbf{u}$ and the FE solution $\mathbf{u}_h$ as $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$. Then, the first property is $a(\mathbf{e}_h, \mathbf{v}_h) = 0$ for $\forall \mathbf{v}_h \in \vec{V}_h$. (orthogonality of error)

(Proof)

PVW: $a(\mathbf{u}, \mathbf{v}) = (\mathbf{f}, \mathbf{v})$

$a(\mathbf{u}, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h)$ because $\mathbf{v}_h \in \vec{V}_h \subset \vec{V}$ (eq. 5.1)

$a(\mathbf{u}_h, \mathbf{v}_h) = (\mathbf{f}, \mathbf{v}_h)$ (eq. 5.2)

Subtracting (eq. 5.2) from (eq. 5.1), $a(\mathbf{u}, \mathbf{v}_h) - a(\mathbf{u}_h, \mathbf{v}_h) = 0$.

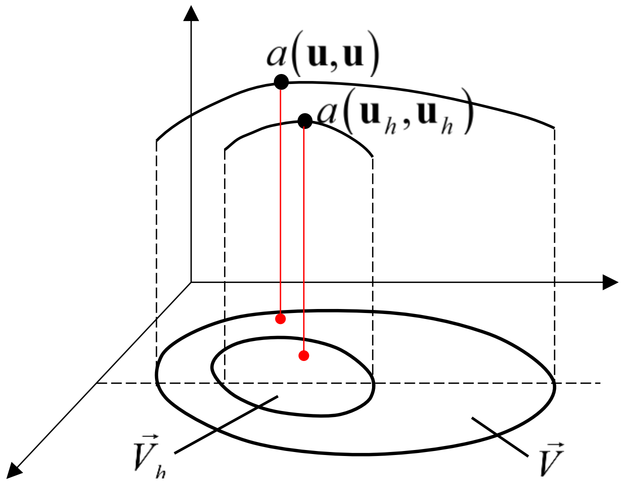
Due to bilinearity, $a(\mathbf{u} - \mathbf{u}_h, \mathbf{v}_h) = 0$

Finally, $a(\mathbf{e}_h, \mathbf{v}_h) = 0$ with $\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$.

∴ The error is "orthogonal in $a(\bullet, \bullet)$" to all $\mathbf{v}_h$ in $\vec{V}_h$.

Property 2.

The strain energy corresponding to the FE solution is always smaller than or equal to the strain energy corresponding to the exact solution: $a(\mathbf{u}_h, \mathbf{u}_h) \le a(\mathbf{u}, \mathbf{u})$.



(Proof)

$$a(\mathbf{u},\mathbf{u}) = a(\mathbf{u}_h + \mathbf{e}_h, \mathbf{u}_h + \mathbf{e}_h)$$

$$= a(\mathbf{u}_h,\mathbf{u}_h) + 2a(\mathbf{u}_h,\mathbf{e}_h) + a(\mathbf{e}_h,\mathbf{e}_h)$$

0

orthogonality

$$= a(\mathbf{u}_h,\mathbf{u}_h) + a(\mathbf{e}_h,\mathbf{e}_h)$$

$a(\mathbf{e}_h,\mathbf{e}_h) \ge 0$ from ellipticity

$\therefore \; a(\mathbf{u}_h,\mathbf{u}_h) \le a(\mathbf{u},\mathbf{u})$.

From "Property 2", the following convergence directions are obtained.

	FE solutions	→	Exact solutions
Energy	E_h	$\leq$	E
Displacement	$\mathbf{u}_h$	$\leq$	$\mathbf{u}$
Stiffness	$\mathbf{K}_h$	$\geq$	$\mathbf{K}$
Strain & Stress	$\boldsymbol{\varepsilon}_h, \ \boldsymbol{\tau}_h$	$\leq$	$\boldsymbol{\varepsilon}, \ \boldsymbol{\tau}$

Property 3.

The FE solution $\mathbf{u}_h$ is chosen from all the possible displacement patterns $\mathbf{v}_h$ in $\vec{V}_h$ such that the strain energy corresponding to the error ($\mathbf{e}_h = \mathbf{u} - \mathbf{u}_h$) is the minimum:

$$a(\mathbf{e}_h, \mathbf{e}_h) \le a(\mathbf{u} - \mathbf{v}_h, \mathbf{u} - \mathbf{v}_h) \quad \text{for } \forall \mathbf{v}_h \in \vec{V}_h.$$

(Proof)

Let us consider $\mathbf{w}_h \in \vec{V}_h$.

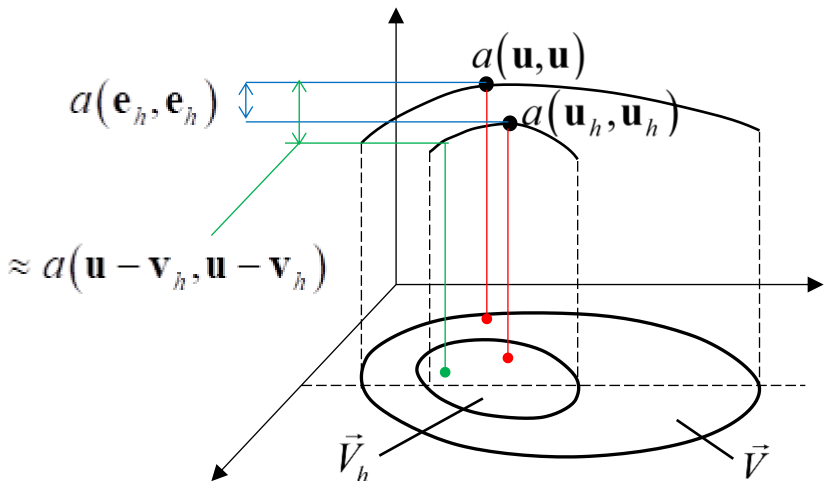
$$a(\mathbf{e}_h + \mathbf{w}_h, \mathbf{e}_h + \mathbf{w}_h) = a(\mathbf{e}_h, \mathbf{e}_h) + a(\mathbf{w}_h, \mathbf{w}_h) \quad \text{for } \forall \mathbf{w}_h \in \vec{V}_h.$$

Due to ellipticity $a(\mathbf{w}_h, \mathbf{w}_h) \ge 0$,

$$a(\mathbf{e}_h, \mathbf{e}_h) \le a(\mathbf{e}_h + \mathbf{w}_h, \mathbf{e}_h + \mathbf{w}_h)$$

Let $\mathbf{w}_h = \mathbf{u}_h - \mathbf{v}_h$ and then $\mathbf{e}_h + \mathbf{w}_h = \mathbf{u} - \mathbf{v}_h$.

$$\therefore \quad a(\mathbf{e}_h, \mathbf{e}_h) \le a(\mathbf{u} - \mathbf{v}_h, \mathbf{u} - \mathbf{v}_h).$$



Principle of Minimum Potential Energy

A structure or body shall deform or displace to a position that minimizes the total potential energy: (Total potential energy) = (Strain energy potential) – (Force potential).

$$\Pi(\mathbf{u}) = \frac{1}{2}\int_V \boldsymbol{\varepsilon}^T \mathbf{C}\boldsymbol{\varepsilon} dV - \left[\int_V \mathbf{u}^T \mathbf{f}^B dV + \int_{S_f} \mathbf{u}^T \mathbf{f}^S dS\right] = \frac{1}{2} a(\mathbf{u},\mathbf{u}) - (\mathbf{f},\mathbf{u})$$

Then, the exact solution is given by

$$\mathbf{u} = \min \Pi(\mathbf{w}) \quad \text{for} \quad \forall \mathbf{w} \in \vec{V}.$$

The solution procedure for linear elastic problems is represented by

Find $\mathbf{u} \in \vec{V}$ such that $\Pi(\mathbf{u}) \le \Pi(\mathbf{w})$ for $\forall \mathbf{w} \in \vec{V}$.

Let $\mathbf{w} = \mathbf{u} + \mathbf{v}$.

$$\Pi(\mathbf{u}) \leq \Pi(\mathbf{w})$$

$$\Pi(\mathbf{w}) - \Pi(\mathbf{u}) \geq 0$$

$$\frac{1}{2} a\left(\mathbf{u}+\mathbf{v}, \mathbf{u}+\mathbf{v}\right) - \left(\mathbf{f}, \mathbf{u}+\mathbf{v}\right) - \left[\frac{1}{2} a\left(\mathbf{u}, \mathbf{u}\right) - \left(\mathbf{f}, \mathbf{u}\right)\right] \geq 0$$

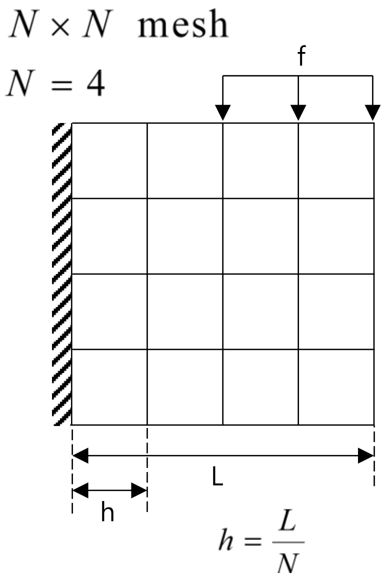
$$a\left(\mathbf{u}, \mathbf{v}\right) - \left(\mathbf{f}, \mathbf{v}\right) + \frac{1}{2} a\left(\mathbf{v}, \mathbf{v}\right) \geq 0$$

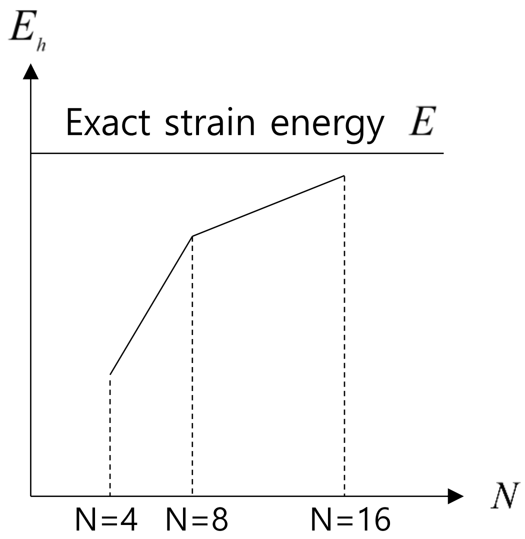
$\therefore \; a\left(\mathbf{u}, \mathbf{v}\right) - \left(\mathbf{f}, \mathbf{v}\right) = 0$: PVW

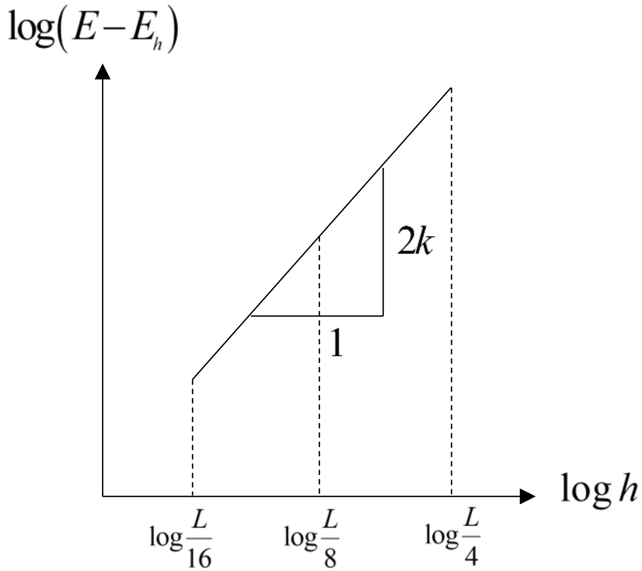
The principle can be used to find the finite element solution $\mathbf{u}_h$.

Find $\mathbf{u}_h \in \vec{V}_h$ such that $\Pi(\mathbf{u}_h) \leq \Pi(\mathbf{w}_h)$ for $\forall \mathbf{w}_h \in \vec{V}_h$.

Rate of Convergence







$$\log\left|E - E_h\right| = \log C + 2k \log h$$

where E_h: FE strain energy, h: element size and k: order of convergence

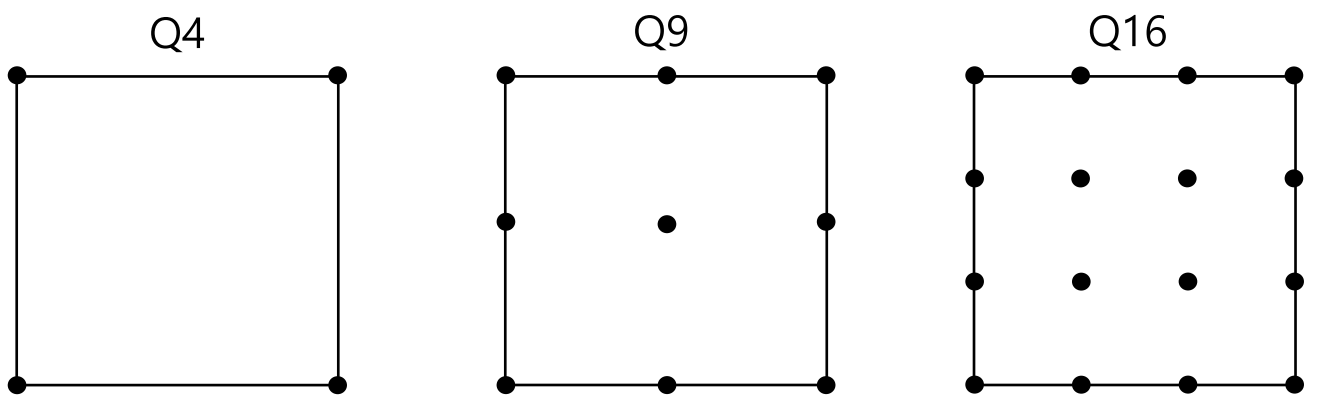
$\left|E - E_h\right| = Ch^{2k}$: Rate of convergence

→ Constant depending on geometry, element, etc.

Convergence for displacement gradients, strain and stress: $\|\mathbf{e}\|_{L^2(Vol)} = \sqrt{\int_{Vol}\left(\sum_{i=1}^{3}\left(e_i\right)^2\right)dVol} = C\,h^k$

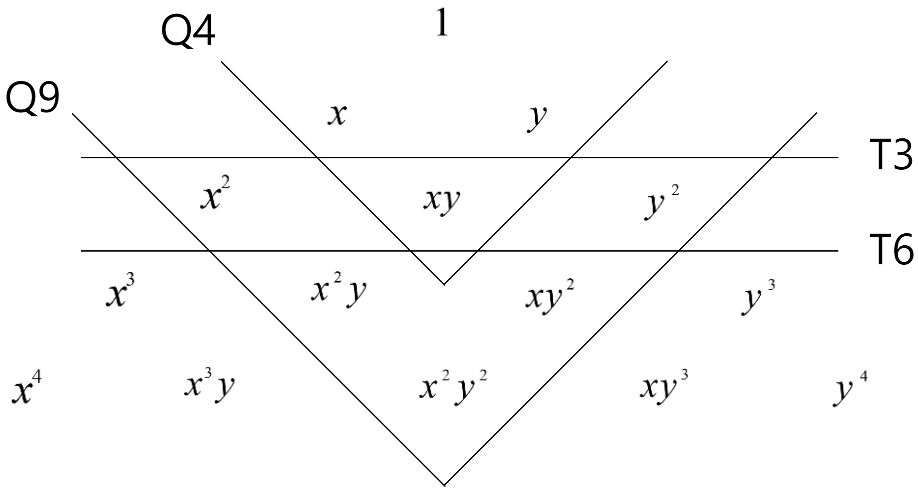
Convergence for displacements: $\|\mathbf{e}\|_{L^2(Vol)} = \sqrt{\int_{Vol}\left(\sum_{i=1}^{3}\left(e_i\right)^2\right)dVol} = C\,h^{k+1}$

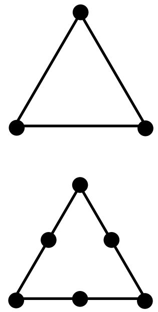
In 2D quadrilateral elements,



4-node element: $k=1$, 9-node element: $k=2$, 16-node element: $k=3$

Pascal Triangle



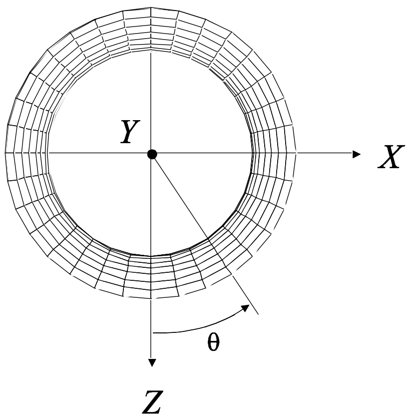


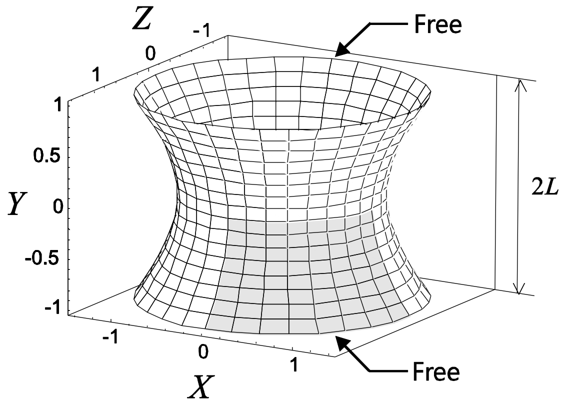
Q4, T3: $k = 1$

Q9, T6: $k = 2$

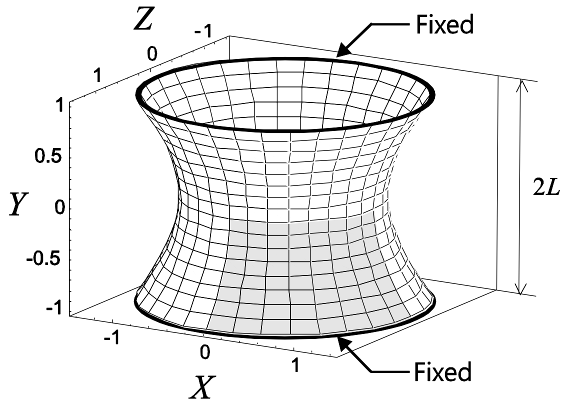
Q16, T10: $k = 3$ with k : degree of complete polynomial

Example - Hyperboloid shell problem

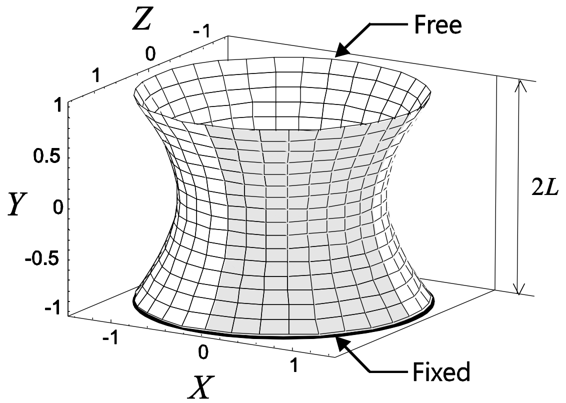




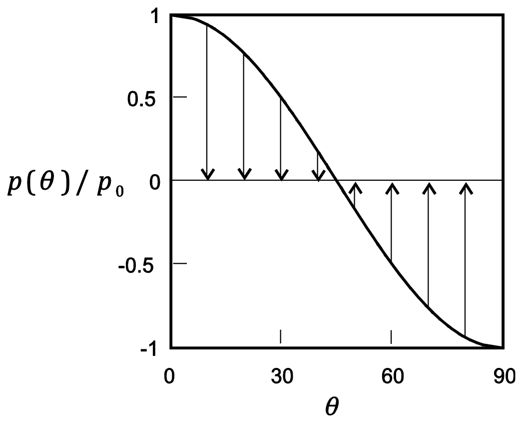
(a)



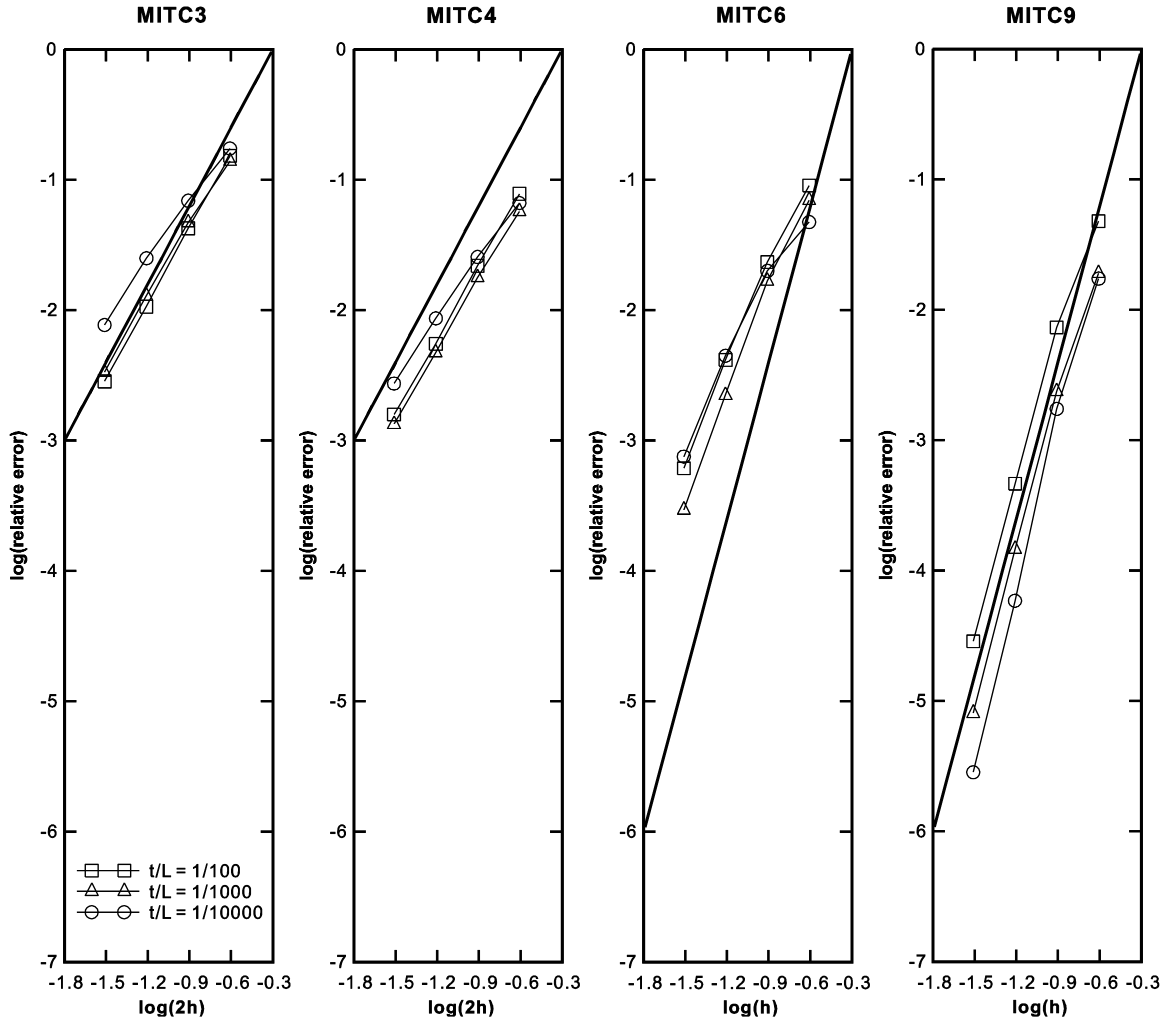
(b)



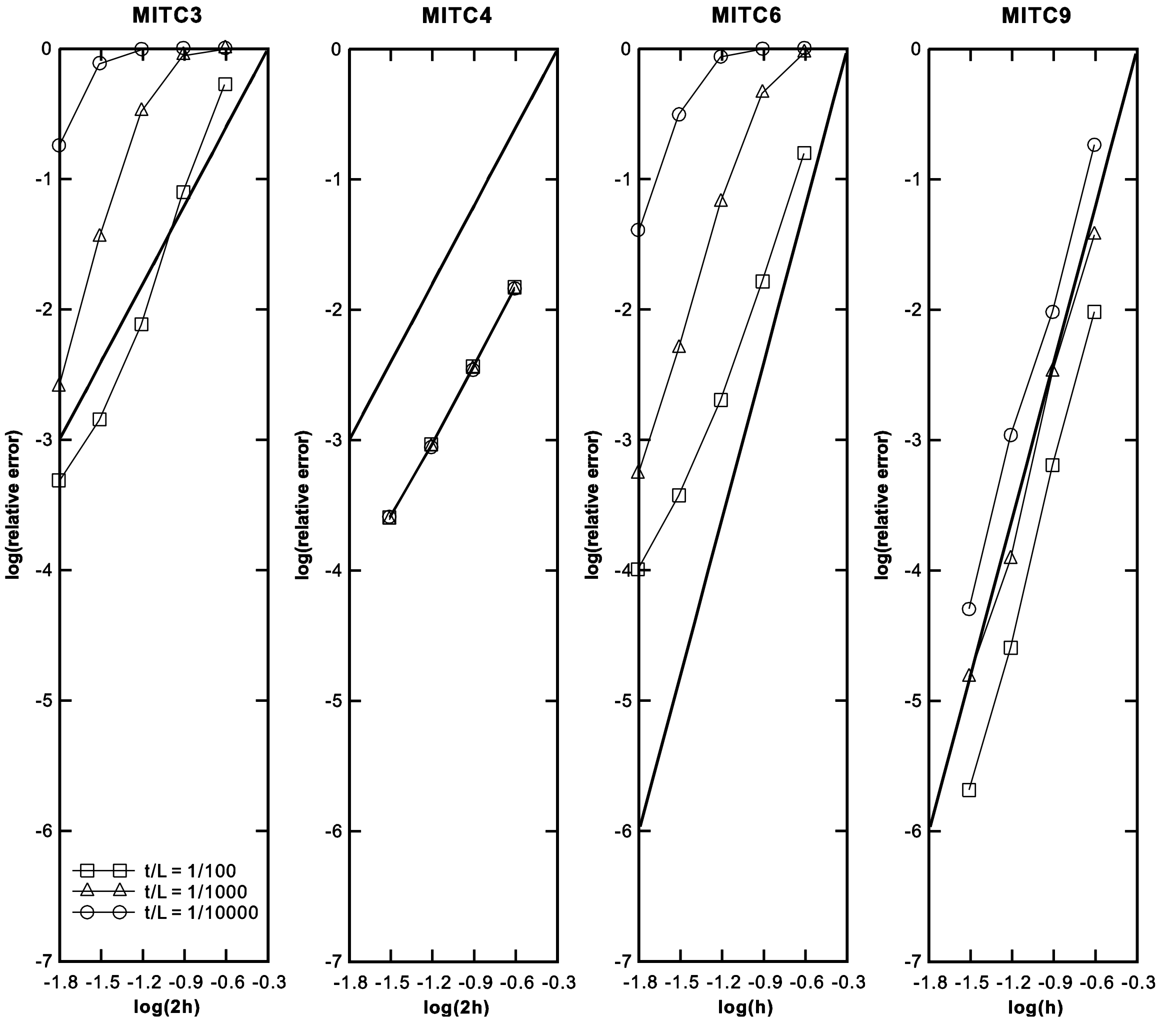
(c)



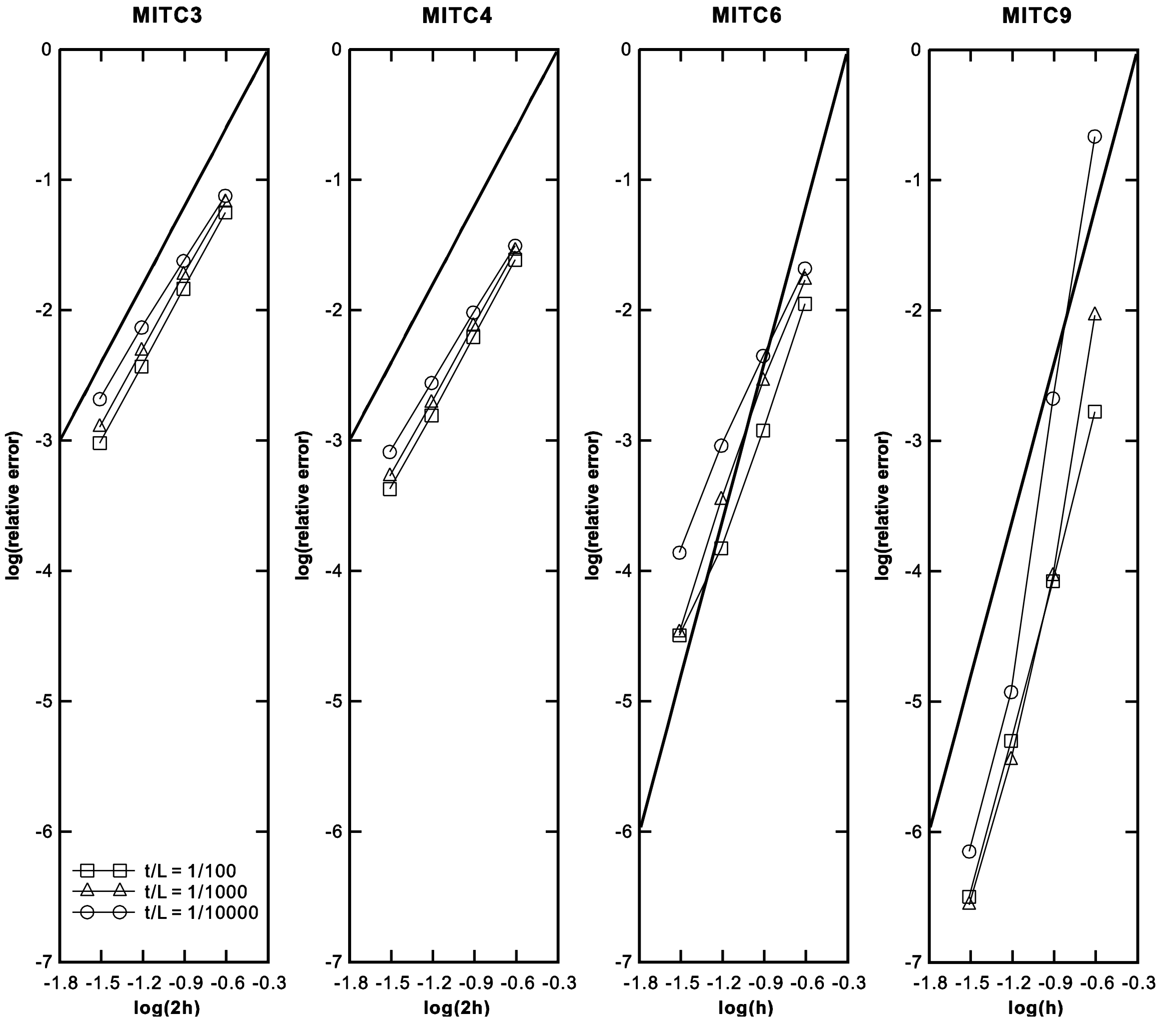
Fixed-fixed boundary



Free-free boundary



Fixed-free boundary

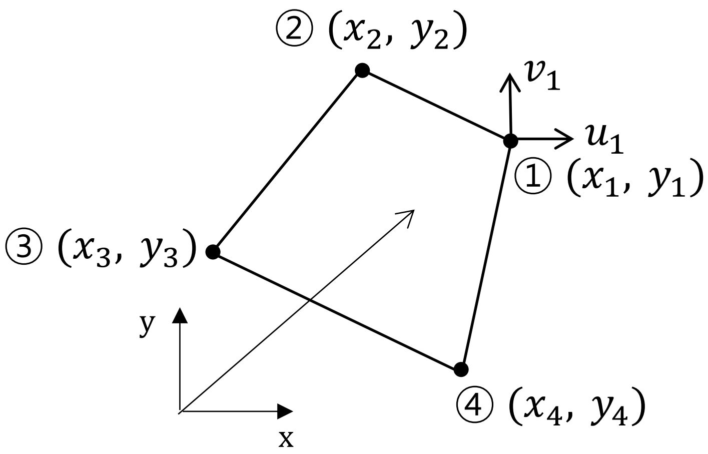


The convergence behavior depends on

- Problems: geometry, material properties, force and displacement BCs …
- Finite elements used: element order (degree of complete polynomial), element geometry (triangular elements, quadrilateral elements…) and so on.
- Meshes: regular meshes, irregular (distorted) meshes …

6. Isoparametric Finite Element Procedure

Let us consider a 2D 4-node element m



Interpolation of geometry

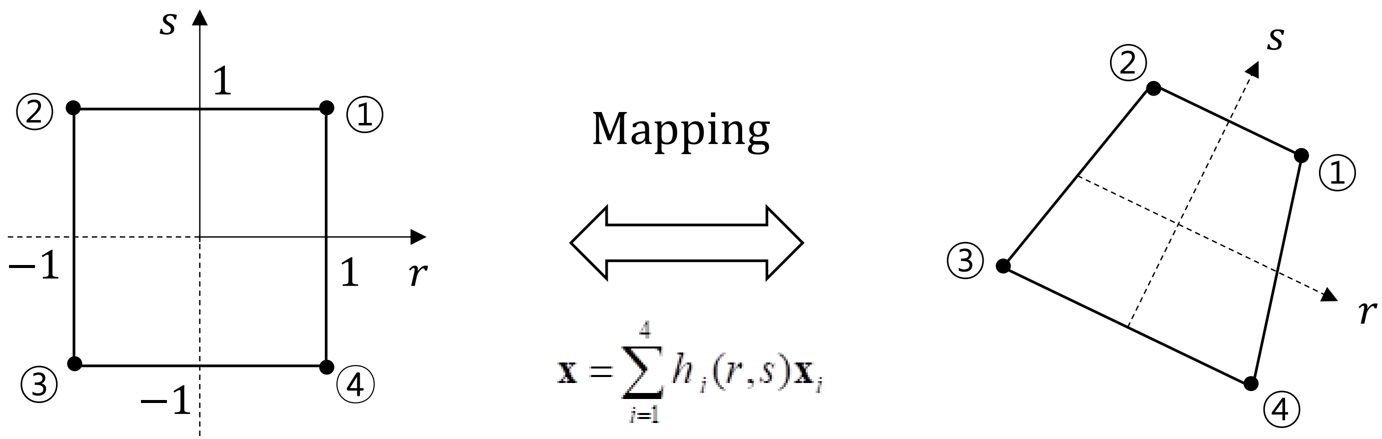
$$x^{(m)} = \sum_{i=1}^{4} h_i x_i , \quad y^{(m)} = \sum_{i=1}^{4} h_i y_i ,$$ where x_i and y_i are nodal coordinates.

Interpolation of displacements

$$u^{(m)} = \sum_{i=1}^{4} h_i u_i , \quad v^{(m)} = \sum_{i=1}^{4} h_i v_i ,$$ where u_i and v_i are nodal displacements.

The same interpolation functions (h_i) are used for geometry and displacements.

Shape functions are defined in the natural coordinate system: $h_i = h_i(r, s)$

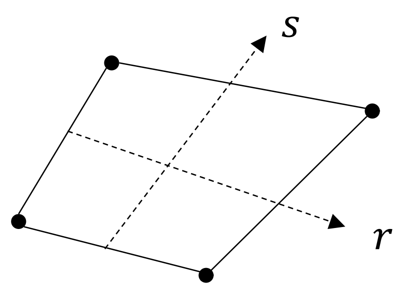


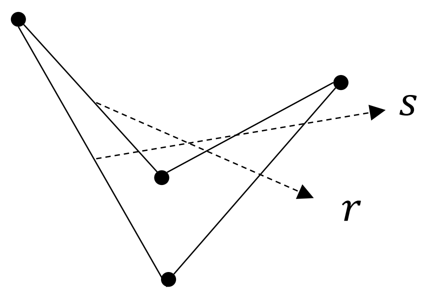
< Natural coordinate system > **< Global coordinate system >**

$$h_1 = \frac{1}{4}(1+r)(1+s),\ \ h_2 = \frac{1}{4}(1-r)(1+s),\ \ h_3 = \frac{1}{4}(1-r)(1-s),\ \ h_4 = \frac{1}{4}(1+r)(1-s)$$

Since the same shape functions are used for coordinates and displacements, we call the element an isoparametric element.

(Note) The element must give a unique correspondence between (r, s) and (x, y).





$$\mathbf{u}(r,s)=\begin{bmatrix} u^{(m)} \\ v^{(m)} \end{bmatrix}=\begin{bmatrix} h_1 & h_2 & h_3 & h_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & h_1 & h_2 & h_3 & h_4 \end{bmatrix}\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \end{bmatrix}=\mathbf{H}^{(m)}\mathbf{u}^{(m)}$$

In order to construct $\mathbf{B}^{(m)}$ and $\mathbf{K}^{(m)}$, we need to calculate $\dfrac{\partial u^{(m)}}{\partial x^{(m)}}$, $\dfrac{\partial u^{(m)}}{\partial y^{(m)}}$, $\dfrac{\partial v^{(m)}}{\partial x^{(m)}}$ and $\dfrac{\partial v^{(m)}}{\partial y^{(m)}}$.

(Note) If $u^{(m)}$ is a function of $x^{(m)}$ and $y^{(m)}$, we can directly calculate $\dfrac{\partial u^{(m)}}{\partial x^{(m)}}$. However, $u^{(m)}$ is not a function of $x^{(m)}$ and $y^{(m)}$, but a function of r and s.

Jacobian matrix $\mathbf{J}$

$$u^{(m)} = u^{(m)}(r,s) \quad \text{and} \quad v^{(m)} = v^{(m)}(r,s)$$

$$x^{(m)} = x^{(m)}(r,s) \quad \text{and} \quad y^{(m)} = y^{(m)}(r,s)$$

"Chain rule"

$$\frac{\partial u^{(m)}}{\partial r} = \frac{\partial x^{(m)}}{\partial r}\frac{\partial u^{(m)}}{\partial x^{(m)}} + \frac{\partial y^{(m)}}{\partial r}\frac{\partial u^{(m)}}{\partial y^{(m)}}$$

$$\frac{\partial u^{(m)}}{\partial s} = \frac{\partial x^{(m)}}{\partial s}\frac{\partial u^{(m)}}{\partial x^{(m)}} + \frac{\partial y^{(m)}}{\partial s}\frac{\partial u^{(m)}}{\partial y^{(m)}}$$

$$\begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial r} \\ \dfrac{\partial u^{(m)}}{\partial s} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial x^{(m)}}{\partial r} & \dfrac{\partial y^{(m)}}{\partial r} \\ \dfrac{\partial x^{(m)}}{\partial s} & \dfrac{\partial y^{(m)}}{\partial s} \end{bmatrix} \begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x} \\ \dfrac{\partial u^{(m)}}{\partial y} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x} \\ \dfrac{\partial u^{(m)}}{\partial y} \end{bmatrix} \quad \text{with} \quad \mathbf{J} = \begin{bmatrix} \dfrac{\partial x^{(m)}}{\partial r} & \dfrac{\partial y^{(m)}}{\partial r} \\ \dfrac{\partial x^{(m)}}{\partial s} & \dfrac{\partial y^{(m)}}{\partial s} \end{bmatrix}$$

Similarly,

$$\begin{bmatrix} \dfrac{\partial v^{(m)}}{\partial r} \\ \dfrac{\partial v^{(m)}}{\partial s} \end{bmatrix} = \mathbf{J} \begin{bmatrix} \dfrac{\partial v^{(m)}}{\partial x} \\ \dfrac{\partial v^{(m)}}{\partial y} \end{bmatrix}.$$

$$\begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x^{(m)}} \\ \dfrac{\partial u^{(m)}}{\partial y^{(m)}} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial r} \\ \dfrac{\partial u^{(m)}}{\partial s} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \dfrac{\partial v^{(m)}}{\partial x} \\ \dfrac{\partial v^{(m)}}{\partial y} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dfrac{\partial v^{(m)}}{\partial r} \\ \dfrac{\partial v^{(m)}}{\partial s} \end{bmatrix}$$

Strain-displacement relation

$$\begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x^{(m)}} \\ \dfrac{\partial u^{(m)}}{\partial y^{(m)}} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial r} \\ \dfrac{\partial u^{(m)}}{\partial s} \end{bmatrix}$$

$$\begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial r} \\ \dfrac{\partial u^{(m)}}{\partial s} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial h_1}{\partial r} & \dfrac{\partial h_2}{\partial r} & \dfrac{\partial h_3}{\partial r} & \dfrac{\partial h_4}{\partial r} & 0 & 0 & 0 & 0 \\ \dfrac{\partial h_1}{\partial s} & \dfrac{\partial h_2}{\partial s} & \dfrac{\partial h_3}{\partial s} & \dfrac{\partial h_4}{\partial s} & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{u}^{(m)}$$

with $\mathbf{u}^{(m)} = \begin{bmatrix} u_1 & u_2 & u_3 & u_4 & v_1 & v_2 & v_3 & v_4 \end{bmatrix}^{\mathrm{T}}$

$$\begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x^{(m)}} \\ \dfrac{\partial u^{(m)}}{\partial y^{(m)}} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} \dfrac{\partial h_1}{\partial r} & \dfrac{\partial h_2}{\partial r} & \dfrac{\partial h_3}{\partial r} & \dfrac{\partial h_4}{\partial r} & 0 & 0 & 0 & 0 \\ \dfrac{\partial h_1}{\partial s} & \dfrac{\partial h_2}{\partial s} & \dfrac{\partial h_3}{\partial s} & \dfrac{\partial h_4}{\partial s} & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{u}^{(m)} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & 0 & 0 & 0 & 0 \\ b_1 & b_2 & b_3 & b_4 & 0 & 0 & 0 & 0 \end{bmatrix} \mathbf{u}^{(m)}$$

Similarly,

$$\begin{bmatrix} \dfrac{\partial v^{(m)}}{\partial x^{(m)}} \\ \dfrac{\partial v^{(m)}}{\partial y^{(m)}} \end{bmatrix} = \mathbf{J}^{-1} \begin{bmatrix} 0 & 0 & 0 & 0 & \dfrac{\partial h_1}{\partial r} & \dfrac{\partial h_2}{\partial r} & \dfrac{\partial h_3}{\partial r} & \dfrac{\partial h_4}{\partial r} \\ 0 & 0 & 0 & 0 & \dfrac{\partial h_1}{\partial s} & \dfrac{\partial h_2}{\partial s} & \dfrac{\partial h_3}{\partial s} & \dfrac{\partial h_4}{\partial s} \end{bmatrix} \mathbf{u}^{(m)} = \begin{bmatrix} 0 & 0 & 0 & 0 & a_1 & a_2 & a_3 & a_4 \\ 0 & 0 & 0 & 0 & b_1 & b_2 & b_3 & b_4 \end{bmatrix} \mathbf{u}^{(m)}$$

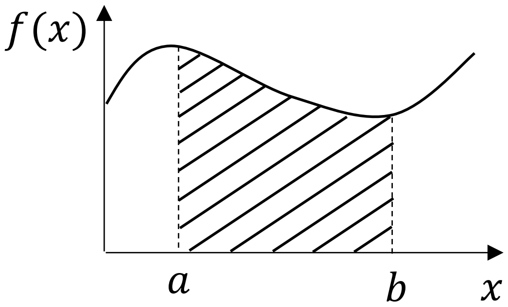
$$\begin{bmatrix} \varepsilon_{xx} \\ \varepsilon_{yy} \\ \gamma_{xy} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial u^{(m)}}{\partial x^{(m)}} \\ \dfrac{\partial v^{(m)}}{\partial y^{(m)}} \\ \dfrac{\partial u^{(m)}}{\partial y^{(m)}} + \dfrac{\partial v^{(m)}}{\partial x^{(m)}} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & a_4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & b_1 & b_2 & b_3 & b_4 \\ b_1 & b_2 & b_3 & b_4 & a_1 & a_2 & a_3 & a_4 \end{bmatrix} \mathbf{u}^{(m)} = \mathbf{B}^{(m)}(r,s)\mathbf{u}^{(m)}$$

Stiffness matrix

$$\mathbf{K}^{(m)} = \int_V \mathbf{B}^{(\mathrm{m})\mathrm{T}} \mathbf{C}^{(m)} \mathbf{B}^{(m)} dV^{(m)} = t\int_{-1}^{1}\int_{-1}^{1} \mathbf{B}^{(m)\mathrm{T}}(r,s)\mathbf{C}^{(m)}\mathbf{B}^{(m)}(r,s)\det\mathbf{J}drds$$

with $dV^{(m)} = \det\mathbf{J} \times tdrds$

Gauss Integration (Gaussian Quadrature)



$$\int_a^b f(x)dx = \sum_{i=1}^{n} w_i f(x_i)$$

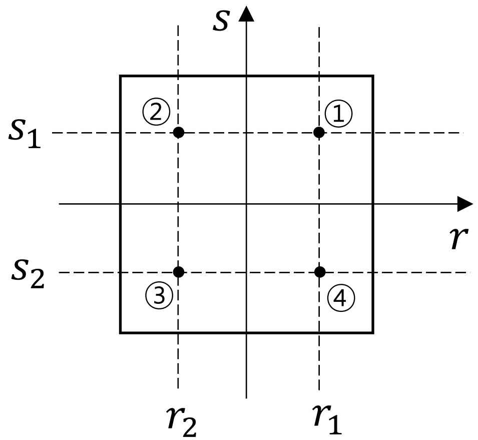
x_i: Gauss points, w_i: Weight factors

2-point Gauss integration in 2D

$$\int_{-1}^{1} f(x)\,dx = w_1 f(x_1) + w_2 f(x_2)$$

$$x_1 = -\frac{1}{\sqrt{3}},\ \ x_2 = \frac{1}{\sqrt{3}},\ \ \ w_1 = w_2 = 1$$

$$\int_{-1}^{1} x^2 dx = 1\times(-\frac{1}{\sqrt{3}})^2 + 1\times(\frac{1}{\sqrt{3}})^2 = \frac{2}{3}$$



$r_1 = s_1 = 1/\sqrt{3}$
$r_2 = s_2 = -1/\sqrt{3}$
$w_{ij} = 1$

Using “2x2 Gauss integration” in 2D,

$$\mathbf{K}^{(m)} = t\sum_{i=1}^{2}\sum_{j=1}^{2} w_{ij}\mathbf{B}^{(m)\mathrm{T}}(r_i, s_j)\mathbf{C}^{(m)}\mathbf{B}^{(m)}(r_i, s_j)\det\mathbf{J}(r_i, s_j)$$

$$= w_{11}\mathbf{K}'(r_1, s_1) + w_{12}\mathbf{K}'(r_1, s_2) + w_{21}\mathbf{K}'(r_2, s_1) + w_{22}\mathbf{K}'(r_2, s_2)$$

with $\mathbf{K}'(r,s) = t\,\mathbf{B}^{(m)\mathrm{T}}(r,s)\mathbf{C}^{(m)}\mathbf{B}^{(m)}(r,s)\det\mathbf{J}(r,s)$

Load vector

$$\mathbf{R} = \mathbf{R}_B + \mathbf{R}_S$$

$$\mathbf{R}_B = \sum_m \mathbf{R}_B^{(m)} \quad \text{with } \mathbf{R}_B^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)T} \mathbf{f}^{(m)} dV^{(m)} = t\int_{-1}^{1}\int_{-1}^{1} \mathbf{H}^{(m)\mathrm{T}}(r,s)\mathbf{f}^{(m)}(r,s) \det \mathbf{J} drds$$

$$\mathbf{R}_S = \sum_m \mathbf{R}_S^{(m)} \quad \text{with } \mathbf{R}_S^{(m)} = \int_{S_f^{(m)}} \mathbf{H}_S^{(m)T} \mathbf{f}^{S(m)} dS^{(m)}$$

$$\mathbf{R}_S^{(m)} = \int_{-1}^{1} \mathbf{H}_S^{(m)T}(r)\mathbf{f}^{S(m)}(r)\frac{\partial l}{\partial r} dr \quad \text{for surface load on } s = \pm 1$$

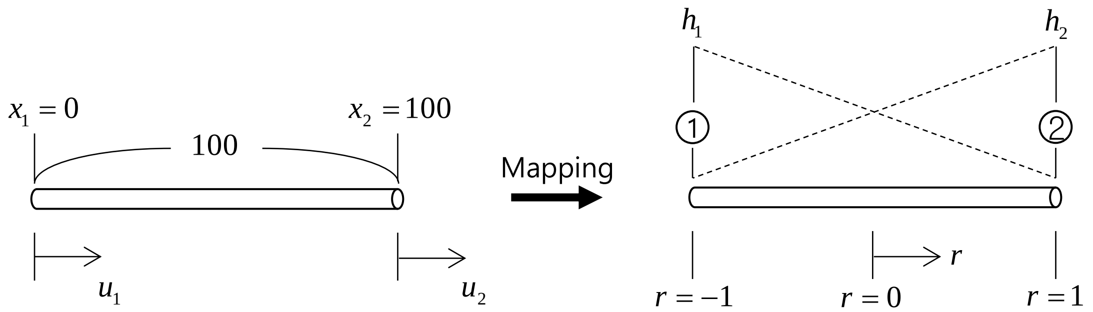
$$\mathbf{R}_S^{(m)} = \int_{-1}^{1} \mathbf{H}_S^{(m)T}(s)\mathbf{f}^{S(m)}(s)\frac{\partial l}{\partial s} ds \quad \text{for surface load on } r = \pm 1$$

Mass matrix

$$\mathbf{M} = \sum_m \mathbf{M}^{(m)}$$

$$\mathbf{M}^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \rho^{(m)} \mathbf{H}^{(m)} dV^{(m)} = t\int_{-1}^{1}\int_{-1}^{1} \mathbf{H}^{(m)\mathrm{T}}(r,s)\rho^{(m)}\mathbf{H}^{(m)}(r,s) \det \mathbf{J} drds$$

Example – Stiffness of a 2-node bar finite element



< Global coordinate > **< Natural coordinate >**

$$u^{(m)} = (1 - \frac{x}{100})u_1 + \frac{x}{100}u_2$$

$$\varepsilon_{xx} = \frac{du^{(m)}}{dx} = -\frac{1}{100}u_1 + \frac{1}{100}u_2 = \begin{bmatrix} -\frac{1}{100} & \frac{1}{100} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\mathbf{K}^{(m)} = \int_0^{100} \mathbf{B}^{(m)T} E \mathbf{B}^{(m)} dx = \frac{E}{100} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

Isoparametric procedure

$$x^{(m)} = \sum_{i=1}^{2} h_i x_i = h_1 x_1 + h_2 x_2, \quad x^{(m)} = x^{(m)}(r)$$

$$u^{(m)} = \sum_{i=1}^{2} h_i u_i = h_1 u_1 + h_2 u_2, \quad u^{(m)} = u^{(m)}(r) \qquad \text{with } h_1 = \frac{1}{2}(1-r),\ h_2 = \frac{1}{2}(1+r)$$

$$\varepsilon_{xx} = \frac{du^{(m)}}{dx^{(m)}}$$

$$\frac{dx^{(m)}}{dr} = \frac{d}{dr}(h_1 x_1 + h_2 x_2) = \frac{d}{dr}\left(h_1 \times 0 + h_2 \times 100\right) = \frac{d}{dr}\left(\frac{1}{2}(1+r)\times 100\right) = 50$$

$$\det \mathbf{J} = 50$$

$$\frac{du^{(m)}}{dr} = \frac{dx^{(m)}}{dr}\frac{du^{(m)}}{dx^{(m)}} = 50\frac{du^{(m)}}{dx^{(m)}}$$

$$\frac{du^{(m)}}{dx^{(m)}} = \frac{1}{50}\frac{du^{(m)}}{dr}$$

$$\frac{du^{(m)}}{dr} = \begin{bmatrix} \frac{dh_1}{dr} & \frac{dh_2}{dr} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\varepsilon_{xx} = \frac{du^{(m)}}{dx^{(m)}} = \frac{1}{50}\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \mathbf{B}^{(m)} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

$$\mathbf{B}^{(m)}(r) = \frac{1}{50}\begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

$$\mathbf{K}^{(m)} = \int_{-1}^{1} \mathbf{B}^{(m)T}(r) E \mathbf{B}^{(m)}(r) \det \mathbf{J} dr$$

$$= w_1 (\mathbf{B}^{(m)T} E \mathbf{B}^{(m)})\Big|_{r=-\frac{1}{\sqrt{3}}} \times 50 + w_2 (\mathbf{B}^{(m)T} E \mathbf{B}^{(m)})\Big|_{r=\frac{1}{\sqrt{3}}} \times 50$$

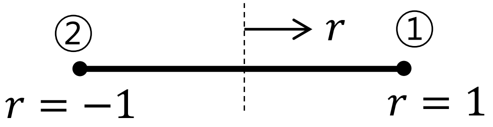
with $w_1 = w_2 = 1$

$$\mathbf{K}^{(m)} = 2(\mathbf{B}^{(m)T} E \mathbf{B}^{(m)}) \det \mathbf{J} = 2 \times \frac{1}{50} \begin{bmatrix} -\frac{1}{2} \\ \frac{1}{2} \end{bmatrix} E \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \frac{1}{50} \times 50 = \frac{E}{100} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

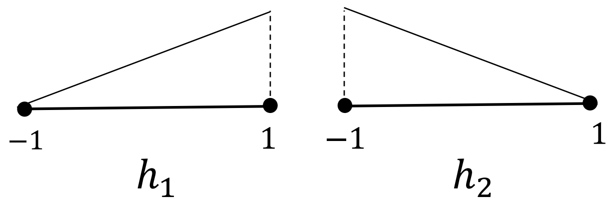
Higher-order Finite Elements

1D elements

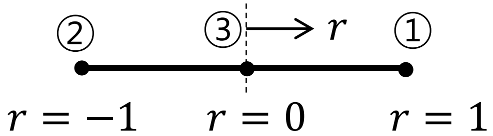
(1) 2-node bar element



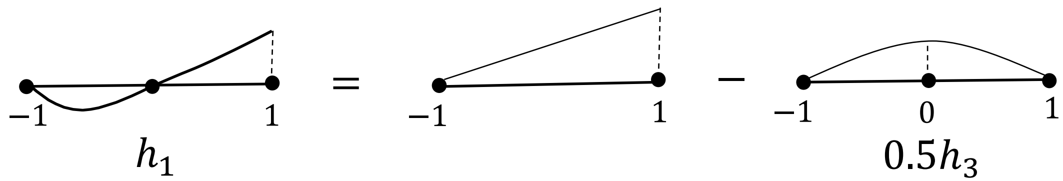
$$\begin{cases} h_1 = \frac{1}{2}(1+r) \\ h_2 = \frac{1}{2}(1-r) \end{cases}$$



(2) 3-node bar element

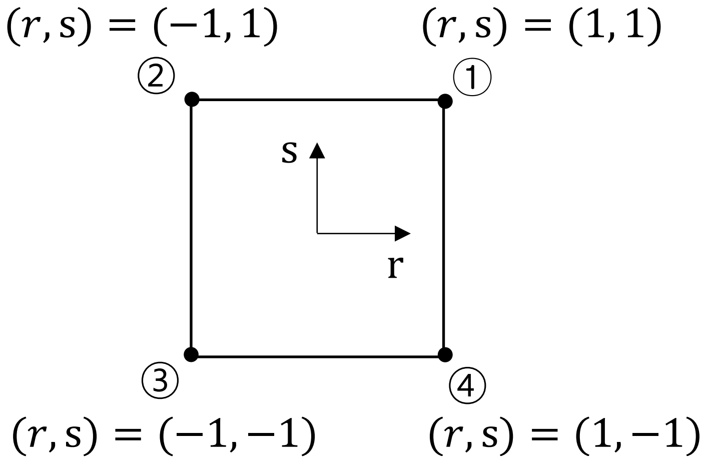


$$\begin{cases} h_1 = \frac{1}{2}(1+r) - \frac{1}{2}h_3 \\ h_2 = \frac{1}{2}(1-r) - \frac{1}{2}h_3 \\ h_3 = (1-\mathrm{r})(1+\mathrm{r}) \end{cases}$$



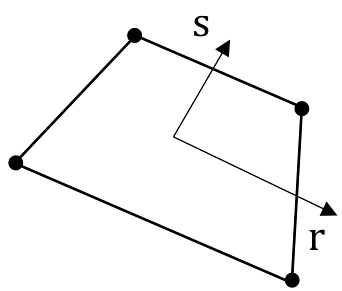
2D quadrilateral elements

(1) 4-node element



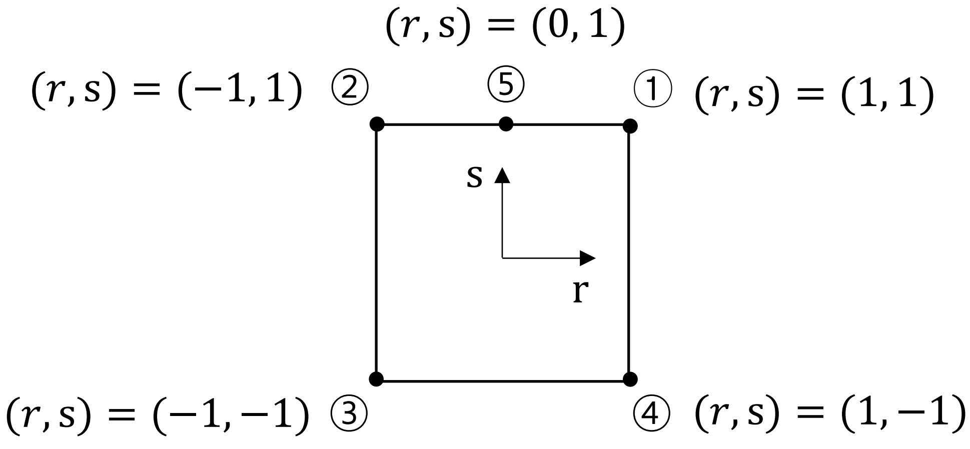
$$
\begin{cases}
h_1 = \frac{1}{4}(1+r)(1+s) \\
h_2 = \frac{1}{4}(1-r)(1+s) \\
h_3 = \frac{1}{4}(1-r)(1-s) \\
h_4 = \frac{1}{4}(1+r)(1-s)
\end{cases}
$$

< In the natural coordintate system >



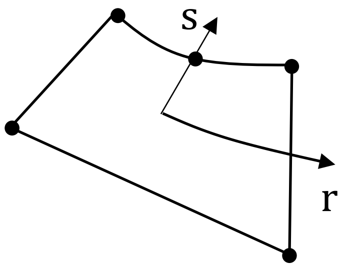
< In the global coordintate system >

(2) 5-node element

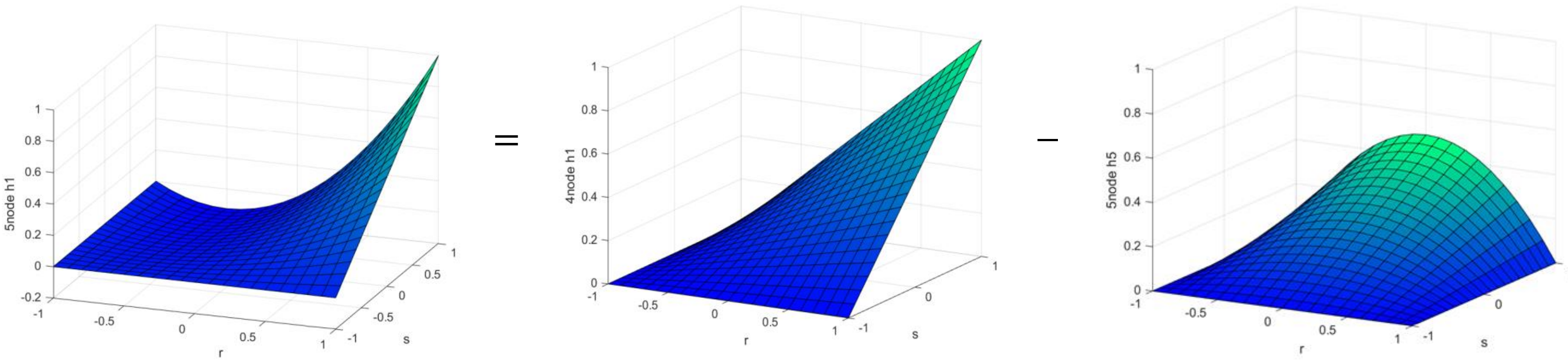


< In the natural coordintate system >

$$
\begin{cases}
h_1 = \frac{1}{4}(1+r)(1+s) - \frac{1}{2}h_5 \\
h_2 = \frac{1}{4}(1-r)(1+s) - \frac{1}{2}h_5 \\
h_3 = \frac{1}{4}(1-r)(1-s) \\
h_4 = \frac{1}{4}(1+r)(1-s) \\
h_5 = \frac{1}{2}(1-r^2)(1+s)
\end{cases}
$$

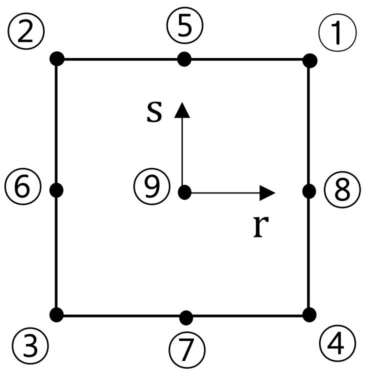


< In the global coordintate system >

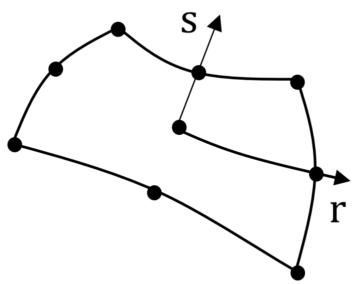


$$h_1 = \frac{1}{4}(1+r)(1+s) - \frac{1}{2}h_5 \qquad \frac{1}{4}(1+r)(1+s) \qquad \frac{1}{2}h_5$$

(3) 9-node element



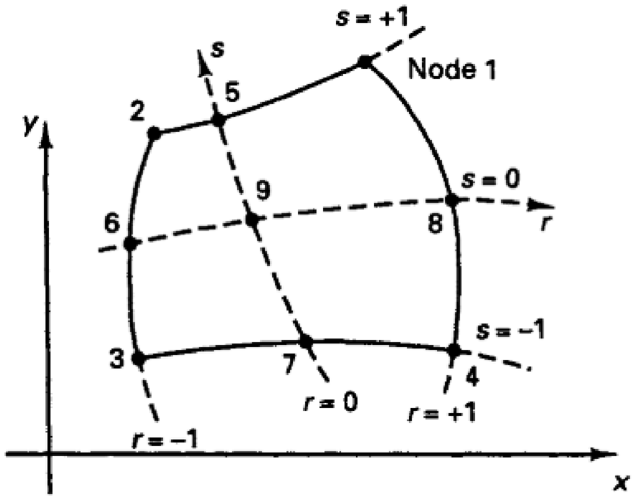
< In the natural coordintate system >



< In the global coordintate system >

$$
\begin{cases}
h_1 = \frac{1}{4}(1+r)(1+s) - \frac{1}{2}h_5 - \frac{1}{2}h_8 - \frac{1}{4}h_9 \\
h_2 = \frac{1}{4}(1-r)(1+s) - \frac{1}{2}h_5 - \frac{1}{2}h_6 - \frac{1}{4}h_9 \\
h_3 = \frac{1}{4}(1-r)(1-s) - \frac{1}{2}h_6 - \frac{1}{2}h_7 - \frac{1}{4}h_9 \\
h_4 = \frac{1}{4}(1+r)(1-s) - \frac{1}{2}h_7 - \frac{1}{2}h_8 - \frac{1}{4}h_9 \\
h_5 = \frac{1}{2}(1-r^2)(1+s) - \frac{1}{2}h_9 \\
h_6 = \frac{1}{2}(1-s^2)(1-r) - \frac{1}{2}h_9 \\
h_7 = \frac{1}{2}(1-r^2)(1-s) - \frac{1}{2}h_9 \\
h_8 = \frac{1}{2}(1-s^2)(1+r) - \frac{1}{2}h_9 \\
h_9 = (1-r^2)(1-s^2)
\end{cases}
$$

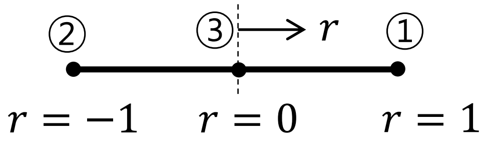
4 to 9 variable-number-nodes two-dimensional elements



		Include only if node i is defined				
		$i = 5$	$i = 6$	$i = 7$	$i = 8$	$i = 9$
$h_1 =$	$\frac{1}{4}(1 + r)(1 + s)$	$-\frac{1}{2}h_5$			$-\frac{1}{2}h_8$	$-\frac{1}{4}h_9$
$h_2 =$	$\frac{1}{4}(1 - r)(1 + s)$	$-\frac{1}{2}h_5$	$-\frac{1}{2}h_6$			$-\frac{1}{4}h_9$
$h_3 =$	$\frac{1}{4}(1 - r)(1 - s)$		$-\frac{1}{2}h_6$	$-\frac{1}{2}h_7$		$-\frac{1}{4}h_9$
$h_4 =$	$\frac{1}{4}(1 + r)(1 - s)$			$-\frac{1}{2}h_7$	$-\frac{1}{2}h_8$	$-\frac{1}{4}h_9$
$h_5 =$	$\frac{1}{2}(1 - r^2)(1 + s)$					$-\frac{1}{2}h_9$
$h_6 =$	$\frac{1}{2}(1 - s^2)(1 - r)$					$-\frac{1}{2}h_9$
$h_7 =$	$\frac{1}{2}(1 - r^2)(1 - s)$					$-\frac{1}{2}h_9$
$h_8 =$	$\frac{1}{2}(1 - s^2)(1 + r)$					$-\frac{1}{2}h_9$
$h_9 =$	$(1 - r^2)(1 - s^2)$					

(b) Interpolation functions

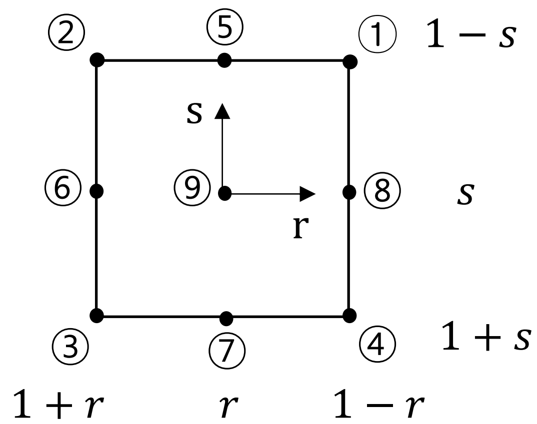
Note: Alternative approach



$$h_1 = C_1 r(1+r), \quad h_1 = 1 \text{ when } \mathrm{r} = 1 \quad \rightarrow \quad C_1 = \frac{1}{2}$$

$$h_2 = C_2 r(1-r), \quad h_2 = 1 \text{ when } \mathrm{r} = -1 \quad \rightarrow \quad C_2 = -\frac{1}{2}$$

$$h_1 = C_3(1+r)(1-r), \quad h_3 = 1 \text{ when } \mathrm{r} = 0 \quad \rightarrow \quad C_3 = 1$$



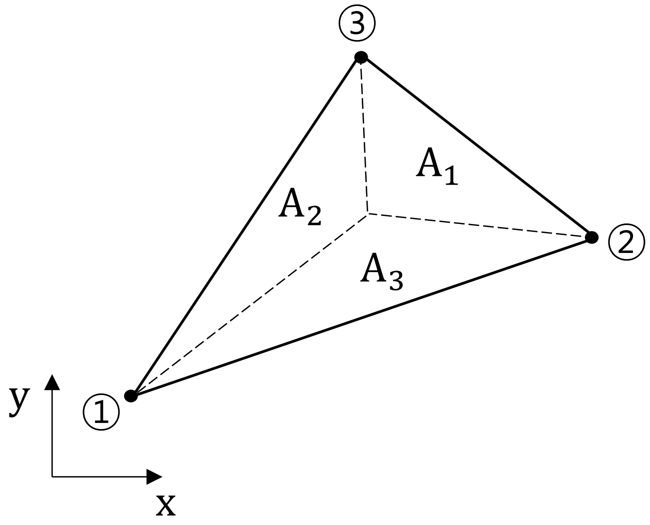
$$h_1 = C_1 r(1+r)s(1+s) \quad \rightarrow \quad C_1 = \frac{1}{4}$$

$$h_2 = C_2 r(1-r)s(1+s) \quad \rightarrow \quad C_2 = \frac{1}{4}$$

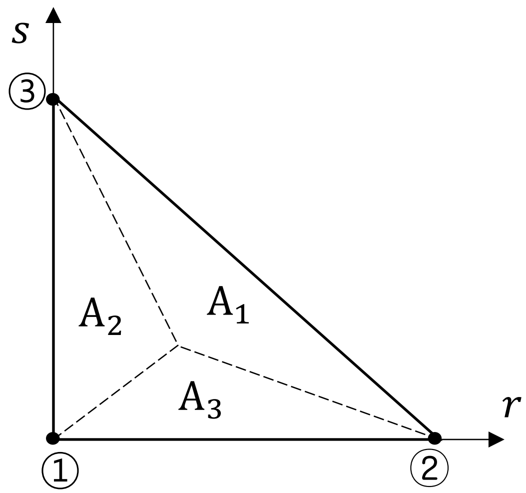
$$\vdots$$

$$h_9 = C_9(1-r^2)(1-s^2) \rightarrow \quad C_9 = 1$$

2D triangular elements



< In the global coordintae system >



< In the natural coordintae system >

Area coordinate system

$$A_1 + A_2 + A_3 = A \quad \rightarrow \quad \frac{A_1}{A} + \frac{A_2}{A} + \frac{A_3}{A} = 1$$

$$L_1 + L_2 + L_3 = 1 \quad \text{with} \quad L_i = \frac{A_i}{A}$$

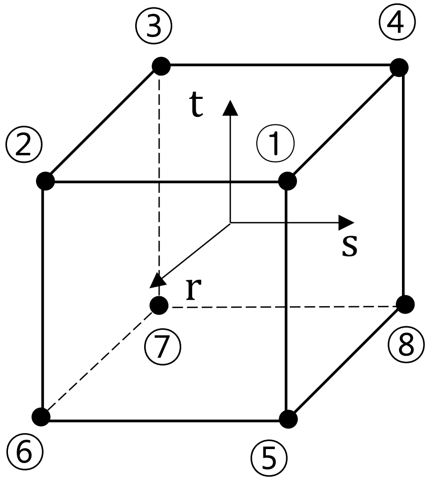
→

$$L_2 = h_2 = A_2 / A = r$$

$$L_3 = h_3 = A_3 / A = s$$

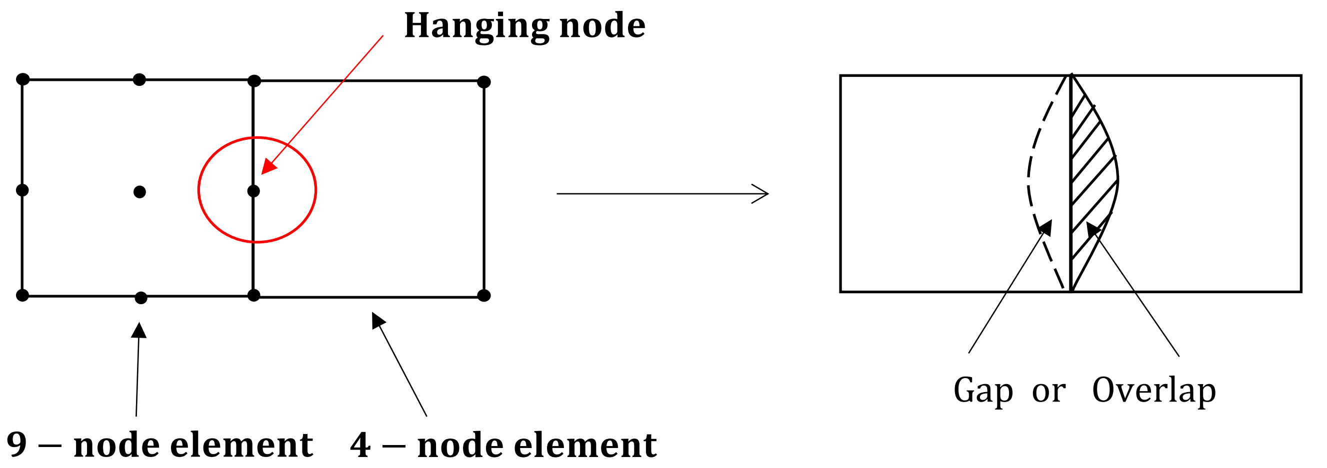
$$L_1 = h_1 = A_1 / A = 1 - r - s$$

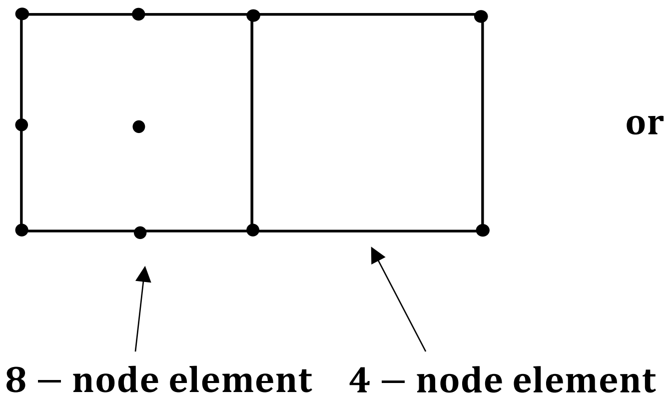
3D elements



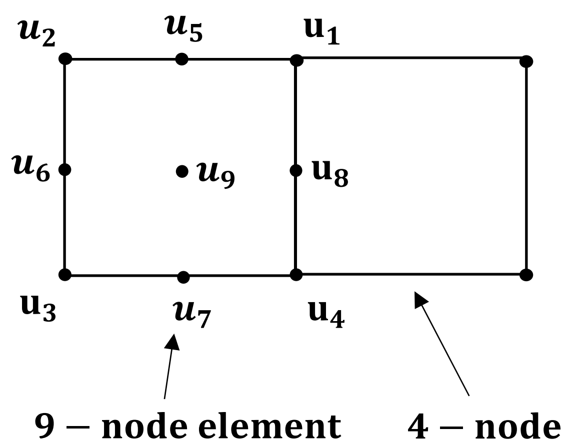
$$
\begin{cases}
h_1 = \frac{1}{8}(1+r)(1+s)(1+t) \\
h_2 = \frac{1}{8}(1-r)(1+s)(1+t) \\
h_3 = \frac{1}{8}(1-r)(1-s)(1+t) \\
h_4 = \frac{1}{8}(1+r)(1-s)(1+t) \\
h_5 = \frac{1}{8}(1+r)(1+s)\,(1-t) \\
h_6 = \frac{1}{8}(1-r)(1+s)(1-t) \\
h_7 = \frac{1}{8}(1-r)(1-s)\,(1-t) \\
h_8 = \frac{1}{8}(1+r)(1-s)(1-t)
\end{cases}
$$

Compatibility between finite elements





or



with constraint

$$u_8 = \frac{1}{2}(u_1 + u_4)$$

Gauss integrations

TABLE 5.7 *Gauss numerical integrations over quadrilateral domains*

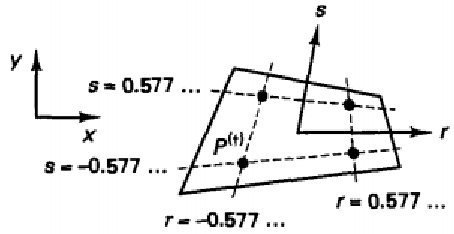
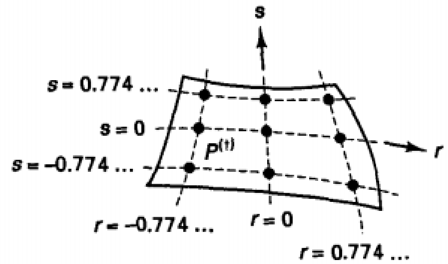
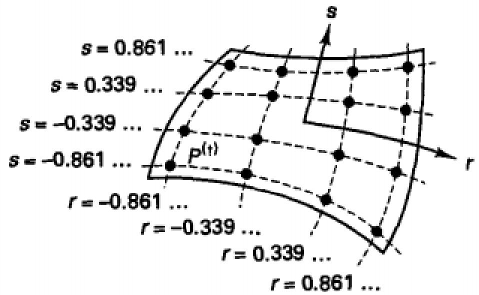
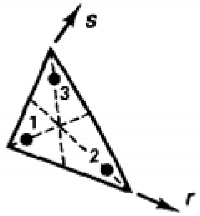
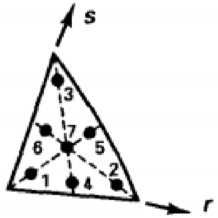
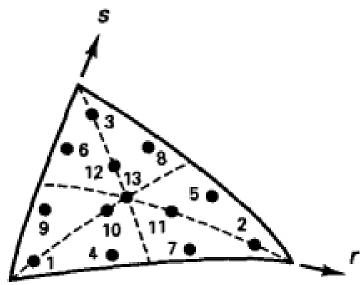
Integration order	Degree of precision	Location of integration points
2 × 2	3	
3 × 3	5	
4 × 4	7	

TABLE 5.6 *Sampling points and weights in Gauss-Legendre numerical integration (interval −1 to +1)*

n	r_i			α_i		
1	0.	(15 zeros)		2.	(15 zeros)	
2	±0.57735	02691	89626	1.00000	00000	00000
3	±0.77459	66692	41483	0.55555	55555	55556
	0.00000	00000	00000	0.88888	88888	88889
4	±0.86113	63115	94053	0.34785	48451	37454
	±0.33998	10435	84856	0.65214	51548	62546
5	±0.90617	98459	38664	0.23692	68850	56189
	±0.53846	93101	05683	0.47862	86704	99366
	0.00000	00000	00000	0.56888	88888	88889
6	±0.93246	95142	03152	0.17132	44923	79170
	±0.66120	93864	66265	0.36076	15730	48139
	±0.23861	91860	83197	0.46791	39345	72691

TABLE 5.8 *Gauss numerical integrations over triangular domains* $\left[\iint F\,dr\,ds = \frac{1}{2}\sum w_i F(r_i, s_i)\right]$

Integration order	Degree of precision	Integration points	*r*-coordinates	*s*-coordinates	Weights
3-point	2		$r_1 = 0.16666\ 66666\ 667$ $r_2 = 0.66666\ 66666\ 667$ $r_3 = r_1$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$	$w_1 = 0.33333\ 33333\ 333$ $w_2 = w_1$ $w_3 = w_1$
7-point	5		$r_1 = 0.10128\ 65073\ 235$ $r_2 = 0.79742\ 69853\ 531$ $r_3 = r_1$ $r_4 = 0.47014\ 20641\ 051$ $r_5 = r_4$ $r_6 = 0.05971\ 58717\ 898$ $r_7 = 0.33333\ 33333\ 333$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$ $s_4 = r_6$ $s_5 = r_4$ $s_6 = r_4$ $s_7 = r_7$	$w_1 = 0.12593\ 91805\ 448$ $w_2 = w_1$ $w_3 = w_1$ $w_4 = 0.13239\ 41527\ 885$ $w_5 = w_4$ $w_6 = w_4$ $w_7 = 0.225$
13-point	7		$r_1 = 0.06513\ 01029\ 022$ $r_2 = 0.86973\ 97941\ 956$ $r_3 = r_1$ $r_4 = 0.31286\ 54960\ 049$ $r_5 = 0.63844\ 41885\ 698$ $r_6 = 0.04869\ 03154\ 253$ $r_7 = r_5$ $r_8 = r_4$ $r_9 = r_6$ $r_{10} = 0.26034\ 59660\ 790$ $r_{11} = 0.47930\ 80678\ 419$ $r_{12} = r_{10}$ $r_{13} = 0.33333\ 33333\ 333$	$s_1 = r_1$ $s_2 = r_1$ $s_3 = r_2$ $s_4 = r_6$ $s_5 = r_4$ $s_6 = r_5$ $s_7 = r_6$ $s_8 = r_5$ $s_9 = r_4$ $s_{10} = r_{10}$ $s_{11} = r_{10}$ $s_{12} = r_{11}$ $s_{13} = r_{13}$	$w_1 = 0.05334\ 72356\ 088$ $w_2 = w_1$ $w_3 = w_1$ $w_4 = 0.07711\ 37608\ 903$ $w_5 = w_4$ $w_6 = w_4$ $w_7 = w_4$ $w_8 = w_4$ $w_9 = w_4$ $w_{10} = 0.17561\ 52574\ 332$ $w_{11} = w_{10}$ $w_{12} = w_{10}$ $w_{13} = -0.14957\ 00444\ 677$

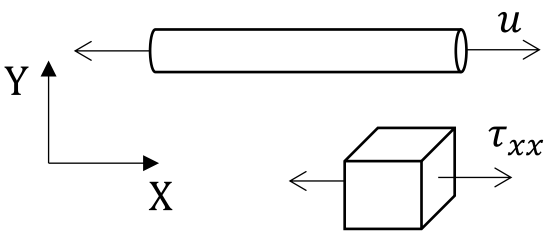
$$\mathbf{K}^{(m)} = \int_V \mathbf{B}^{(m)T}\mathbf{C}^{(m)}\mathbf{B}^{(m)}\,dV^{(m)} = t\int_0^1\int_0^{1-s}\mathbf{B}^{(m)T}(r,s)\mathbf{C}^{(m)}\mathbf{B}^{(m)}(r,s)\det\mathbf{J}\,dr\,ds$$

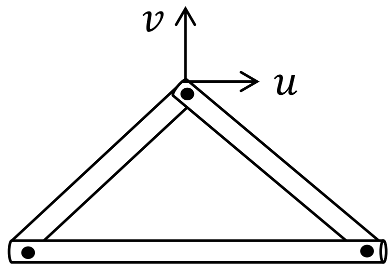
Finite Elements

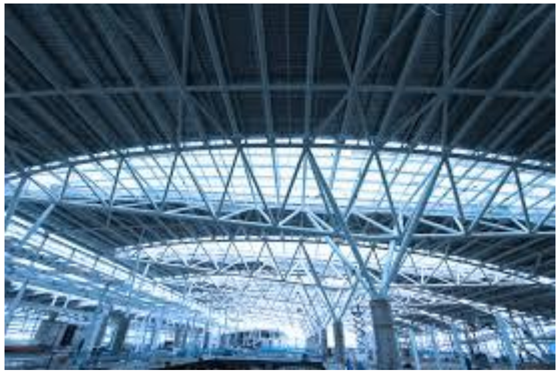
1D elements

- Truss elements or Bar elements (stretching)

$$\begin{cases} \textbf{in 1D}: u \\ \textbf{in 2D}: u, v \\ \textbf{in 3D}: u, v, w \end{cases}$$

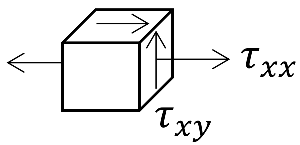


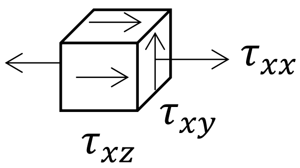


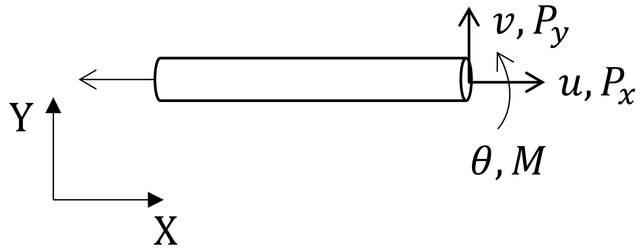


- Beam elements (stretching, bending, shearing, twisting)

$$\begin{cases} \textbf{in 2D}: u, v, \theta \\ \textbf{in 3D}: u, v, w, \theta_x, \theta_y, \theta_z \end{cases}$$

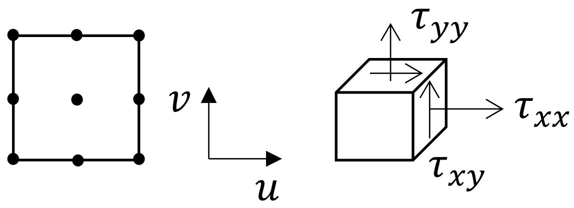






2D elements

- Plane stress element: u, v

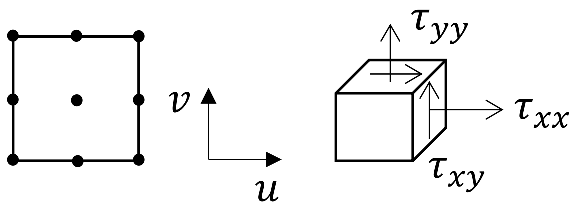


$$\boldsymbol{\tau}\cdot\mathbf{n}_z = 0 \rightarrow \quad \tau_{zz} = \tau_{xz} = \tau_{yz} = 0$$

$$\tau_{xx},\ \tau_{yy},\ \tau_{xy} \neq 0$$

$$\Rightarrow \begin{bmatrix}\tau_{xx}\\ \tau_{yy}\\ \tau_{xy}\end{bmatrix} = \frac{E}{1-\nu^2}\begin{bmatrix}1 & \nu & 0\\ \nu & 1 & 0\\ 0 & 0 & 1-\nu\end{bmatrix}\begin{bmatrix}\varepsilon_{xx}\\ \varepsilon_{yy}\\ \gamma_{xy}\end{bmatrix}$$

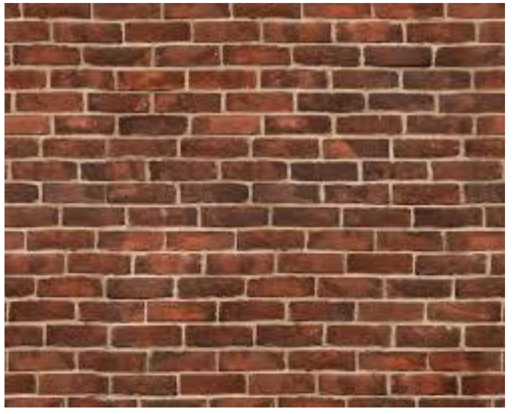
- Plane strain element: u, v



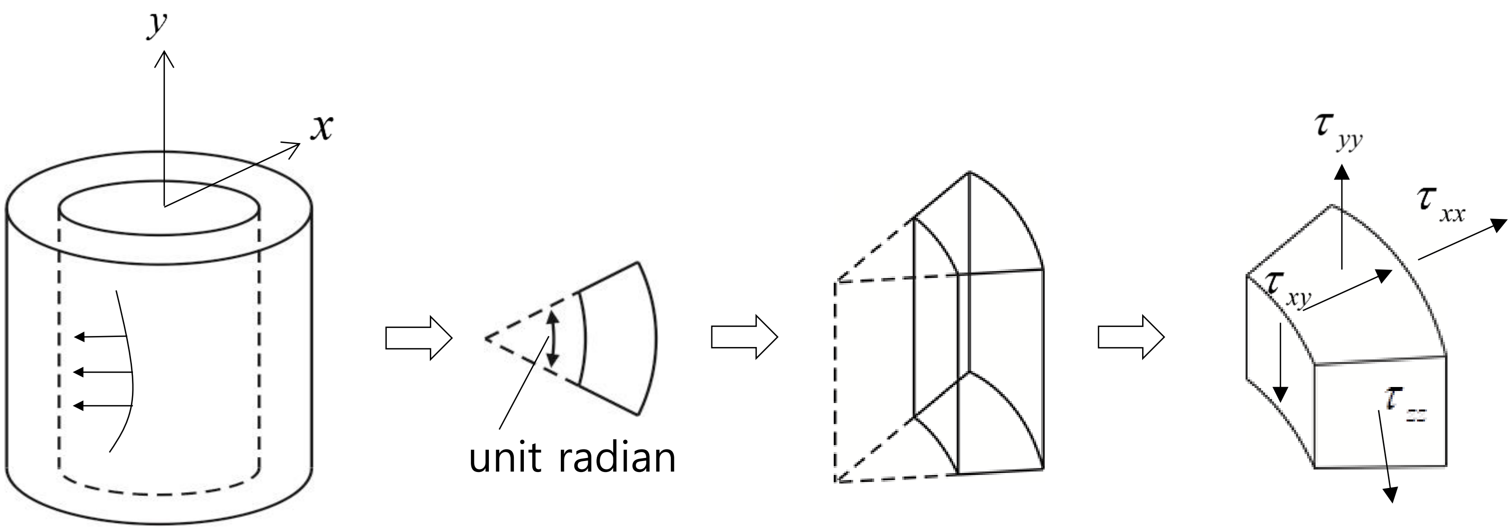
$$\boldsymbol{\varepsilon}\cdot\mathbf{n}_z = 0 \rightarrow \quad \varepsilon_{zz} = \gamma_{xz} = \gamma_{yz} = 0$$

$$\varepsilon_{xx},\ \varepsilon_{yy},\ \gamma_{xy} \neq 0$$

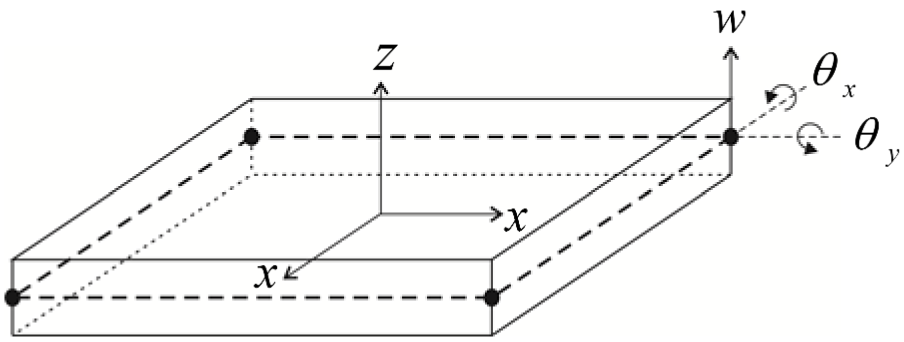
$$\Rightarrow \begin{bmatrix}\tau_{xx}\\ \tau_{yy}\\ \tau_{xy}\end{bmatrix} = \frac{E(1-\nu)}{(1+\nu)(1-2\nu)}\begin{bmatrix}1 & \dfrac{\nu}{1-\nu} & 0\\ \dfrac{\nu}{1-\nu} & 1 & 0\\ 0 & 0 & \dfrac{1-2\nu}{1-\nu}\end{bmatrix}\begin{bmatrix}\varepsilon_{xx}\\ \varepsilon_{yy}\\ \gamma_{xy}\end{bmatrix}$$



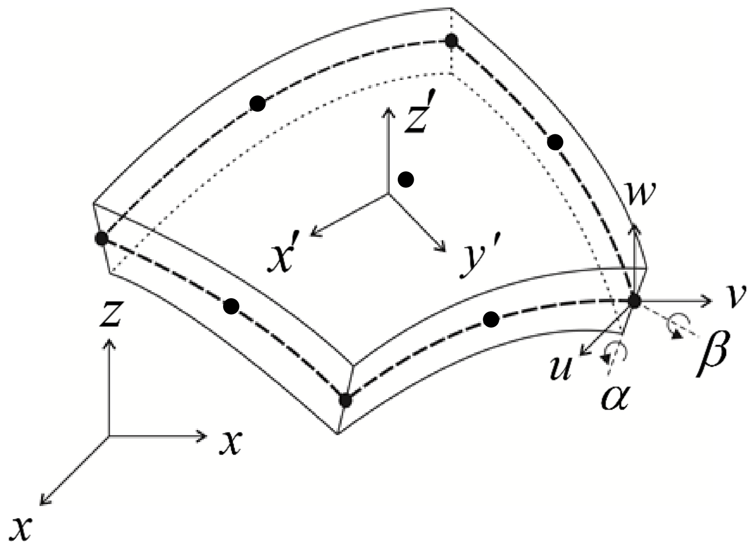

- Axisymmetric elements: u, v
  - ✓ Geometry, loading and boundary conditions are axisymmetric: $\tau_{yz} = \tau_{zx} = 0$



- Plate elements (bending, shearing): v, θ_x, θ_y
  - ✓ 2-dimensinal extension of beams
  - ✓ used in analysis of slab or deck structures
  - ✓ $\tau_{zz} = 0$, other stress components are not zero.



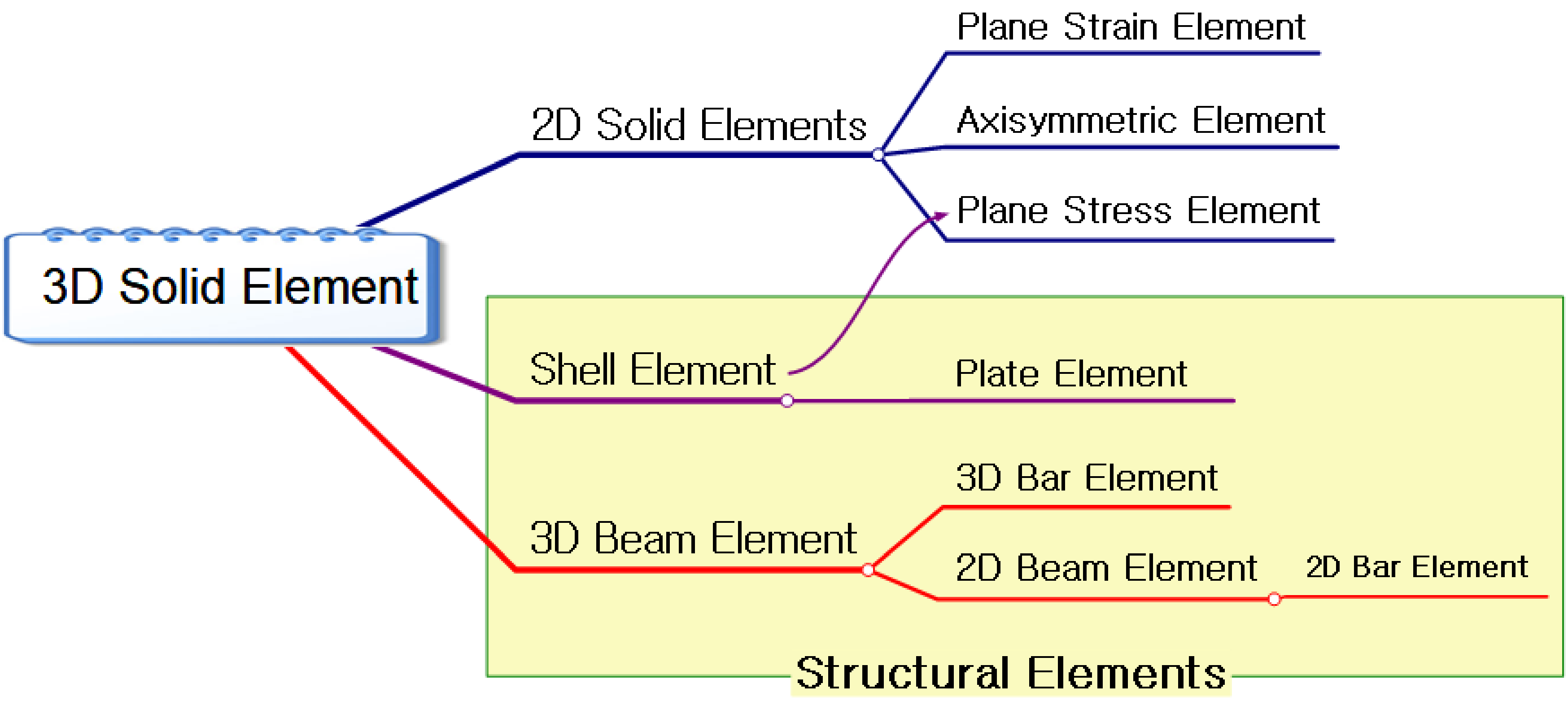
- Shell elements (stretching, bending, shearing): u, v, w, α, β
  - ✓ used in analysis of general curved thin wall structures like domes, hulls, etc.
  - ✓ x, y, z: global Cartesian coordinates, and x', y', z': locally defined coordinates
  - ✓ Shell elements have 5 degrees of freedom per node.
  - ✓ α, β: locally defined rotations at nodes
  - ✓ $\tau_{z'z'} = 0$, other stress components are not zero.



3D elements (Most general element)

- 3D solid elements: 8-node hexahedral element, 4-node tetrahedral element

Modeling Hierarchy Map



Selection of finite elements

(1) Scale and complexity of the problem

(2) Scale, interest and objective of behavior

(3) Availability of computational resources and time

7. Introduction to Variational Analysis

Functional

A real valued function F whose domain is a set of functions is called a functional

Function $y(x)$

$$x \longrightarrow \boxed{y(x)} \longrightarrow y$$

real number mapping real number

Ex) $y = x + 1,\ \ y = \sin(x),\ \ldots\ldots$

Functional $F(y)$

$$y(x) \longrightarrow \boxed{F(y)} \longrightarrow F$$

function mapping real number

Functional F is a mapping from function(s) to a real number.

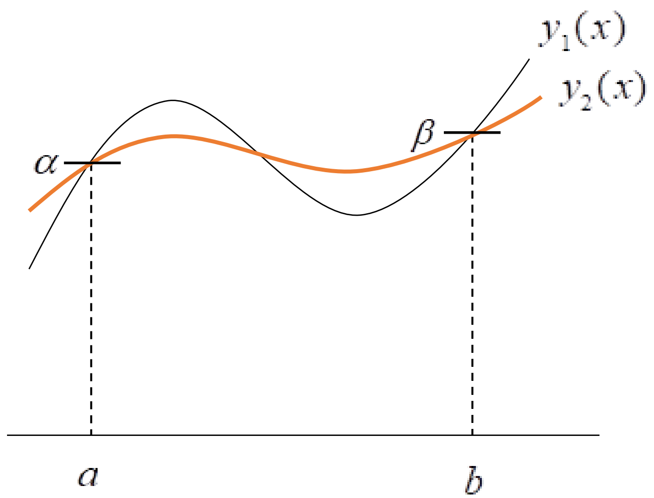
Ex) $F = \int_a^b f(y)dx = \int_a^b y dx$

$$F = \int_a^b f(y')dx = \int_a^b \sqrt{1+y'^2}\,dx$$

$$F = \int_a^b f(y,y')dx = \int_a^b \frac{y'}{y}dx$$

Note: A bilinear operator $a(\cdot,\cdot)$ and a linear operator $(\mathbf{f},\cdot)$ can produce functionals, and the vector functions are the input for them.

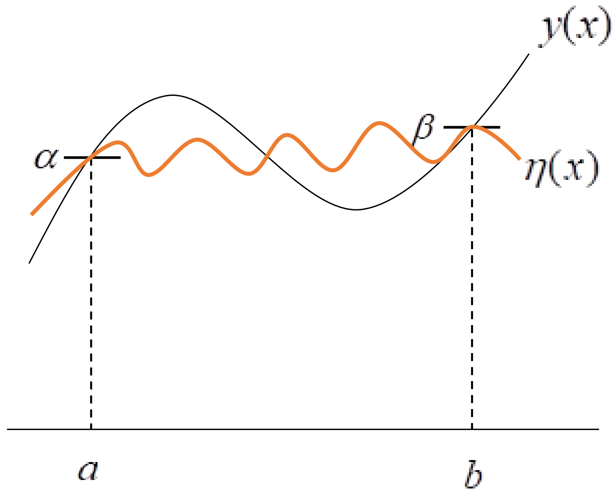
Variation of Functional F



"y_1 and y_2 can be the inputs"

Let f be a function of three variables, $f = f(y, y', x)$. The functional is $F = \int_a^b f(y, y', x)dx$ and the BCs are given as $y(a) = \alpha$ and $y(b) = \beta$.

In variational analysis, we want to find a function y which will make the functional F an extremum, or at least give a stationary value.



When the input function $y(x)$ is changed into $y(x)+\varepsilon\eta(x)$ with a small value ε, the term $\varepsilon\eta(x)$ is called the variation of $y(x)$. The variation is denoted by δy (change of the input function).

The variation of functional F due to δy can be expanded through Taylor series

$$\begin{aligned}\Delta F &= F(y+\delta y, y'+\delta y', x) - F(y, y', x) \\ &= F(y, y', x) + \frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y' + \cdots\cdots - F(y, y', x) \\ &= \frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y' + \cdots\cdots\end{aligned}$$

in which

$$\delta F = \frac{\partial F}{\partial y}\delta y + \frac{\partial F}{\partial y'}\delta y'$$ → The leading term is called the first order variation.

The functional $F = \int_a^b f(y, y', x)dx$ is stationary if and only if its (first) variation vanishes for every permissible variation δy.

$\boxed{\delta F = 0}$ → We can find a function that maximizes or minimizes the functional.

Some Rules

$$\delta(F_1+F_2)=\delta F_1+\delta F_2, \qquad \delta(F_1F_2)=F_1\delta F_2+F_2\delta F_1$$

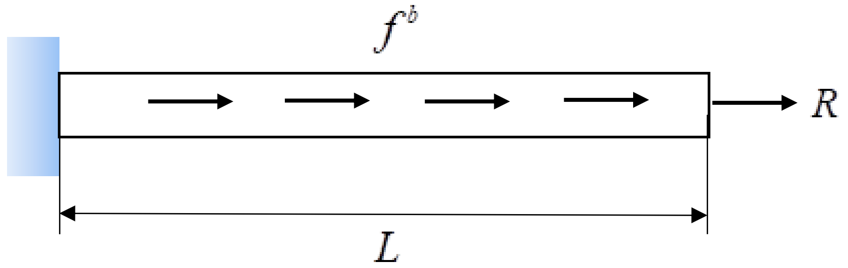
$$\delta\left(\frac{F_1}{F_2}\right)=\frac{F_2\delta F_1-F_1\delta F_2}{F_2^2}, \qquad \delta(F^n)=nF^{n-1}\delta F$$

Note: " δ " and " $\frac{d}{dx}$ " can be calculated with the same rules even though their meaning is completely different.

$$\frac{d}{dx}\delta y=\delta\left(\frac{dy}{dx}\right)$$: " δ " and " $\frac{d}{dx}$ " commutes.

$$\int_a^b \delta y dx=\delta\int_a^b y dx$$: " δ " and " $\int_a^b dx$ " commutes.

1D Bar Problem



Strong form of G.E.

$$EA\frac{d^2u}{dx^2} + f^b = 0, \quad 0 \le x \le L \quad \text{(equilibrium equation)}$$

$$u(0) = 0 \quad \text{(displacement B.C.)}$$

$$EA\frac{du}{dx}\bigg|_{x=L} = R \quad \text{(force B.C.)}$$

Weak form (PVW) of G.E.

$$\int_0^L \frac{d\delta u}{dx} EA \frac{du}{dx} dx = \int_0^L f^b \delta u dx + R\delta u\big|_{x=L}$$

The virtual displacement δu is the variation of u.

$$\int_0^L \frac{d\delta u}{dx} EA \frac{du}{dx} dx - \int_0^L f^b \delta u dx - R\delta u\big|_{x=L} = 0$$

$$\delta\left[\frac{1}{2}\int_0^L EA\left(\frac{du}{dx}\right)^2 dx - \int_0^L f^b u dx - Ru\big|_{x=L}\right] = 0$$

Note: $\delta\left[\frac{1}{2}\int_0^L EA\left(\frac{du}{dx}\right)^2 dx\right] = \frac{1}{2}\int_0^L EA\delta\left[\left(\frac{du}{dx}\right)^2\right] dx = \frac{1}{2}\int_0^L EA2\frac{du}{dx}\frac{d\delta u}{dx} dx = \int_0^L \frac{d\delta u}{dx} EA \frac{du}{dx} dx$

→ $\delta\Pi = 0$ with a functional $\Pi(u) = \frac{1}{2}\int_0^L EA\left(\frac{du}{dx}\right)^2 dx - \int_0^L f^b u dx - Ru\big|_{x=L}$

(potential energy) = (strain energy potential) - (force potential)

From PVW, we can obtain the principle of minimum potential energy.

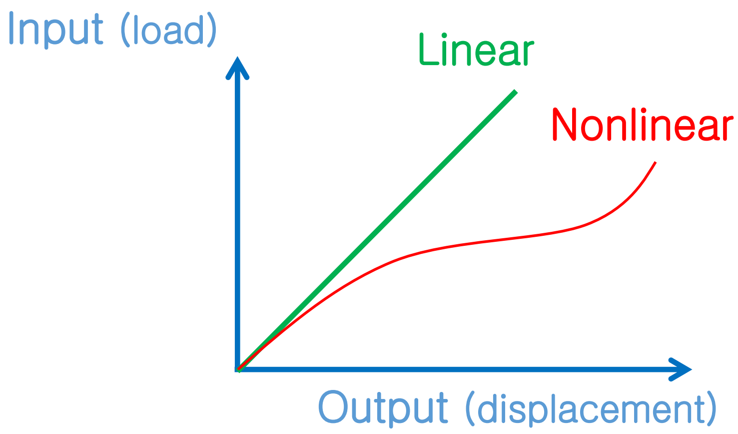
$\delta\Pi = 0$: Principle of minimum potential energy

- ✓ $\mathbf{u} = \min \Pi(\mathbf{w}), \ \mathbf{w} \in V$
- ✓ Find $\mathbf{u} \in V$ such that $\Pi(\mathbf{u}) < \Pi(\mathbf{w})$ for $\forall \mathbf{w} \in V$

8. Introduction to Nonlinear Analysis

Almost all real phenomena are basically nonlinear. In other words, the relationship between inputs and output is not directly proportional.

Linear Analysis



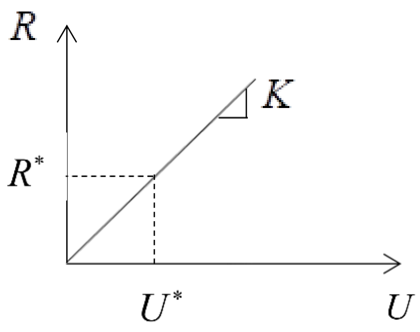
Assumptions for linear analysis

- Small displacements
- Small strain: $\varepsilon_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i})$
- Linear elastic material

Load-displacement relation is directly proportional.

$KU^* = R^* \rightarrow U^* = K^{-1}R^*$ in a single DOF system,

$\mathbf{KU}^* = \mathbf{R}^* \rightarrow \mathbf{U}^* = \mathbf{K}^{-1}\mathbf{R}^*$ in a multi DOFs system.



The principle of superposition is applicable.

$\mathbf{KU}^a = \mathbf{R}^a$ and $\mathbf{KU}^b = \mathbf{R}^b \rightarrow \mathbf{K}(\mathbf{U}^a + \mathbf{U}^b) = (\mathbf{R}^a + \mathbf{R}^b)$

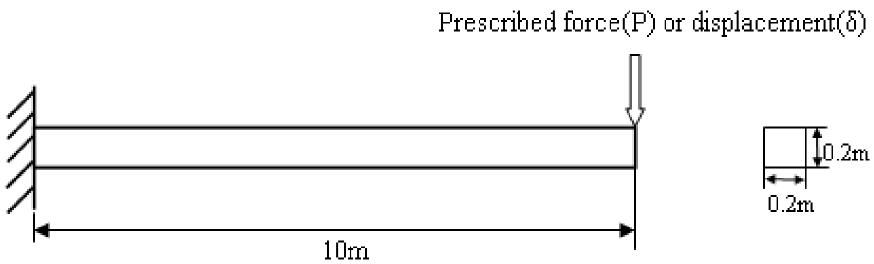
Nonlinear Analysis

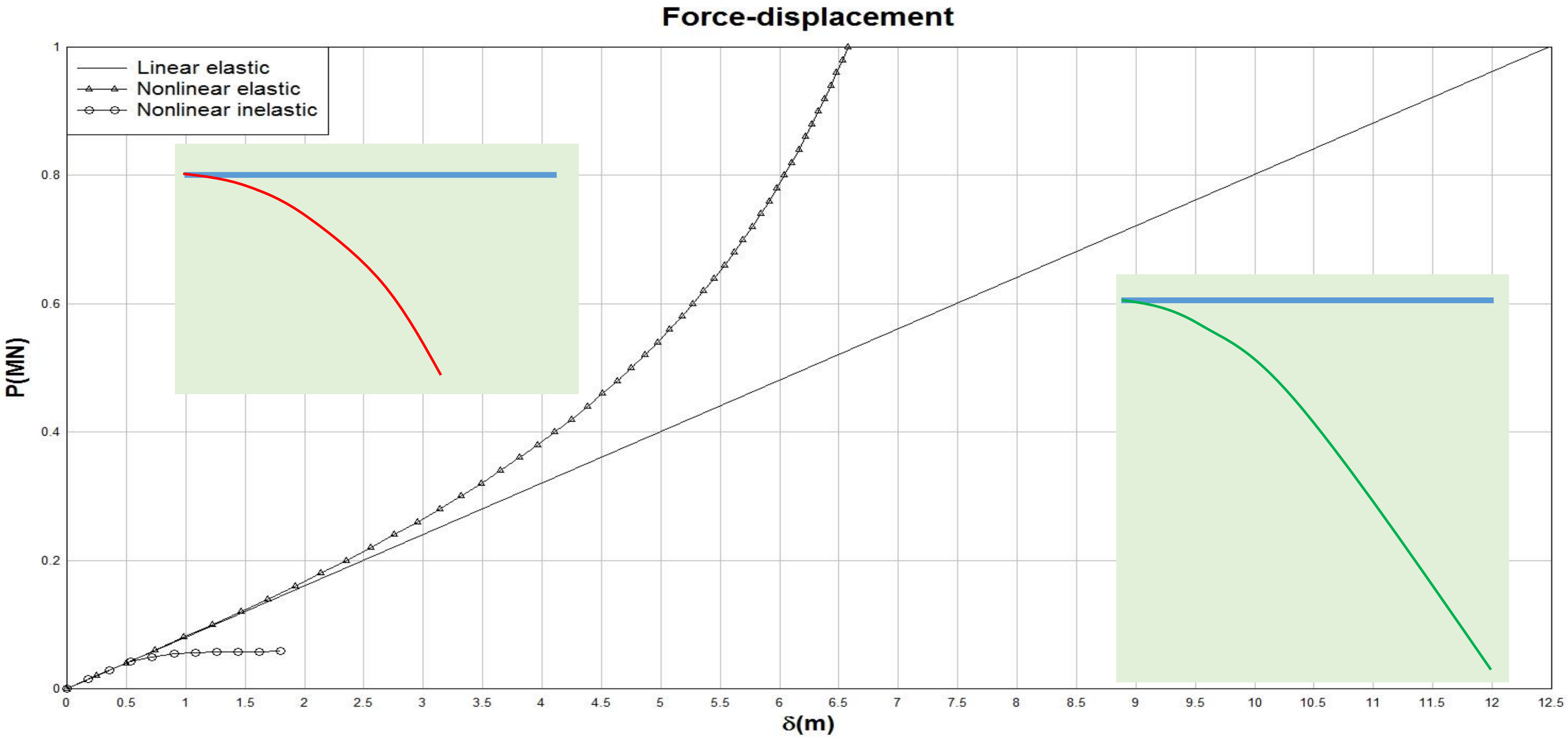
If any of the assumptions about "Linear Analysis" do not hold, we are faced with a situation where we have to analyze nonlinear phenomena.

Classification

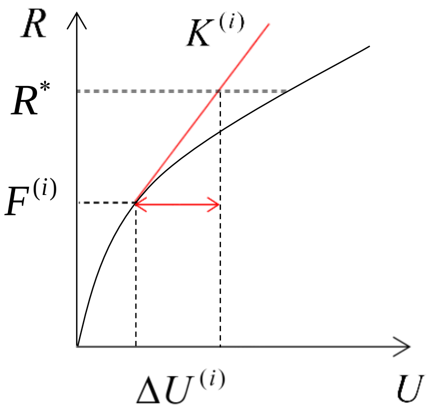
- Kinematic nonlinear analysis: Large displacements & small strain, large displacements & large strain)
- Material nonlinear analysis: Inelastic material (plastic material, nonlinear elastic material, ...)
- Kinematic + material nonlinear analysis
- Contact analysis

Example – Cantilever Beam Problem





Load-displacement relation



Incremental equilibrium equation (at $U^{(i)}$)

$$K^{(i)}\Delta U^{(i)} = R^* - F^{(i)}$$

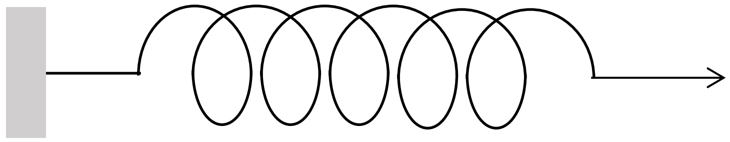
with $K^{(i)}$: tangential stiffness (known)

R^*: external force (known)

$F^{(i)}$: internal force (known)

$\Delta U^{(i)}$: incremental displacement (unknown)

Example – Nonlinear Spring



Stiffness: $K = \dfrac{1}{2\sqrt{U+1}}$, Internal force: $F = \sqrt{U+1} - 1$ (We do not know its inverse relation.)

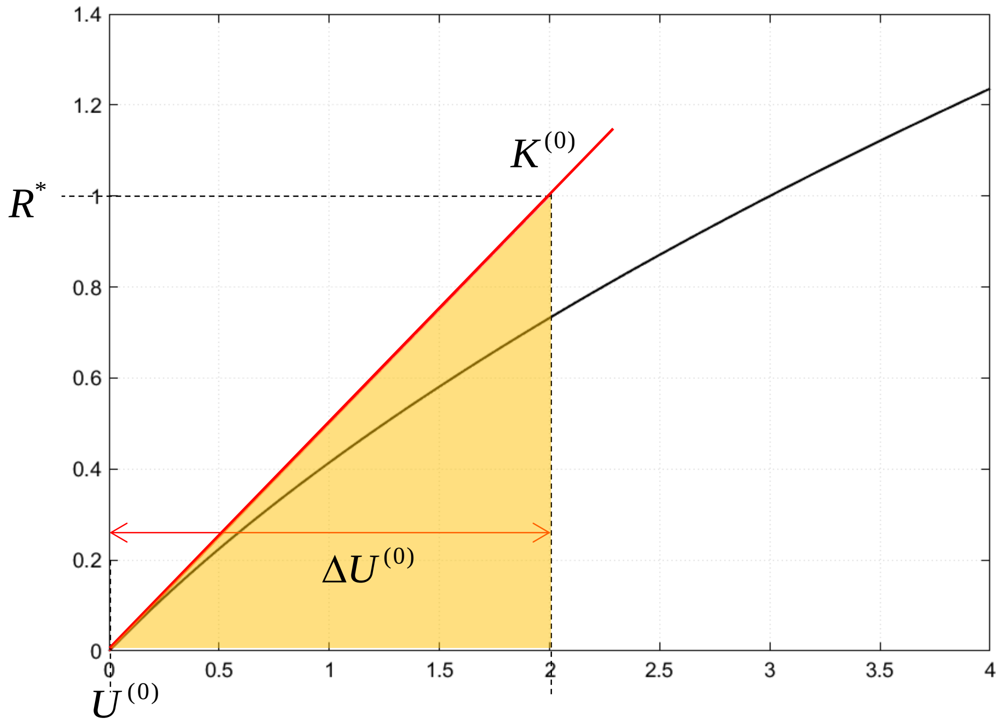
When $R^* = 1$, find U^*. (Exact solution: $U^* = 3$)

Newton-Raphson method: A root-finding algorithm which produces successively better approximations to the roots of a real-valued function.

Step 0 $i = 0: \; K^{(0)} \Delta U^{(0)} = R^* - F^{(0)}$ with $K = \dfrac{1}{2\sqrt{U+1}}, \; F = \sqrt{U+1} - 1$

$U^{(0)} = 0$

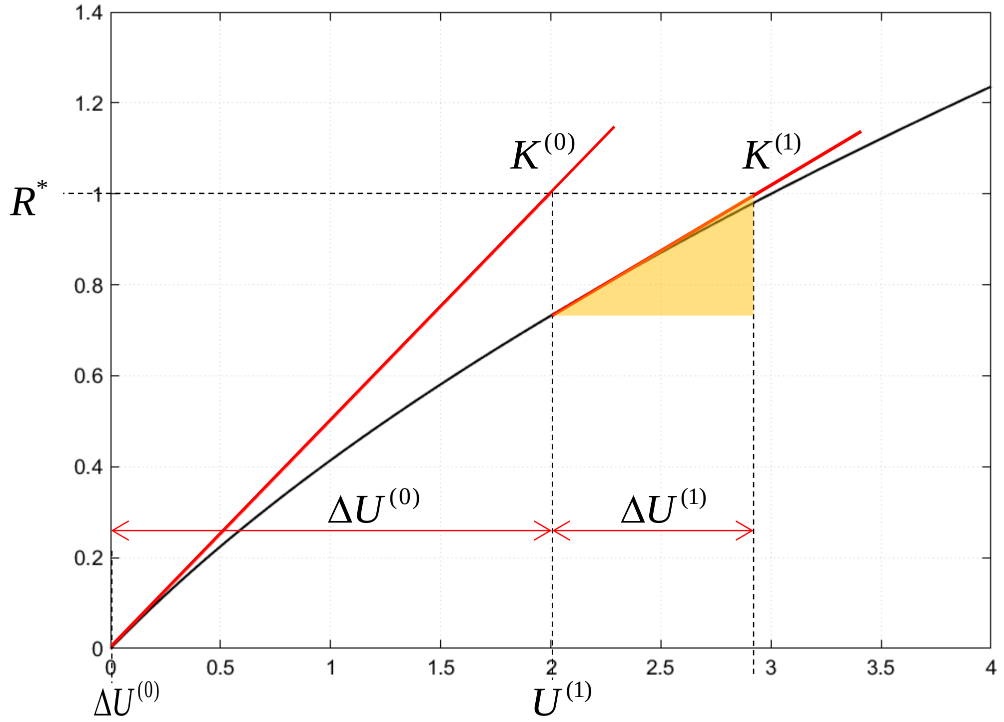
$K^{(0)} = \dfrac{1}{2}, \; R^* = 1, \; F^{(0)} = 0 \;\rightarrow\; \dfrac{1}{2}\Delta U^{(0)} = 1, \; \Delta U^{(0)} = 2$



Step 1 $i=1:\ K^{(1)}\Delta U^{(1)} = R^{*} - F^{(1)}$ with $K = \dfrac{1}{2\sqrt{U+1}},\quad F = \sqrt{U+1}-1$

$U^{(1)} = U^{(0)} + \Delta U^{(0)} = 2$ (error = 33.3 %)

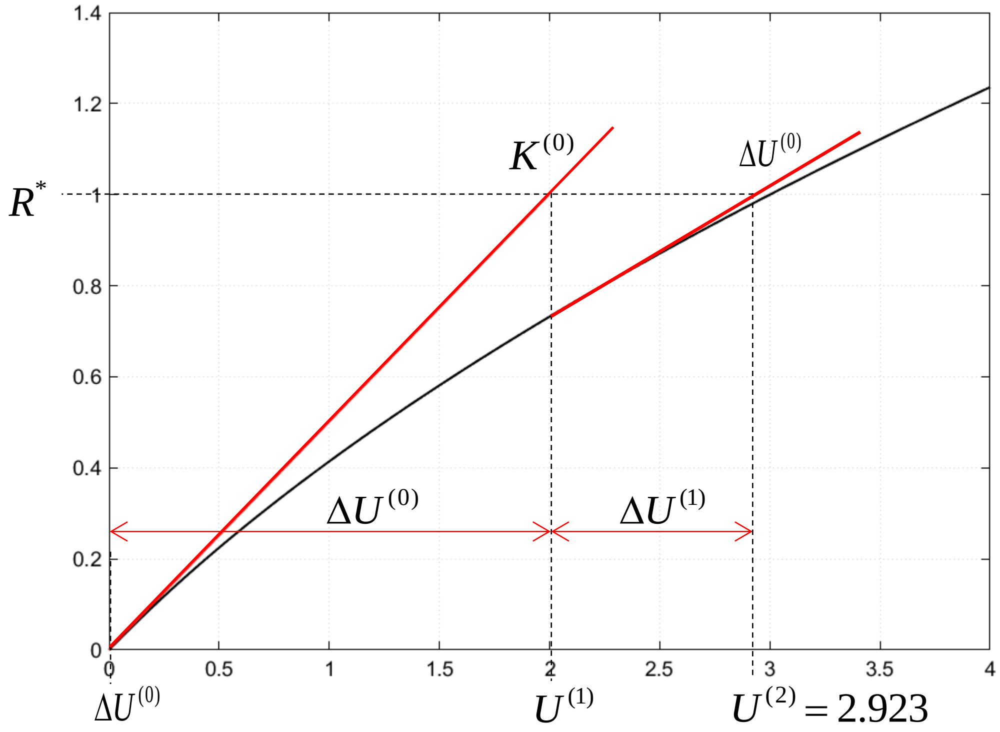
$$K^{(1)} = \frac{1}{2\sqrt{3}},\quad R^{*}=1,\quad F^{(1)} = \sqrt{3}-1 \ \rightarrow\ \frac{1}{2\sqrt{3}}\Delta U^{(1)} = 1-(\sqrt{3}-1),\quad \Delta U^{(1)} = 4\sqrt{3}-6\ \ (\approx 0.9282)$$



Step 2 $i = 2:\ K^{(2)}\Delta U^{(2)} = R^* - F^{(2)}$ with $K = \dfrac{1}{2\sqrt{U+1}},\quad F = \sqrt{U+1} - 1$

$U^{(2)} = U^{(1)} + \Delta U^{(1)} = 4\sqrt{3} - 4 \approx 2.9282$ (error = 2.4 %)

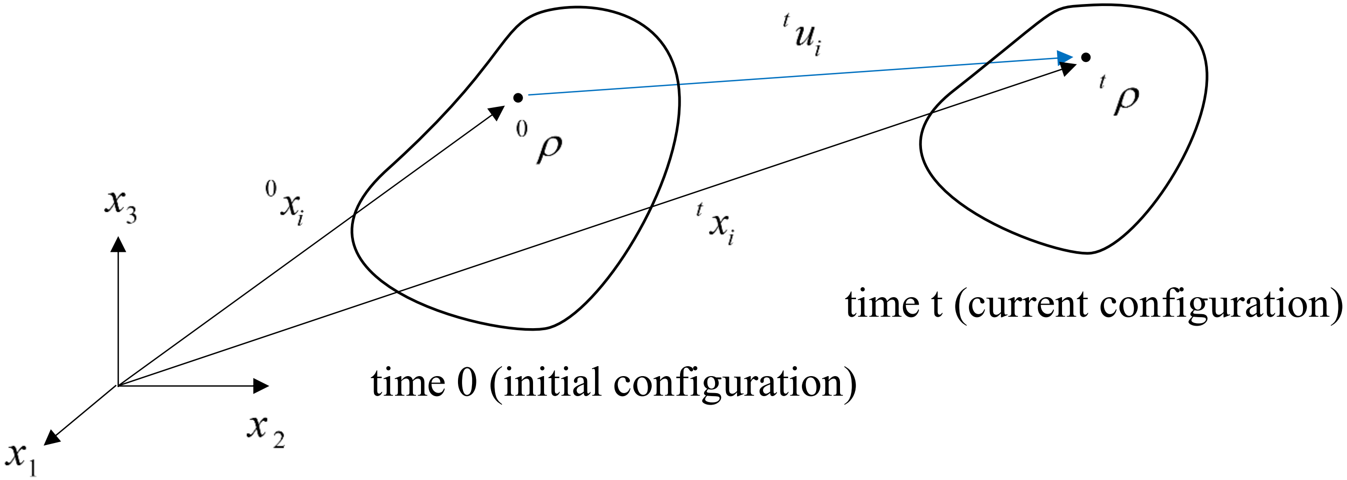
$K^{(2)} \approx 0.2523,\quad R^* = 1,\quad F^{(2)} = 0.9820$ → $0.2523 \times \Delta U^{(2)} = 1 - 0.9820,\quad \Delta U^{(2)} \approx 0.0713$



$U^{(3)} = U^{(2)} + \Delta U^{(2)}$
$= 2.9282 + 0.0713 = 2.995$

9. Large Displacement Kinematics

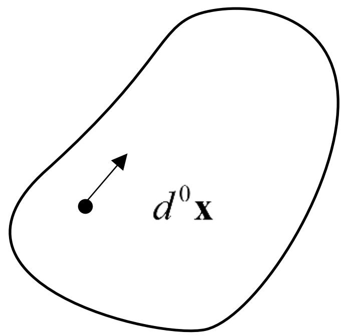
Deformation Tensor



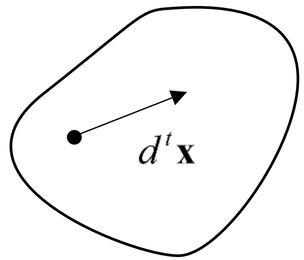
$${}^t x_i = {}^0 x_i + {}^t u_i$$

$${}^t x_i = {}^t x_i({}^0 x_i) \rightarrow \begin{cases} {}^t x_1 = {}^t x_1({}^0 x_1, {}^0 x_2, {}^0 x_3) \\ {}^t x_2 = {}^t x_2({}^0 x_1, {}^0 x_2, {}^0 x_3) \\ {}^t x_3 = {}^t x_3({}^0 x_1, {}^0 x_2, {}^0 x_3) \end{cases}$$

$${}^t u_i = {}^t u_i({}^0 x_i)$$



(initial configuration)



(current configuration)

Material vectors: $d^0\mathbf{x} = \begin{bmatrix} d^0x_1 \\ d^0x_2 \\ d^0x_3 \end{bmatrix}$, $d^t\mathbf{x} = \begin{bmatrix} d^tx_1 \\ d^tx_2 \\ d^tx_3 \end{bmatrix}$

Chain Rule

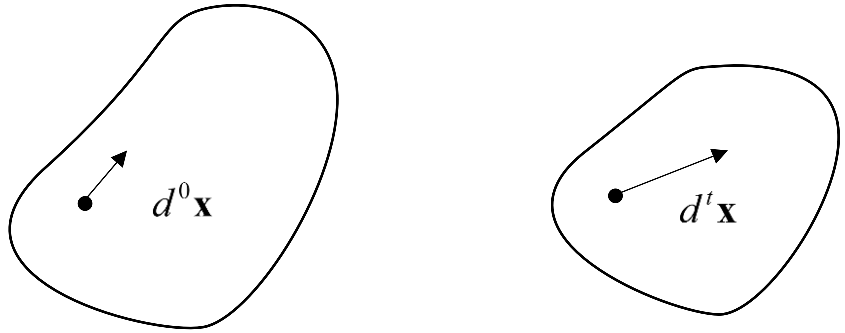
$$d^tx_1 = \frac{\partial^t x_1}{\partial^0 x_1} d^0x_1 + \frac{\partial^t x_1}{\partial^0 x_2} d^0x_2 + \frac{\partial^t x_1}{\partial^0 x_3} d^0x_3, \quad d^tx_2 = \frac{\partial^t x_2}{\partial^0 x_1} d^0x_1 + \frac{\partial^t x_2}{\partial^0 x_2} d^0x_2 + \frac{\partial^t x_2}{\partial^0 x_3} d^0x_3$$

$$d^tx_3 = \frac{\partial^t x_3}{\partial^0 x_1} d^0x_1 + \frac{\partial^t x_3}{\partial^0 x_2} d^0x_2 + \frac{\partial^t x_3}{\partial^0 x_3} d^0x_3$$

Matrix form	Tensor form
$d\,^t\mathbf{x} = {}_0^t\mathbf{X}\, d\,^0\mathbf{x}$ with ${}_0^t\mathbf{X} = \dfrac{\partial\,^t\mathbf{x}}{\partial\,^0\mathbf{x}} = \begin{bmatrix} \dfrac{\partial\,^t x_1}{\partial\,^0 x_1} & \dfrac{\partial\,^t x_1}{\partial\,^0 x_2} & \dfrac{\partial\,^t x_1}{\partial\,^0 x_3} \\ \dfrac{\partial\,^t x_2}{\partial\,^0 x_1} & \dfrac{\partial\,^t x_2}{\partial\,^0 x_2} & \dfrac{\partial\,^t x_2}{\partial\,^0 x_3} \\ \dfrac{\partial\,^t x_3}{\partial\,^0 x_1} & \dfrac{\partial\,^t x_3}{\partial\,^0 x_2} & \dfrac{\partial\,^t x_3}{\partial\,^0 x_3} \end{bmatrix}$ (deformation matrix from time 0 to time t)	$d\,^t x_i = {}_0^t x_{ij}\, d\,^0 x_j$ with ${}_0^t x_{ij} = \dfrac{\partial\,^t x_i}{\partial\,^0 x_j}$ (deformation tensor)

"Length, angle and volume changes can be calculated using deformation gradient tensor."

(i) Length change



time 0 (initial configuration) time t (current configuration)

$$d\,{}^{0}\mathbf{x} = d\,{}^{0}s\,{}^{0}\mathbf{n} \qquad d\,{}^{t}\mathbf{x} = d\,{}^{t}s\,{}^{t}\mathbf{n}$$

$d\,{}^{0}s$, $d\,{}^{t}s$: lengths at time 0 and t, respectively

${}^{0}\mathbf{n}$, ${}^{t}\mathbf{n}$: direction vectors at time 0 and t, respectively

The stretch is defined by ${}^{t}\lambda = \dfrac{d\,{}^{t}s}{d\,{}^{0}s}$

$$(d\,{}^{t}s)^2 = d\,{}^{t}\mathbf{x}^T d\,{}^{t}\mathbf{x} \quad \leftarrow \quad d\,{}^{t}\mathbf{x} = {}_{0}^{t}\mathbf{X}\, d\,{}^{0}\mathbf{x}$$

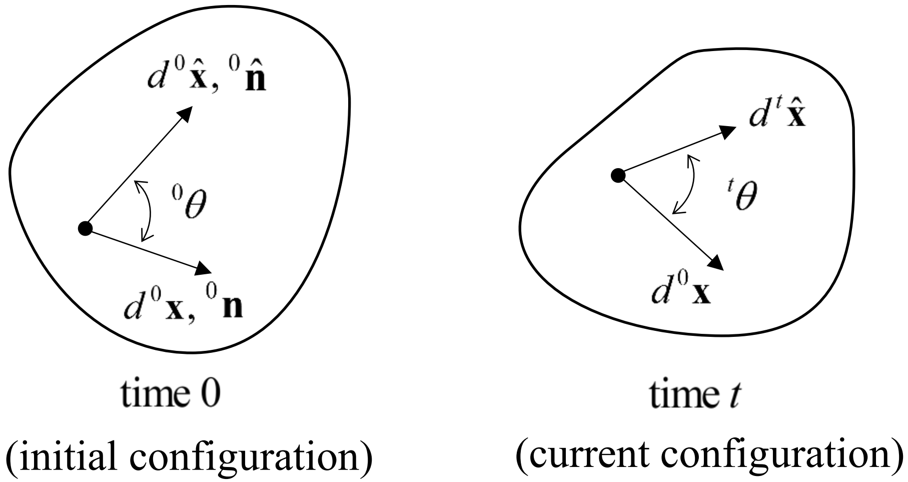
$$(d\,{}^{t}s)^2 = d\,{}^{0}\mathbf{x}^T\, {}_{0}^{t}\mathbf{X}^T\, {}_{0}^{t}\mathbf{X}\, d\,{}^{0}\mathbf{x} = d\,{}^{0}\mathbf{x}^T\, {}_{0}^{t}\mathbf{C}\, d\,{}^{0}\mathbf{x},$$

with ${}_{0}^{t}\mathbf{C} = {}_{0}^{t}\mathbf{X}^T\, {}_{0}^{t}\mathbf{X}$: Right Cauchy-Green deformation tensor

$${}^{t}\lambda^2 = \frac{(d\,{}^{t}s)^2}{(d\,{}^{0}s)^2} = \frac{d\,{}^{0}\mathbf{x}^T\, {}_{0}^{t}\mathbf{C}\, d\,{}^{0}\mathbf{x}}{(d\,{}^{0}s)^2} = \frac{d\,{}^{0}\mathbf{x}^T}{d\,{}^{0}s}\, {}_{0}^{t}\mathbf{C}\, \frac{d\,{}^{0}\mathbf{x}}{d\,{}^{0}s} \quad \leftarrow \quad \frac{d\,{}^{0}\mathbf{x}}{d\,{}^{0}s} = {}^{0}\mathbf{n}$$

$${}^{t}\lambda = \left({}^{0}\mathbf{n}^T\, {}_{0}^{t}\mathbf{C}\, {}^{0}\mathbf{n}\right)^{1/2}$$

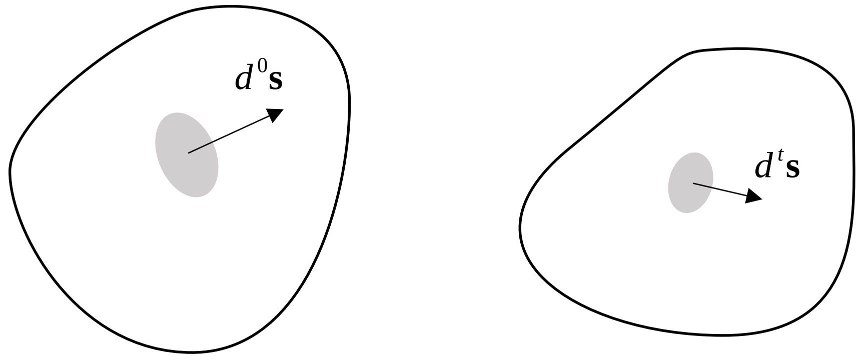
(ii) Angle change



The angle $^t\theta$ between $d\,^0\mathbf{x}$ and $d\,^t\hat{\mathbf{x}}$ at time t is calculated by

$$\cos\,^t\theta = \frac{^0\mathbf{n}^T\,{}^t_0\mathbf{C}\,^0\hat{\mathbf{n}}}{^t\lambda\,^t\hat{\lambda}}.$$

(iii) Surface change



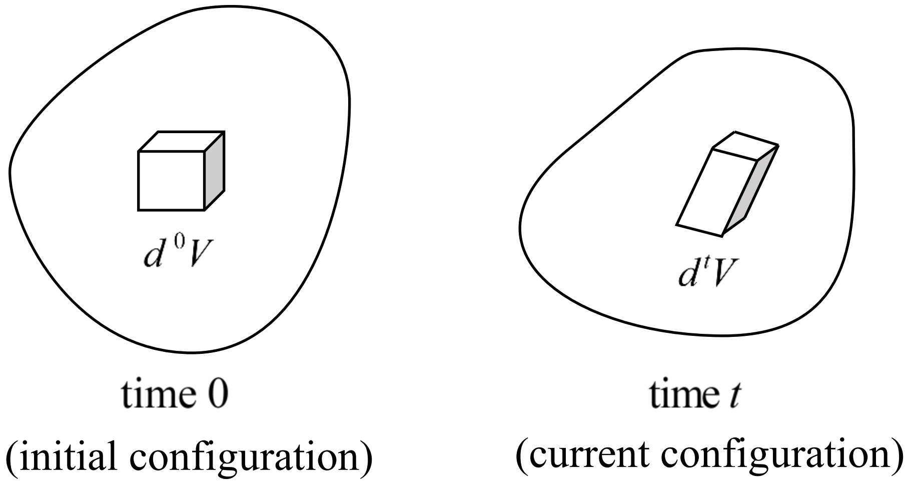
time 0
(initial configuration)

time t
(current configuration)

$$d^0\mathbf{s} = d^0 s\, {}^0\mathbf{n}, \qquad d^t\mathbf{s} = d^t s\, {}^t\mathbf{n}$$

$$d^t\mathbf{s} = \frac{{}^0\rho}{{}^t\rho}\, {}_t^0\mathbf{X}^{\mathrm{T}}\, d^0\mathbf{s} \quad \text{or} \quad {}^t\mathbf{n}\, d^t s = \frac{{}^0\rho}{{}^t\rho}\, {}_t^0\mathbf{X}^{\mathrm{T}}\, {}^0\mathbf{n}\, d^0 s$$

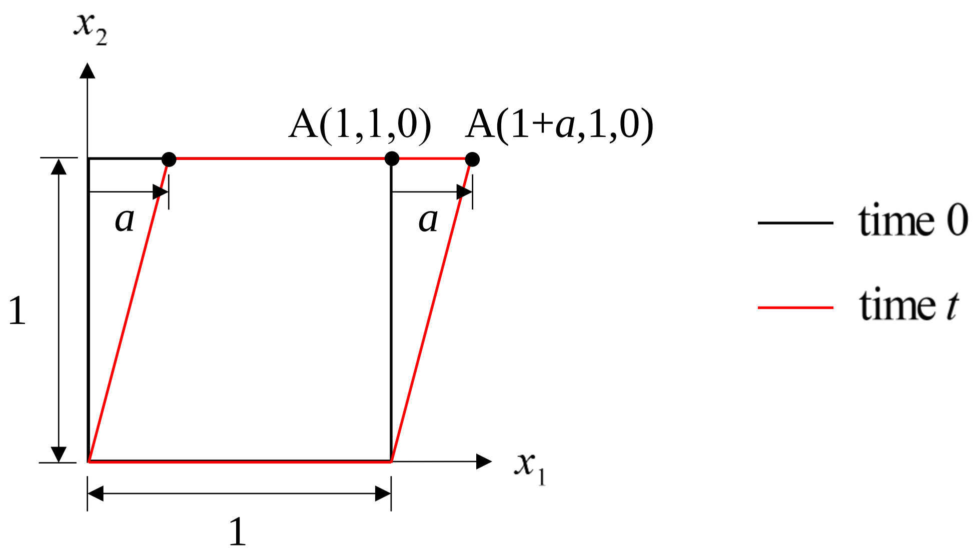
(iv) Volume change



$$\det({}_0^t\mathbf{X}) = \frac{d\,{}^tV}{d\,{}^0V} = \frac{{}^0\rho}{{}^t\rho}$$ ← ${}^0\rho d\,{}^0V = {}^t\rho d\,{}^tV$: mass conservation

$$ {}^t\rho = \frac{{}^0\rho}{\det({}_0^t\mathbf{X})}$$

Example - Simple Shear



$$^{t}x_1 = {}^{0}x_1 + a\,{}^{0}x_2$$

$$^{t}x_2 = {}^{0}x_2$$

$$^{t}x_3 = {}^{0}x_3$$

$$^{t}_{0}\mathbf{X} = \begin{bmatrix} 1 & a & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Length change in the direction of ${}^{0}\mathbf{n}=\begin{bmatrix}1 & 0 & 0\end{bmatrix}^T$ (at ${}^{0}x_1={}^{0}x_2=0$)

$$ {}_{0}^{t}\mathbf{C}={}_{0}^{t}\mathbf{X}^T\,{}_{0}^{t}\mathbf{X}=\begin{bmatrix}1 & 0 & 0\\ a & 1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1 & a & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}=\begin{bmatrix}1 & a & 0\\ a & a^2+1 & 0\\ 0 & 0 & 1\end{bmatrix}, $$

$$ {}^{t}\lambda=\left({}^{0}\mathbf{n}^T\,{}_{0}^{t}\mathbf{C}\,{}^{0}\mathbf{n}\right)^{1/2}=\left(\begin{bmatrix}1 & 0 & 0\end{bmatrix}\begin{bmatrix}1 & a & 0\\ a & a^2+1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}1\\ 0\\ 0\end{bmatrix}\right)^{1/2}=1 $$

Length change in the direction of ${}^{0}\hat{\mathbf{n}}=\begin{bmatrix}0 & 1 & 0\end{bmatrix}^T$ (at ${}^{0}x_1={}^{0}x_2=0$)

$$ {}^{t}\hat{\lambda}=\left({}^{0}\hat{\mathbf{n}}^T\,{}_{0}^{t}\mathbf{C}\,{}^{0}\hat{\mathbf{n}}\right)^{1/2}=\left(\begin{bmatrix}0 & 1 & 0\end{bmatrix}\begin{bmatrix}1 & a & 0\\ a & a^2+1 & 0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}0\\ 1\\ 0\end{bmatrix}\right)^{1/2}=\sqrt{a^2+1} $$

Angle change between ${}^{0}\mathbf{n}=\begin{bmatrix}1 & 0 & 0\end{bmatrix}^{T}$ and ${}^{0}\hat{\mathbf{n}}=\begin{bmatrix}0 & 1 & 0\end{bmatrix}^{T}$ (at ${}^{0}x_{1}={}^{0}x_{2}=0$).

$${}^{t}\lambda=1$$

$${}^{t}\hat{\lambda}=\sqrt{a^{2}+1}$$

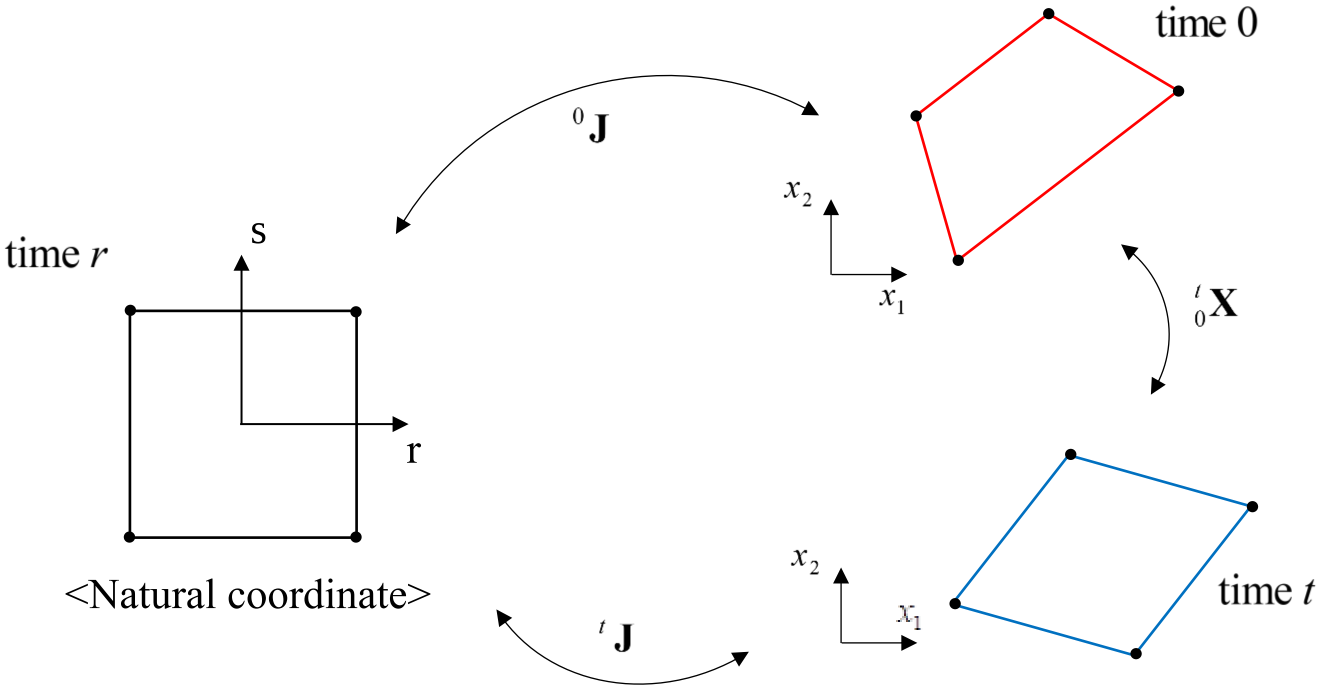
$${}_{0}^{t}\mathbf{C}=\begin{bmatrix}1 & a & 0\\ a & a^{2}+1 & 0\\ 0 & 0 & 1\end{bmatrix}$$

$$\cos{}^{t}\theta=\frac{{}^{0}\mathbf{n}^{T}\,{}_{0}^{t}\mathbf{C}\,{}^{0}\hat{\mathbf{n}}}{{}^{t}\lambda\,{}^{t}\hat{\lambda}}=\frac{a}{\sqrt{a^{2}+1}}$$

Volume change

$\det({}_{0}^{t}\mathbf{X})=1$: no change of volume

Deformation Tensor in Isoparametric Procedure



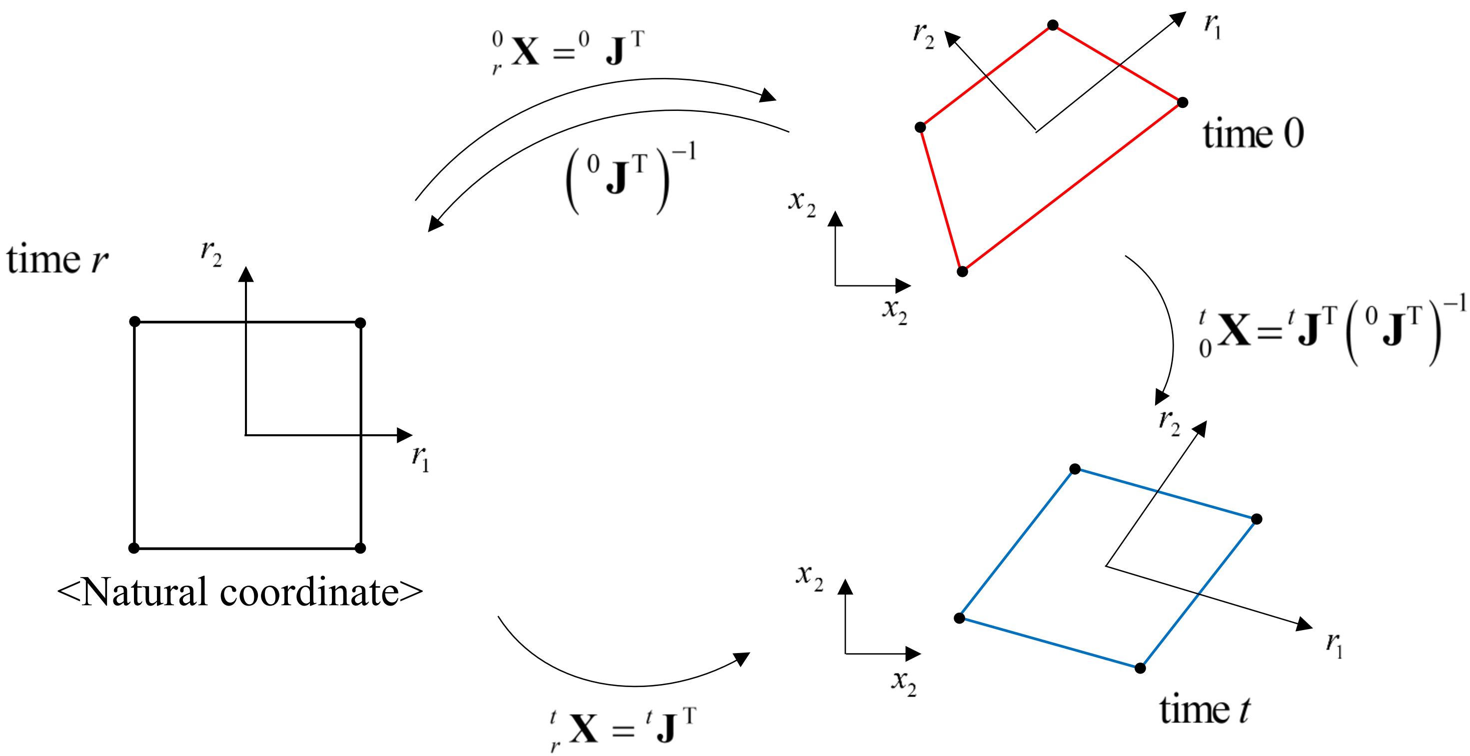
$$\begin{cases} \mathbf{x} = h_i(r,s)\mathbf{x}_i \\ \mathbf{u} = h_i(r,s)\mathbf{u}_i \\ h_i = h_i(r,s) \end{cases} \rightarrow \begin{cases} {}^0\mathbf{x} = h_i(r,s)\,{}^0\mathbf{x}_i \\ {}^t\mathbf{x} = h_i(r,s)\,{}^t\mathbf{x}_i \\ {}^t\mathbf{u} = {}^t\mathbf{x} - {}^0\mathbf{x} = h_i(r,s)({}^t\mathbf{x}_i - {}^0\mathbf{x}_i) = h_i(r,s)\,{}^t\mathbf{u}_i \end{cases}$$

$$\boxed{d\,{}^t\mathbf{x} = {}^t_0\mathbf{X}\,d\,{}^0\mathbf{x}}$$ with $${}^t_0\mathbf{X} = \frac{\partial\,{}^t x_i}{\partial\,{}^0 x_j}(\mathbf{e}_i \otimes \mathbf{e}_j)$$: deformation matrix

$$\begin{cases} r,\ s \\ x,\ y \end{cases} \rightarrow \begin{cases} r_1,\ r_2 \\ x_1,\ x_2 \end{cases}$$

$$\mathbf{J} = \begin{bmatrix} \dfrac{\partial x}{\partial r} & \dfrac{\partial y}{\partial r} \\ \dfrac{\partial x}{\partial s} & \dfrac{\partial y}{\partial s} \end{bmatrix} \rightarrow \mathbf{J} = \begin{bmatrix} \dfrac{\partial x_1}{\partial r_1} & \dfrac{\partial x_2}{\partial r_1} \\ \dfrac{\partial x_1}{\partial r_2} & \dfrac{\partial x_2}{\partial r_2} \end{bmatrix} \rightarrow \mathbf{J} = \frac{\partial x_j}{\partial r_i}(\mathbf{e}_i \otimes \mathbf{e}_j),\quad \mathbf{J}^T = \frac{\partial x_i}{\partial r_j}(\mathbf{e}_i \otimes \mathbf{e}_j)$$

We can identify ${}^{0}_{r}\mathbf{X} = {}^{0}\mathbf{J}^{\mathrm{T}} = \dfrac{\partial\, {}^{0}x_i}{\partial r_j}(\mathbf{e}_i \otimes \mathbf{e}_j)$ and ${}^{t}_{r}\mathbf{X} = {}^{t}\mathbf{J}^{\mathrm{T}} = \dfrac{\partial\, {}^{t}x_i}{\partial r_j}(\mathbf{e}_i \otimes \mathbf{e}_j)$.



$$\boxed{{}^{t}_{0}\mathbf{X} = {}^{t}\mathbf{J}^{\mathrm{T}}\left({}^{0}\mathbf{J}^{\mathrm{T}}\right)^{-1}}$$

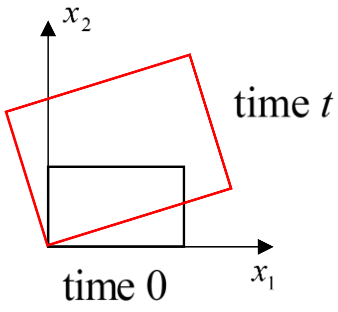
Green-Lagrange Strain Tensor

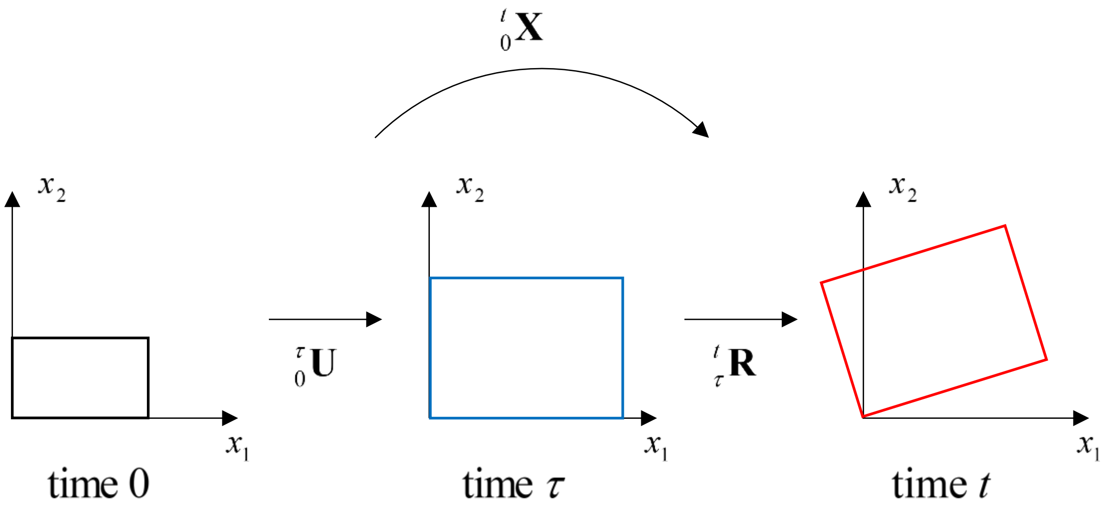
Polar decomposition

$$ {}_0^t\mathbf{X} = {}_\tau^t\mathbf{R}\,{}_0^\tau\mathbf{U} = {}_0^t\mathbf{R}\,{}_0^t\mathbf{U} $$

${}_0^t\mathbf{R}$: Rotation matrix (orthogonal matrix: ${}_0^t\mathbf{R}^{\mathrm{T}}\,{}_0^t\mathbf{R} = \mathbf{I}$, ${}_0^t\mathbf{R}^{\mathrm{T}} = {}_0^t\mathbf{R}^{-1} = {}_t^0\mathbf{R}$)

${}_0^t\mathbf{U}$: Stretch matrix (symmetric matrix)





Green-Lagrange strain

$${}_0^t\boldsymbol{\varepsilon} = \frac{1}{2}\left({}_0^t\mathbf{X}^{\mathrm{T}}\, {}_0^t\mathbf{X} - \mathbf{I} \right) \qquad \text{(eq. 9.1)}$$

$$= \frac{1}{2}\left(\left({}_0^t\mathbf{R}\, {}_0^t\mathbf{U} \right)^{\mathrm{T}} \left({}_0^t\mathbf{R}\, {}_0^t\mathbf{U} \right) - \mathbf{I} \right)$$

$$= \frac{1}{2}\left({}_0^t\mathbf{U}^{\mathrm{T}}\, {}_0^t\mathbf{U} - \mathbf{I} \right)$$

- under rigid body rotations: ${}_0^t\mathbf{X} = {}_0^t\mathbf{R}$ → ${}_0^t\boldsymbol{\varepsilon} = \mathbf{0}$
- under rigid body translations

  ${}^t\mathbf{x} = {}^0\mathbf{x} + \mathbf{c}$ with $\mathbf{c}$: constant vector

  $${}_0^t\mathbf{X} = \frac{\partial\, {}^t\mathbf{x}}{\partial\, {}^0\mathbf{x}} = \mathbf{I} \quad \rightarrow \quad {}_0^t\boldsymbol{\varepsilon} = \mathbf{0}$$

- under no displacement

  ${}^t\mathbf{x} = {}^0\mathbf{x}$

  $${}_0^t\mathbf{X} = \mathbf{I} \quad \rightarrow \quad {}_0^t\boldsymbol{\varepsilon} = \mathbf{0}$$

$$ {}_{0}^{t}\mathbf{X} = \frac{\partial^{t}\mathbf{x}}{\partial^{0}\mathbf{x}} = \frac{\partial}{\partial^{0}\mathbf{x}}\left({}^{0}\mathbf{x} + {}^{t}\mathbf{u}\right) = \frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}} + \mathbf{I} \qquad \text{(eq. 9.2)} $$

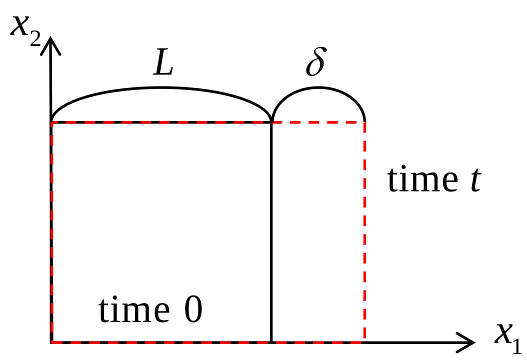
Using (eq. 9.2) in (eq. 9.1), Green-Lagrange strain w.r.t displacements can be derived as follows

$$ {}_{0}^{t}\boldsymbol{\varepsilon} = \frac{1}{2}\left[\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}} + \mathbf{I}\right)^{\mathrm{T}}\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}} + \mathbf{I}\right) - \mathbf{I}\right] $$

$$ = \frac{1}{2}\left[\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)^{\mathrm{T}} + \frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}} + \left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)^{\mathrm{T}}\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)\right] $$

Tensor form	**Matrix form**
${}_{0}^{t}\varepsilon_{ij} = \frac{1}{2}\left({}_{0}^{t}u_{i,j} + {}_{0}^{t}u_{j,i} + {}_{0}^{t}u_{k,i}\,{}_{0}^{t}u_{k,j}\right)$	${}_{0}^{t}\boldsymbol{\varepsilon} = \frac{1}{2}\left[\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)^{\mathrm{T}} + \frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}} + \left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)^{\mathrm{T}}\left(\frac{\partial^{t}\mathbf{u}}{\partial^{0}\mathbf{x}}\right)\right]$

Example – One directional stretch



$$ {}^{t}x_1 = \frac{\delta}{L}\,{}^{0}x_1 + {}^{0}x_1 $$

$$ {}^{t}x_2 = {}^{0}x_2 $$

$$ {}^{t}x_3 = {}^{0}x_3 $$

$$ {}^{t}u_1 = {}^{t}x_1 - {}^{0}x_1 = \frac{\delta}{L}\,{}^{0}x_1, \quad {}^{t}u_2 = {}^{t}u_3 = 0 $$

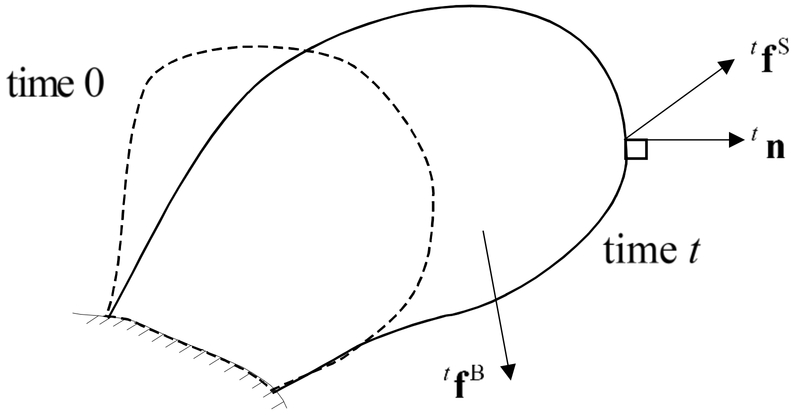
$$ {}_{0}^{t}\varepsilon_{ij} = \frac{1}{2}\left({}_{0}^{t}u_{i,j} + {}_{0}^{t}u_{j,i} + {}_{0}^{t}u_{k,i}\,{}_{0}^{t}u_{k,j}\right) \quad : \quad \text{GL-strain} $$

$$ {}_{0}^{t}\varepsilon_{11} = \frac{1}{2}\left({}_{0}^{t}u_{1,1} + {}_{0}^{t}u_{1,1} + {}_{0}^{t}u_{1,1}\,{}_{0}^{t}u_{1,1} + \cancel{{}_{0}^{t}u_{2,1}\,{}_{0}^{t}u_{2,1}} + \cancel{{}_{0}^{t}u_{3,1}\,{}_{0}^{t}u_{3,1}}\right) \quad \leftarrow \quad {}_{0}^{t}u_{1,1} = \frac{d\,{}^{t}u_1}{d\,{}^{0}x_1} = \frac{\delta}{L} $$

$$ = \frac{1}{2}\left(\frac{\delta}{L} + \frac{\delta}{L} + \left(\frac{\delta}{L}\right)^2\right) = \frac{\delta}{L} + \frac{1}{2}\left(\frac{\delta}{L}\right)^2 $$

10. Total Lagrangian Formulation (TL-formulation)

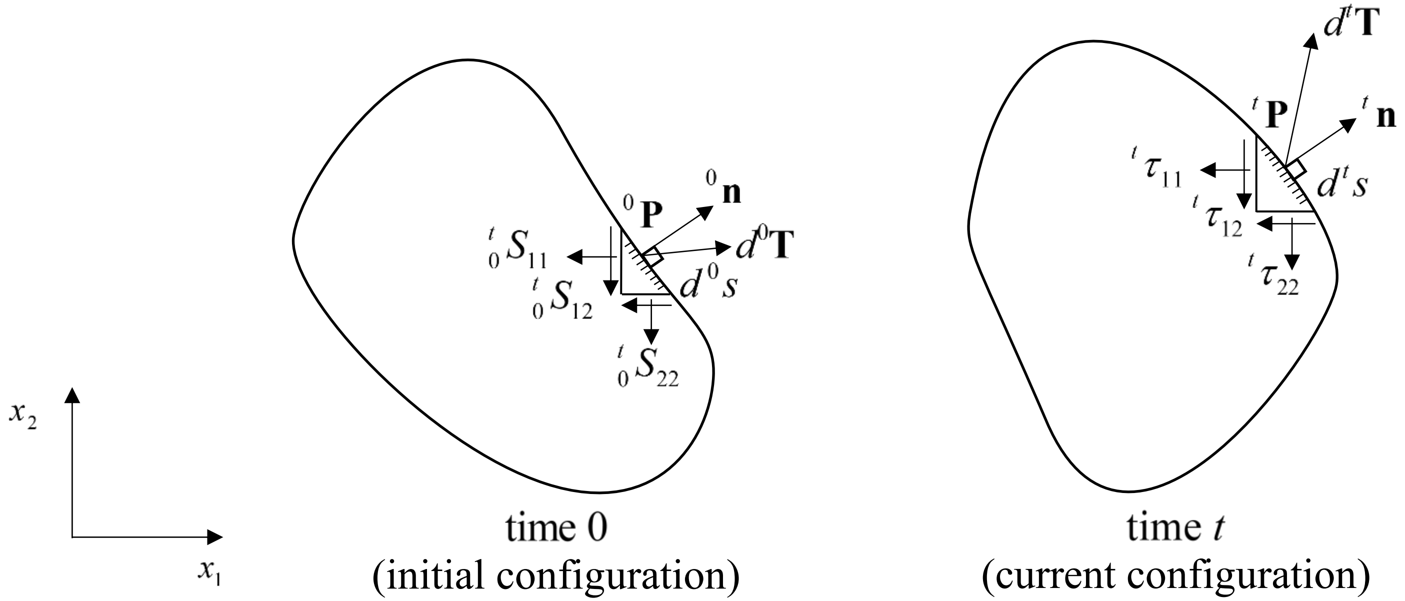
Equilibrium equations at time *t*



$$\begin{cases} \dfrac{\partial \tau_{ij}}{\partial x_j} + f_i^{B} = 0 & \text{in } V \\ \tau_{ij} n_j = f_i^{S} & \text{on } S_f \end{cases} \rightarrow \begin{cases} \dfrac{\partial\, {}^{t}\tau_{ij}}{\partial\, {}^{t}x_j} + {}^{t}f_i^{B} = 0 & \text{in } {}^{t}V \\ {}^{t}\tau_{ij}\, {}^{t}n_j = {}^{t}f_i^{S} & \text{on } {}^{t}S_f \end{cases}$$

(Note) The configuration of the body at time t is unknown!

Second Piola-Kirchhoff Stress & Cauchy Stress



Let us define a fictitious force $d\,{}^0\mathbf{T}$

$d\,{}^t\mathbf{T} = {}_0^t\mathbf{X}\,d\,{}^0\mathbf{T}$ or $d\,{}^0\mathbf{T} = {}_t^0\mathbf{X}\,d\,{}^t\mathbf{T}$, (eq. 10.1)

$d\,{}^0\mathbf{T} = {}_0^t\mathbf{S}\,{}^0\mathbf{n}\,d\,{}^0s$, (eq. 10.2)

where $d\,{}^t\mathbf{T}$ is a force applied in a small area in the current configuration.

${}_0^t\mathbf{S}$ is the stress tensor, which is in equilibrium with $d\,{}^0\mathbf{T}$.

${}_0^t\mathbf{S}$ → "Second Piola-Kirchhoff stress (2-PK stress)"

$d\,{}^t\mathbf{T} = {}^t\boldsymbol{\tau}\,{}^t\mathbf{n}\,d\,{}^ts$ (Cauchy stress is in equilibrium with $d\,{}^t\mathbf{T}$) (eq. 10.3)

${}^t\mathbf{n}\,d\,{}^ts = \dfrac{{}^0\rho}{{}^t\rho}\,{}_t^0\mathbf{X}^{\mathrm{T}}\,{}^0\mathbf{n}\,d\,{}^0s$ (Transport of an oriented surface) (eq. 10.4)

Using (eq. 10.4) in (eq. 10.3), $d\,{}^t\mathbf{T} = {}^t\boldsymbol{\tau}\dfrac{{}^0\rho}{{}^t\rho}\,{}_t^0\mathbf{X}^{\mathrm{T}}\,{}^0\mathbf{n}\,d\,{}^0s$. (eq. 10.5)

Substituting (eq. 10.2) and (eq. 10.5) into (eq. 10.1), the relation between the 2PK stress and Cauchy stress is obtained

$${}_0^t\mathbf{S}\,\cancel{{}^0\mathbf{n}\,d\,{}^0s} = {}_t^0\mathbf{X}\,{}^t\boldsymbol{\tau}\dfrac{{}^0\rho}{{}^t\rho}\,{}_t^0\mathbf{X}^{\mathrm{T}}\,\cancel{{}^0\mathbf{n}\,d\,{}^0s} \quad \rightarrow \quad \boxed{{}_0^t\mathbf{S} = \dfrac{{}^0\rho}{{}^t\rho}\,{}_t^0\mathbf{X}\,{}^t\boldsymbol{\tau}\,{}_t^0\mathbf{X}^{\mathrm{T}}}$$

(Note)

The components of the Second Piola-Kirchhoff stress do not change under rigid body motions.

Tensor form	**Matrix form**
${}_0^tS_{ij} = \frac{{}^0\rho}{{}^t\rho}\, {}_t^0x_{im}\, {}_t^0x_{jn}\, {}^t\tau_{mn}$ ${}^t\tau_{mn} = \frac{{}^t\rho}{{}^0\rho}\, {}_0^tx_{mi}\, {}_0^tx_{nj}\, {}_0^tS_{ij}$	${}_0^t\mathbf{S} = \frac{{}^0\rho}{{}^t\rho}\, {}_t^0\mathbf{X}\, {}^t\boldsymbol{\tau}\, {}_t^0\mathbf{X}^{\mathrm{T}}$ ${}^t\boldsymbol{\tau} = \frac{{}^t\rho}{{}^0\rho}\, {}_0^t\mathbf{X}\, {}_0^t\mathbf{S}\, {}_0^t\mathbf{X}^{\mathrm{T}}$

<u>Variation of Green-Lagrange strain</u> (Virtual Green-Lagrange strain)

$$ {}_0^t\varepsilon_{ij} = \frac{1}{2}\left({}_0^t x_{ki}\, {}_0^t x_{kj} - \delta_{ij} \right) $$

Variation of GL strain

$$ \delta {}_0^t\varepsilon_{ij} = \frac{1}{2}\left(\delta {}_0^t x_{ki}\, {}_0^t x_{kj} + {}_0^t x_{ki}\, \delta {}_0^t x_{kj} \right) $$

$$ \delta {}_0^t x_{ij} = \delta\left(\frac{\partial^t x_i}{\partial^0 x_j} \right) = \delta\left(\frac{\partial^t u_i}{\partial^0 x_j} + \delta_{ij} \right) = \frac{\partial \delta^t u_i}{\partial^0 x_j} = \frac{\partial \delta u_i}{\partial^t x_m} \cdot \frac{\partial^t x_m}{\partial^0 x_j} = \delta\, {}_t u_{i,m}\, {}_0^t x_{mj} \quad \leftarrow \delta^t u_i \rightarrow \delta u_i,\ \ \delta\, {}_t u_{i,m} = \frac{\partial \delta u_i}{\partial^t x_m} $$

$$ \delta {}_0^t\varepsilon_{ij} = \frac{1}{2}\left(\delta\, {}_t u_{k,m}\, {}_0^t x_{mi}\, {}_0^t x_{kj} + {}_0^t x_{ki}\, \delta\, {}_t u_{k,m}\, {}_0^t x_{mj} \right) $$

$$ = \frac{1}{2}\left({}_0^t x_{mi}\, \delta\, {}_t u_{k,m}\, {}_0^t x_{kj} + {}_0^t x_{ki}\, \delta\, {}_t u_{k,m}\, {}_0^t x_{mj} \right) $$

← m and k can be switched.

$$ = \frac{1}{2}\left({}_0^t x_{ki}\, \delta\, {}_t u_{m,k}\, {}_0^t x_{mj} + {}_0^t x_{ki}\, \delta\, {}_t u_{k,m}\, {}_0^t x_{mj} \right) $$

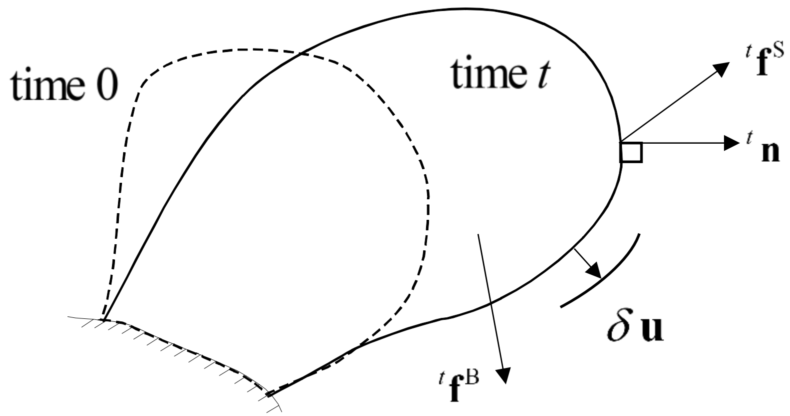
$$= {}_0^t x_{ki} \left[\frac{1}{2} \left(\delta\, {}_t u_{m,k} + \delta\, {}_t u_{k,m} \right) \right] {}_0^t x_{mj}$$

$\leftarrow$ $\delta_t e_{km} = \frac{1}{2} \left(\delta\, {}_t u_{m,k} + \delta\, {}_t u_{k,m} \right)$: virtual strain at time t

$$= {}_0^t x_{ki}\ \delta_t e_{km}\ {}_0^t x_{mj}$$

Tensor form	**Matrix form**
$\delta {}_0^t \varepsilon_{ij} = {}_0^t x_{mi}\ {}_0^t x_{nj} \delta_t e_{mn}$	$\delta {}_0^t \boldsymbol{\varepsilon} = {}_0^t \mathbf{X}^{\mathrm{T}} \delta_t \mathbf{e}\, {}_0^t \mathbf{X}$
$\delta_t e_{mn} = {}_t^0 x_{im}\ {}_t^0 x_{jn} \delta {}_0^t \varepsilon_{ij}$	$\delta_t \mathbf{e} = {}_t^0 \mathbf{X}^{\mathrm{T}} \delta {}_0^t \boldsymbol{\varepsilon}\, {}_t^0 \mathbf{X}$

Principle of Virtual Work at time *t*



$$\begin{cases} \dfrac{\partial\, {}^{t}\tau_{ij}}{\partial\, {}^{t}x_j} + {}^{t}f_i^{\mathrm{B}} = 0 & \text{in } {}^{t}V \\ {}^{t}\tau_{ij}\, {}^{t}n_j = {}^{t}f_i^{\mathrm{S}} & \text{on } {}^{t}S_f \end{cases}$$

$\delta\mathbf{u}$: Virtual displacement imposed on the current configuration

Applying δu_i to the body in the configuration at time t, PVW at time t is obtained.

$$\boxed{\int_{{}^tV} {}^t\tau_{ij}\delta_t e_{ij} d^tV = {}^tR}$$ (in tensor form)

where

$\int_{{}^tV} {}^t\tau_{ij}\delta_t e_{ij} d^tV$: internal virtual work at time t

${}^tR$: external virtual work at time t

${}^t\tau_{ij}$: Cauchy stress at time t

$\delta_t e_{ij}$: virtual strain at time t, $\delta_t e_{ij} = \frac{1}{2}\left(\delta_t u_{i,j} + \delta_t u_{j,i}\right) = \frac{1}{2}\left(\frac{\partial \delta u_i}{\partial {}^t x_j} + \frac{\partial \delta u_j}{\partial {}^t x_i}\right)$

δu_i : virtual displacement

$$\boxed{\int_{{}^tV} {}^t\boldsymbol{\tau}\cdot\delta_t\mathbf{e}\, d^tV = {}^tR}$$ (in matrix form with ${}^t\boldsymbol{\tau}\cdot\delta_t\mathbf{e} = {}^t\tau_{ij}\delta_t e_{ij}$)

$$\int_{{}^tV} {}^t\tau_{kl}\,\delta_t e_{kl}\,d^tV = \int_{{}^tV} \left(\frac{{}^t\rho}{{}^0\rho}\, {}^t_0x_{ki}\, {}^t_0x_{lj}\, {}^t_0S_{ij} \right) \left({}^0_tx_{pk}\, {}^0_tx_{ql}\, \delta^t_0\varepsilon_{pq} \right) d^tV \qquad \leftarrow \quad {}^t\tau_{mn} = \frac{{}^t\rho}{{}^0\rho}\, {}^t_0x_{mi}\, {}^t_0x_{nj}\, {}^t_0S_{ij}$$

$$\delta_t e_{mn} = {}^0_tx_{pm}\, {}^0_tx_{qn}\, \delta^t_0\varepsilon_{pq}$$

$${}^t_0x_{ki}\, {}^0_tx_{pk} = \delta_{pi}, \quad {}^t_0x_{lj}\, {}^0_tx_{ql} = \delta_{qj}$$

$$= \int_{{}^tV} \frac{{}^t\rho}{{}^0\rho}\, {}^t_0S_{ij}\, \delta^t_0\varepsilon_{ij}\, d^tV \qquad \leftarrow \quad {}^t\rho\, d^tV = {}^0\rho\, d^0V$$

$$= \int_{{}^0V} {}^t_0S_{ij}\, \delta^t_0\varepsilon_{ij}\, d^0V$$

$$\boxed{\int_{{}^0V} {}^t_0S_{ij}\, \delta^t_0\varepsilon_{ij}\, d^0V = {}^tR} \quad \text{or} \quad \boxed{\int_{{}^0V} {}^t_0\mathbf{S} \cdot \delta^t_0\boldsymbol{\varepsilon}\, d^0V = {}^tR}$$

(Note)

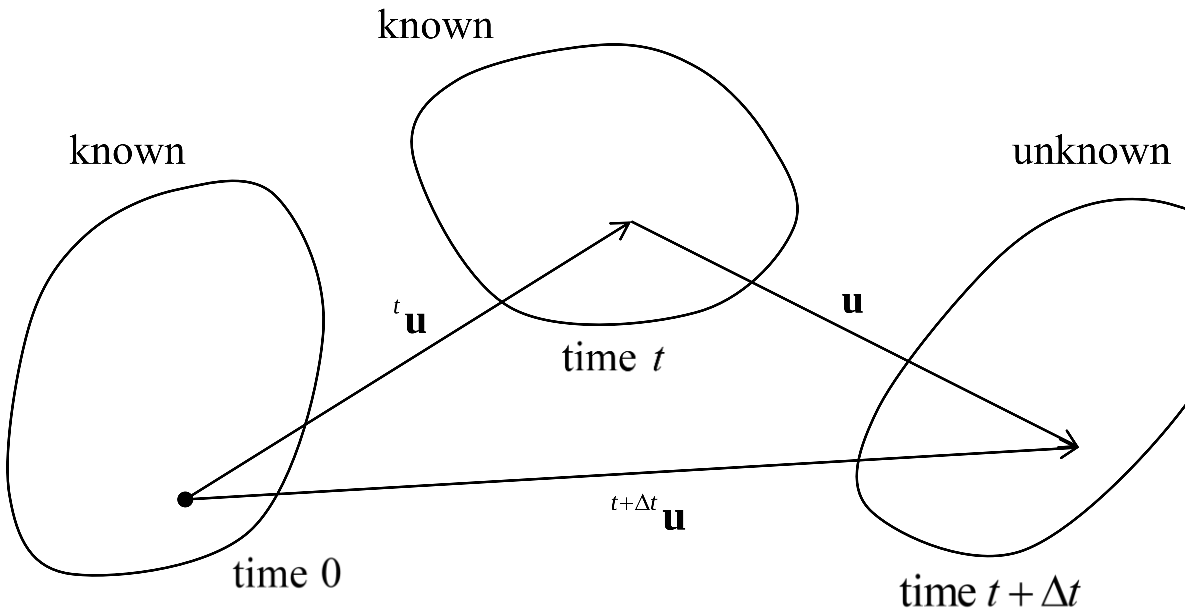
$^{t}\boldsymbol{\tau} \longleftrightarrow \delta_t\mathbf{e}$: energy conjugate

${}_0^t\mathbf{S} \longleftrightarrow \delta_0^t\boldsymbol{\varepsilon}$: energy conjugate

$^{t}\boldsymbol{\tau}$: Stress measured from the current configuration (time = $\boldsymbol{t}$)

${}_0^t\mathbf{S}$: Stress measured from the initial configuration (time = 0)

Principle of Virtual Work at time *t+Δt*



$$^{t+\Delta t}\mathbf{u} = {}^{t}\mathbf{u} + \mathbf{u}$$

PVW at time $t+\Delta t$ (Configurations at time 0 and t are known)

$$\boxed{\int_{^0V} {}^{t+\Delta t}_{0}S_{ij}\, \delta_0^{t+\Delta t}\varepsilon_{ij}\, d^0V = {}^{t+\Delta t}R}$$

$^{t+\Delta t}_{0}S_{ij} = ^{t}_{0}S_{ij} + _{0}S_{ij}$ ← $_{0}S_{ij}$: incremental 2-PK stress ($^{t}_{0}S_{ij}$ is known)

$^{t+\Delta t}_{0}\varepsilon_{ij} = ^{t}_{0}\varepsilon_{ij} + _{0}\varepsilon_{ij}$ ← $_{0}\varepsilon_{ij}$: incremental GL strain ($^{t}_{0}\varepsilon_{ij}$ is known)

Separation of GL strain

$$^{t+\Delta t}_{0}\varepsilon_{ij} = \frac{1}{2}\left(^{t+\Delta t}_{0}u_{i,j} + ^{t+\Delta t}_{0}u_{j,i} + ^{t+\Delta t}_{0}u_{k,i}\ ^{t+\Delta t}_{0}u_{k,j}\right) \leftarrow \ ^{t+\Delta t}u_i = {}^{t}u_i + u_i\ ,\ ^{t+\Delta t}u_{i,j} = {}^{t}u_{i,j} + _{0}u_{i,j}$$

$$= ^{t}_{0}\varepsilon_{ij} + \frac{1}{2}\left(_{0}u_{i,j} + _{0}u_{j,i} + ^{t}_{0}u_{k,i}\ _{0}u_{k,j} + _{0}u_{k,i}\ ^{t}_{0}u_{k,j}\right) + \frac{1}{2}\left(_{0}u_{k,i}\ _{0}u_{k,j}\right)$$

$$= ^{t}_{0}\varepsilon_{ij} + _{0}\varepsilon_{ij} \quad \leftarrow \quad _{0}\varepsilon_{ij} = _{0}e_{ij} + _{0}\eta_{ij} \quad \text{: incremental GL strain}$$

$$= ^{t}_{0}\varepsilon_{ij} + _{0}e_{ij} + _{0}\eta_{ij}$$

where

$^{t}_{0}u_{k,i}\ _{0}u_{k,j} + _{0}u_{k,i}\ ^{t}_{0}u_{k,j}$: initial displacement effect ($^{t}_{0}u_{k,i}$ is known)

$_{0}e_{ij}$: linear part of the incremental GL strain

$_{0}\eta_{ij}$: nonlinear part of the incremental GL strain

Then, PVW becomes

$$\int_{^0V} {}_0^{t+\Delta t}S_{ij}\,\delta_0^{t+\Delta t}\varepsilon_{ij}\,d\,{}^0V = {}^{t+\Delta t}R \qquad \leftarrow \quad {}_0^{t+\Delta t}S_{ij} = {}_0^t S_{ij} + {}_0S_{ij}, \qquad \delta_0^{t+\Delta t}\varepsilon_{ij} = \cancel{\delta_0^t\varepsilon_{ij}} + \delta_0\varepsilon_{ij} == \delta_0 e_{ij} + \delta_0\eta_{ij}$$

$$\int_{^0V} \left({}_0^t S_{ij} + {}_0 S_{ij}\right)\left(\delta_0 e_{ij} + \delta_0\eta_{ij}\right) d\,{}^0V = {}^{t+\Delta t}R$$

$$\int_{^0V} \left({}_0^t S_{ij}\,\delta_0 e_{ij} + {}_0^t S_{ij}\,\delta_0\eta_{ij} + {}_0 S_{ij}\,\delta_0 e_{ij} + {}_0 S_{ij}\,\delta_0\eta_{ij}\right) d\,{}^0V = {}^{t+\Delta t}R$$

- Virtual terms are given.
- Principle unknown is ${}_0u_i$.

$${}_0^tS_{ij}\,\delta_0 e_{ij}$$

known

- ${}_0^tS_{ij}$: known
- $\delta_0 e_{ij} = \frac{1}{2}\left(\delta_0 u_{i,j} + \delta_0 u_{j,i} + {}_0^t u_{k,i}\,\delta_0 u_{k,j} + \delta u_{k,i}\,{}_0^t u_{k,j}\right)$: known

  known: ${}_0^t u_{k,i}$, ${}_0^t u_{k,j}$

$${}_0^tS_{ij}\,\delta_0 \eta_{ij}$$

Linear

- ${}_0^tS_{ij}$: known
- $\delta_0 \eta_{ij} = \frac{1}{2}\left(\delta_0 u_{k,i}\,{}_0u_{k,j} + {}_0u_{k,i}\,\delta_0 u_{k,j}\right)$: linear

  unknown: ${}_0u_{k,j}$, ${}_0u_{k,i}$

$${}_0S_{ij}\,\delta_0 e_{ij}$$

Linear

- ${}_0S_{ij} \approx C_{ijrs}\,{}_0e_{rs}$: linear

  → ${}_0e_{ij} = \frac{1}{2}\left({}_0u_{i,j} + {}_0u_{j,i} + {}_0^t u_{k,i}\,{}_0u_{k,j} + {}_0u_{k,i}\,{}_0^t u_{k,j}\right)$

  known: ${}_0^t u_{k,i}$, ${}_0^t u_{k,j}$

  unknown: ${}_0u_{i,j}$, ${}_0u_{j,i}$

  unknown: ${}_0u_{k,j}$, ${}_0u_{k,i}$
- $\delta_0 e_{ij}$: known

$${}_0S_{ij}\,\delta_0 \eta_{ij}$$

nonlinear

- ${}_0S_{ij}$: linear
- $\delta_0 \eta_{ij}$: linear

To linearize PVW, the nonlinear term is neglected.

$$\int_{^0V} \left({}_0^tS_{ij}\, \delta_0 e_{ij} + {}_0^tS_{ij}\, \delta_0 \eta_{ij} + {}_0S_{ij}\, \delta_0 e_{ij} + \cancel{{}_0S_{ij}\, \delta_0 \eta_{ij}} \right) d\,^0V = {}^{t+\Delta t}R$$

$$\int_{^0V} {}_0S_{ij}\, \delta_0 e_{ij}\, d\,^0V + \int_{^0V} {}_0^tS_{ij}\, \delta_0 \eta_{ij}\, d\,^0V = {}^{t+\Delta t}R - \int_{^0V} {}_0^tS_{ij}\, \delta_0 e_{ij}\, d\,^0V$$

Using ${}_0S_{ij} \approx {}_0C_{ijrs}\, {}_0e_{rs}$, the total Lagrange formulation is obtained

$$\boxed{\int_{^0V} \delta_0 e_{ij}\, {}_0C_{ijrs}\, {}_0e_{rs}\, d\,^0V + \int_{^0V} {}_0^tS_{ij}\, \delta_0 \eta_{ij}\, d\,^0V = {}^{t+\Delta t}R - \int_{^0V} {}_0^tS_{ij}\, \delta_0 e_{ij}\, d\,^0V} \qquad \text{(eq. 10.6)}$$

Finite Element Discretization

Using ${}_0\mathbf{e}^{(m)} = {}_0^t\mathbf{B}_L^{(m)}\mathbf{U}$ and $\delta_0\mathbf{e}^{(m)} = {}_0^t\mathbf{B}_L^{(m)}\delta\mathbf{U}$ in (eq. 10.6),

$$\sum_m \left[\int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_L^{(m)T}\, {}_0\mathbf{C}^{(m)}\, {}_0^t\mathbf{B}_L^{(m)}\, dV^{(m)} + \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_{NL}^{(m)T}\, {}_0^t\mathbf{S}^{(m)}\, {}_0^t\mathbf{B}_{NL}^{(m)}\, dV^{(m)} \right] \mathbf{U} = {}^{t+\Delta t}\mathbf{R} - \sum_m \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_L^{(m)T}\, {}_0^t\mathbf{S}^{(m)}\, dV^{(m)},$$

where ${}_0^t\mathbf{B}_{NL}^{(m)}$ is a matrix utilized to calculate $\delta_0\boldsymbol{\eta}^{(m)}$.

Finally, the following incremental equilibrium equation is given

$$\boxed{{}_0^t\mathbf{K}\,\mathbf{U} = {}^{t+\Delta t}\mathbf{R} - {}_0^t\mathbf{F}}$$

with

$${}_0^t\mathbf{K} = \sum_m {}_0^t\mathbf{K}^{(m)}, \quad {}_0^t\mathbf{K}^{(m)} = {}_0^t\mathbf{K}_L^{(m)} + {}_0^t\mathbf{K}_{NL}^{(m)}, \quad {}_0^t\mathbf{F} = \sum_m {}_0^t\mathbf{F}^{(m)}$$

where

$\mathbf{U}$: incremental nodal displacement vector

${}_0^t\mathbf{K}$: tangential stiffness matrix,

${}_0^t\mathbf{F}$: internal force vector

$${}_0^t\mathbf{K}_L^{(m)} = \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_L^{(m)T} \, {}_0\mathbf{C}^{(m)} \, {}_0^t\mathbf{B}_L^{(m)} \, dV^{(m)}$$

$${}_0^t\mathbf{K}_{NL}^{(m)} = \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_{NL}^{(m)T} \, {}_0^t\mathbf{S}^{(m)} \, {}_0^t\mathbf{B}_{NL}^{(m)} \, dV^{(m)}$$

$${}_0^t\mathbf{F}^{(m)} = \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_L^{(m)T} \, {}_0^t\mathbf{S}^{(m)} \, dV^{(m)}$$

Updated Lagrangian Formulation (UL-formulation)

Total Lagrangian formulation

$$\int_{{}^0V} \delta_0 e_{ij}\ {}_0C_{ijrs}\ {}_0e_{rs}\ d^0V + \int_{{}^0V} {}^t_0S_{ij}\ \delta_0\eta_{ij}\ d^0V = {}^{t+\Delta t}R - \int_{{}^0V} {}^t_0S_{ij}\ \delta_0 e_{ij}\ d^0V$$

In the TL formulation, set time 0 → t

$$\int_{{}^tV} \delta_t e_{ij}\ {}_tC_{ijrs}\ {}_te_{rs}\ d^tV + \int_{{}^tV} {}^t_tS_{ij}\ \delta_t\eta_{ij}\ d^tV = {}^{t+\Delta t}R - \int_{{}^tV} {}^t_tS_{ij}\ \delta_t e_{ij}\ d^tV$$

$${}^t_tS_{ij} = {}^t\tau_{ij}$$

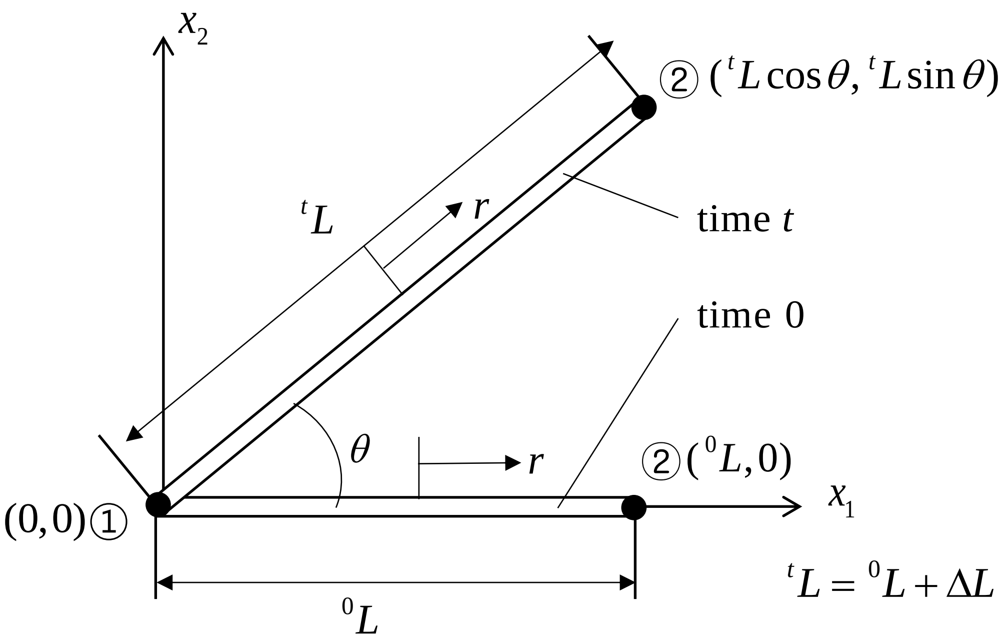
$${}^tx_i = {}^0x_i + {}^tu_i \quad \rightarrow \quad {}^tx_i = {}^tx_i + {}^tu_i,\ \ {}^tu_i = 0$$

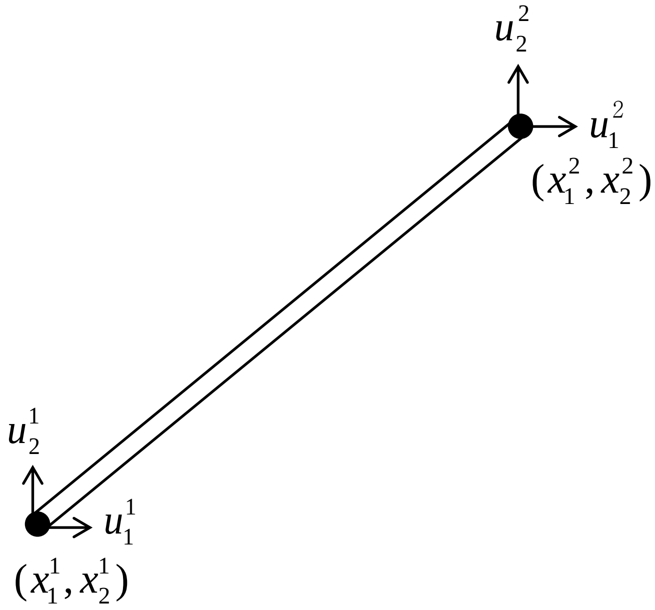
$${}_te_{ij} = \frac{1}{2}\left({}_tu_{i,j} + {}_tu_{j,i} + \cancel{{}^t_tu_{k,i}}\ {}_tu_{k,j} + u_{k,i}\ \cancel{{}^t_tu_{k,j}}\right) = \frac{1}{2}\left({}_tu_{i,j} + {}_tu_{j,i}\right)$$

Updated Lagrangian formulation

$$\boxed{\int_{{}^tV} \delta_t e_{ij}\ {}_tC_{ijrs}\ {}_te_{rs}\ d^tV + \int_{{}^tV} {}^t\tau_{ij}\ \delta_t\eta_{ij}\ d^tV = {}^{t+\Delta t}R - \int_{{}^tV} {}^t\tau_{ij}\ \delta_t e_{ij}\ d^tV}$$

Example – Large displacement of a 2-node bar element (TL-formulation)





<DOFs>

Interpolations

$^{o}\mathbf{x} = h_i(r)\,^{o}\mathbf{x}_i$: Geometry at $\text{time} = 0$, $\begin{bmatrix} ^{o}x_1 \\ ^{o}x_2 \end{bmatrix} = h_i(r)\begin{bmatrix} ^{o}x_1^i \\ ^{o}x_2^i \end{bmatrix}$

$^{t}\mathbf{x} = h_i(r)\,^{t}\mathbf{x}_i$: Geometry at $\text{time} = t$, $\begin{bmatrix} ^{t}x_1 \\ ^{t}x_2 \end{bmatrix} = h_i(r)\begin{bmatrix} ^{t}x_1^i \\ ^{t}x_2^i \end{bmatrix}$

$^{t}\mathbf{u} = h_i(r)\,^{t}\mathbf{u}_i$: Displacement at $\text{time} = t$, $\begin{bmatrix} ^{t}u_1 \\ ^{t}u_2 \end{bmatrix} = h_i(r)\begin{bmatrix} ^{t}u_1^i \\ ^{t}u_2^i \end{bmatrix}$

$\mathbf{u} = h_i(r)\mathbf{u}_i$: Incremental displacement, $\begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = h_i(r)\begin{bmatrix} u_1^i \\ u_2^i \end{bmatrix}$

$\delta\mathbf{u} = h_i(r)\delta\mathbf{u}_i$: Virtual displacement, $\begin{bmatrix} \delta u_1 \\ \delta u_2 \end{bmatrix} = h_i(r)\begin{bmatrix} \delta u_1^i \\ \delta u_2^i \end{bmatrix}$

$h_1 = \frac{1}{2}(1-r)$, $h_2 = \frac{1}{2}(1+r)$

Linear part of the incremental GL strain

$$ {}^{0}e_{ij} = \frac{1}{2}\left({}_{0}u_{i,j} + {}_{0}u_{j,i} + {}_{0}^{t}u_{k,i}\,{}_{0}u_{k,j} + {}_{0}u_{k,i}\,{}_{0}^{t}u_{k,j}\right) $$

$$ {}^{0}e_{11} = {}_{0}u_{1,1} + {}_{0}^{t}u_{1,1}\,{}_{0}u_{1,1} + {}_{0}u_{2,1}\,{}_{0}^{t}u_{2,1} = \frac{\partial u_1}{\partial {}^{0}x_1} + \frac{\partial {}^{t}u_1}{\partial {}^{0}x_1}\frac{\partial u_1}{\partial {}^{0}x_1} + \frac{\partial u_2}{\partial {}^{0}x_1}\frac{\partial {}^{t}u_2}{\partial {}^{0}x_1} $$

$$ \mathbf{U} = \begin{bmatrix} u_1^1 \\ u_1^2 \\ u_2^1 \\ u_2^2 \end{bmatrix}, \quad {}^{t}\mathbf{U} = \begin{bmatrix} {}^{t}u_1^1 \\ {}^{t}u_1^2 \\ {}^{t}u_2^1 \\ {}^{t}u_2^2 \end{bmatrix} = \begin{bmatrix} 0 \\ \left({}^{0}L + \Delta L\right)\cos\theta - {}^{0}L \\ 0 \\ \left({}^{0}L + \Delta L\right)\sin\theta \end{bmatrix} $$

$$ {}^{t}u_1 = \begin{bmatrix} \frac{1}{2}(1-r) & \frac{1}{2}(1+r) & 0 & 0 \end{bmatrix} {}^{t}\mathbf{U}, \quad \frac{\partial {}^{t}u_1}{\partial r} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 & 0 \end{bmatrix} {}^{t}\mathbf{U}, \quad \frac{\partial {}^{0}x_1}{\partial r} = \frac{{}^{0}L}{2} $$

$$ \rightarrow \frac{\partial {}^{t}u_1}{\partial {}^{0}x_1} = \frac{\partial {}^{t}u_1}{\partial r}\frac{\partial r}{\partial {}^{0}x_1} = \begin{bmatrix} -\frac{1}{2} & \frac{1}{2} & 0 & 0 \end{bmatrix} \frac{2}{{}^{0}L}\,{}^{t}\mathbf{U} = \frac{{}^{0}L + \Delta L}{{}^{0}L}\cos\theta - 1 $$

$$ {}^{t}u_2 = \left[0 \;\; 0 \;\; \frac{1}{2}(1-r) \;\; \frac{1}{2}(1+r) \right] {}^{t}\mathbf{U}, \quad \frac{\partial\, {}^{t}u_2}{\partial r} = \left[0 \;\; 0 \;\; -\frac{1}{2} \;\; \frac{1}{2} \right] {}^{t}\mathbf{U} $$

$$ \rightarrow \frac{\partial\, {}^{t}u_2}{\partial\, {}^{0}x_1} = \frac{\partial\, {}^{t}u_2}{\partial r} \frac{\partial r}{\partial\, {}^{0}x_1} = \left[0 \;\; 0 \;\; -\frac{1}{2} \;\; \frac{1}{2} \right] \frac{2}{{}^{0}L} {}^{t}\mathbf{U} = \frac{{}^{0}L + \Delta L}{{}^{0}L} \sin\theta $$

$$ u_1 = \left[\frac{1}{2}(1-r) \;\; \frac{1}{2}(1+r) \;\; 0 \;\; 0 \right] \mathbf{U}, \quad \frac{\partial u_1}{\partial r} = \left[-\frac{1}{2} \;\; \frac{1}{2} \;\; 0 \;\; 0 \right] \mathbf{U} $$

$$ \rightarrow \frac{\partial u_1}{\partial\, {}^{0}x_1} = \frac{\partial u_1}{\partial r} \frac{\partial r}{\partial\, {}^{0}x_1} = \left[-\frac{1}{2} \;\; \frac{1}{2} \;\; 0 \;\; 0 \right] \frac{2}{{}^{0}L} \mathbf{U} = \frac{1}{{}^{0}L} [-1 \;\; 1 \;\; 0 \;\; 0] \mathbf{U} $$

$$ u_2 = \left[0 \;\; 0 \;\; \frac{1}{2}(1-r) \;\; \frac{1}{2}(1+r) \right] \mathbf{U}, \quad \frac{\partial u_2}{\partial r} = \left[0 \;\; 0 \;\; -\frac{1}{2} \;\; \frac{1}{2} \right] \mathbf{U} $$

$$ \rightarrow \frac{\partial u_2}{\partial\, {}^{0}x_1} = \frac{\partial u_2}{\partial r} \frac{\partial r}{\partial\, {}^{0}x_1} = \frac{1}{{}^{0}L} [0 \;\; 0 \; -1 \;\; 1] \mathbf{U} $$

$$
{}^{0}e_{11} = \frac{1}{{}^{0}L}\left[-1\ 1\ 0\ \ 0\right]\mathbf{U} + \left(\frac{{}^{0}L+\Delta L}{{}^{0}L}\cos\theta - 1\right)\frac{1}{{}^{0}L}\left[-1\ 1\ 0\ \ 0\right]\mathbf{U} + \left(\frac{{}^{0}L+\Delta L}{{}^{0}L}\sin\theta\right)\frac{1}{{}^{0}L}\left[0\ \ 0\ \ -1\ 1\right]\mathbf{U}
$$

$$
= \frac{{}^{0}L+\Delta L}{\left({}^{0}L\right)^{2}}\left[-\cos\theta\ \ \cos\theta\ \ -\sin\theta\ \ \sin\theta\right]\mathbf{U} = {}_{0}^{t}\mathbf{B}_{L}^{(m)}\mathbf{U}
$$

$$
{}_{0}^{t}\mathbf{B}_{L}^{(m)} = \frac{{}^{0}L+\Delta L}{\left({}^{0}L\right)^{2}}\left[-\cos\theta\ \ \cos\theta\ \ -\sin\theta\ \ \sin\theta\right]
$$

Nonlinear part of the incremental GL strain

$$ {}^{0}\eta_{ij} = \frac{1}{2}\, {}_{0}u_{k,i}\, {}_{0}u_{k,j} $$

$$ {}^{0}\eta_{11} = \frac{1}{2}\left\{ \left({}_{0}u_{1,1} \right)^{2} + \left({}_{0}u_{2,1} \right)^{2} \right\} = \frac{1}{2}\left\{ \left(\frac{\partial u_1}{\partial^{0} x_1} \right)^{2} + \left(\frac{\partial u_2}{\partial^{0} x_1} \right)^{2} \right\} $$

$$ \delta^{0}\eta_{11} = \frac{\partial u_1}{\partial^{0} x_1}\frac{\partial \delta u_1}{\partial^{0} x_1} + \frac{\partial u_2}{\partial^{0} x_1}\frac{\partial \delta u_2}{\partial^{0} x_1} $$

$$ = \begin{bmatrix} \dfrac{\partial \delta u_1}{\partial^{0} x_1} & \dfrac{\partial \delta u_2}{\partial^{0} x_1} \end{bmatrix} \begin{bmatrix} \dfrac{\partial u_1}{\partial^{0} x_1} \\ \dfrac{\partial u_2}{\partial^{0} x_1} \end{bmatrix} = \delta \mathbf{U}^{\mathrm{T}}\ {}_{0}^{t}\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})\mathrm{T}}\ {}_{0}^{t}\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})}\mathbf{U} $$

$$ {}_{0}^{t}\mathbf{B}_{\mathrm{NL}}^{(m)}\mathbf{U} = \begin{bmatrix} \dfrac{\partial u_1}{\partial^{0} x_1} \\ \dfrac{\partial u_2}{\partial^{0} x_1} \end{bmatrix} = \frac{1}{{}^{0}L}\begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}\mathbf{U} \quad \rightarrow \quad \underline{ {}_{0}^{t}\mathbf{B}_{\mathrm{NL}}^{(m)} = \frac{1}{{}^{0}L}\begin{bmatrix} -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix} } $$

Tangential stiffness matrix

$$ {}_0^t\mathbf{K}^{(m)} = {}_0^t\mathbf{K}_L^{(m)} + {}_0^t\mathbf{K}_{NL}^{(m)} = \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_L^{(m)T}\ {}_0\mathbf{C}^{(m)}\ {}_0^t\mathbf{B}_L^{(m)}\, dV^{(m)} + \int_{{}^0V^{(m)}} {}_0^t\mathbf{B}_{NL}^{(m)T}\ {}_0^t\mathbf{S}^{(m)}\ {}_0^t\mathbf{B}_{NL}^{(m)}\, dV^{(m)} $$

$$ {}_0\mathbf{C} \quad \rightarrow \quad E $$

$$ {}_0^t\mathbf{S}^{(m)} \quad \rightarrow \quad {}_0^tS_{11} = E\ {}_0^t\varepsilon_{11} = E\left(\frac{\Delta L}{{}^0L} + \frac{1}{2}\left(\frac{\Delta L}{{}^0L}\right)^2\right) $$

$$ {}_0^t\mathbf{K}^{(m)} = \int_{{}^0V} {}_0^t\mathbf{B}_{\mathrm{L}}^{(\mathrm{m})\mathrm{T}} E\ {}_0^t\mathbf{B}_{\mathrm{L}}^{(\mathrm{m})} dV + \int_{{}^0V} {}_0^t\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})\mathrm{T}}\ {}_0^t\mathrm{S}_{11}\ {}_0^t\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})}\, d\,{}^0V^{(m)} $$

$$ = A_0\int_{-1}^{1} {}_0^t\mathbf{B}_{\mathrm{L}}^{(\mathrm{m})\mathrm{T}} E\ {}_0^t\mathbf{B}_{\mathrm{L}}^{(\mathrm{m})}\left(\frac{{}^0L}{2}\right)dr + A_0\int_{-1}^{1} {}_0^t\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})\mathrm{T}}\ {}_0^t\mathrm{S}_{11}\ {}_0^t\mathbf{B}_{\mathrm{NL}}^{(\mathrm{m})}\left(\frac{{}^0L}{2}\right)dr $$

Internal force vector

$$ {}_{0}^{t}\mathbf{F}^{(m)} = \int_{{}^{0}V^{(m)}} {}_{0}^{t}\mathbf{B}_{L}^{(m)T}\ {}_{0}^{t}\mathbf{S}^{(m)}\ dV^{(m)} $$

$$ {}_{0}^{t}\mathbf{F}^{(m)} = \int_{{}^{0}V^{(m)}} {}_{0}^{t}\mathbf{B}_{L}^{(m)T}\ {}_{0}^{t}S_{11}\ dV^{(m)} = A_{0}\int_{-1}^{1} {}_{0}^{t}\mathbf{B}_{\mathrm{L}}^{(\mathrm{m})\mathrm{T}}\ {}_{0}^{t}S_{11}\left(\frac{{}^{0}L}{2}\right)dr $$

Newton-Raphson Method

We iterate the following equations for $i = 0, 1, 2, 3 \ldots\ldots$

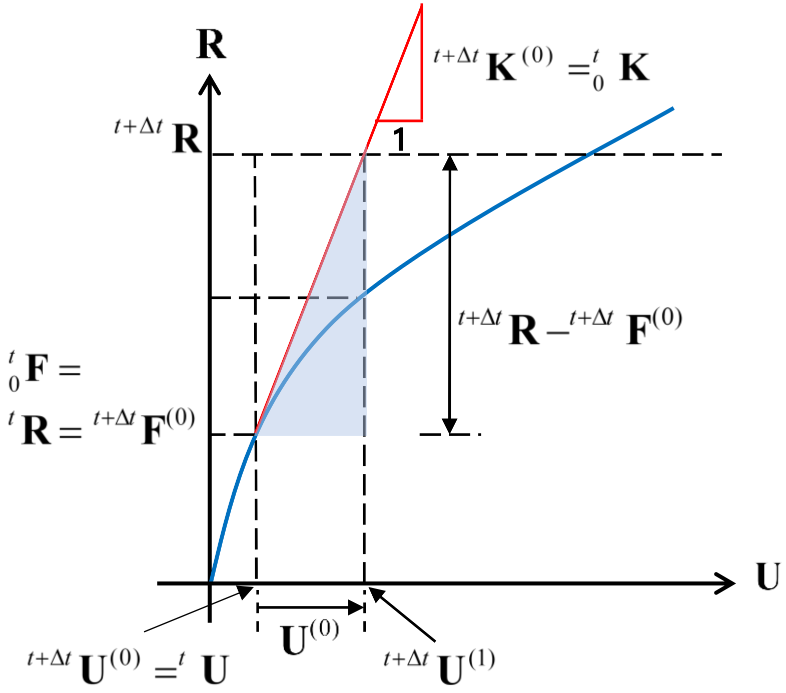
$$ {}^{t+\Delta t}\mathbf{K}^{(i)}\,\Delta\mathbf{U}^{(i)} = {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i)} $$

$$ {}^{t+\Delta t}\mathbf{U}^{(i+1)} = {}^{t+\Delta t}\mathbf{U}^{(i)} + \Delta\mathbf{U}^{(i)} $$

with initial conditions at $i = 0$: ${}^{t+\Delta t}\mathbf{U}^{(0)} = {}^{t}\mathbf{U}$, ${}^{t+\Delta t}\mathbf{K}^{(0)} = {}^{t}_{0}\mathbf{K}$, ${}^{t+\Delta t}\mathbf{F}^{(0)} = {}^{t}_{0}\mathbf{F}$

until a convergence criterion is satisfied.

$(i = 0)$

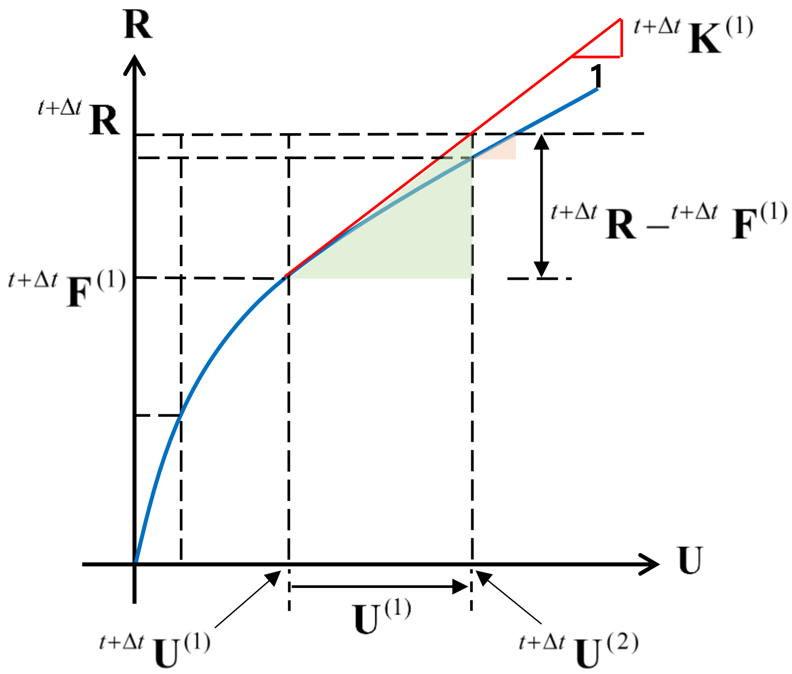


① $^{t+\Delta t}\mathbf{K}^{(0)}\,\mathbf{U}^{(0)} = {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(0)}$

② $^{t+\Delta t}\mathbf{U}^{(1)} = {}^{t+\Delta t}\mathbf{U}^{(0)} + \mathbf{U}^{(0)}$

③ $^{t+\Delta t}\mathbf{U}^{(1)} \rightarrow {}^{t+\Delta t}\mathbf{F}^{(1)}$

$(i = 1)$



① $^{t+\Delta t}\mathbf{K}^{(1)}\,\mathbf{U}^{(1)} = {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(1)}$

② $^{t+\Delta t}\mathbf{U}^{(2)} = {}^{t+\Delta t}\mathbf{U}^{(1)} + \mathbf{U}^{(1)}$

③ $^{t+\Delta t}\mathbf{U}^{(2)} \rightarrow {}^{t+\Delta t}\mathbf{F}^{(2)}$

Convergence Criterions

- Displacement criteria

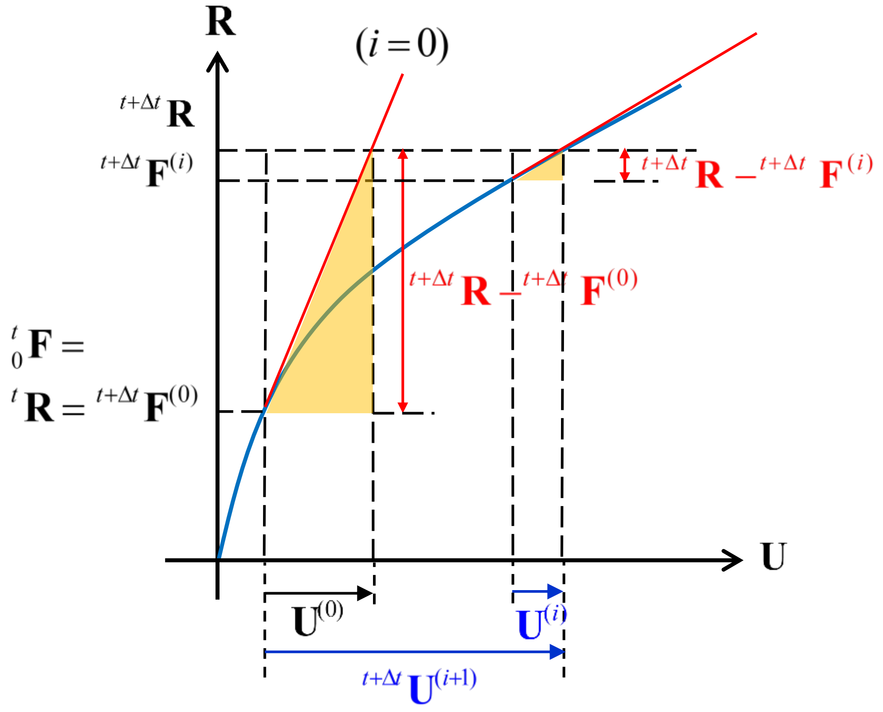
$$\frac{\left\| \mathbf{U}^{(i)} \right\|_2}{\left\| {}^{t+\Delta t}\mathbf{U}^{(i+1)} \right\|_2} \le e_D \quad \text{with} \quad \left\| \mathbf{U} \right\|_2 = \sqrt{(U_1)^2 + (U_2)^2 + \cdots + (U_N)^2}$$

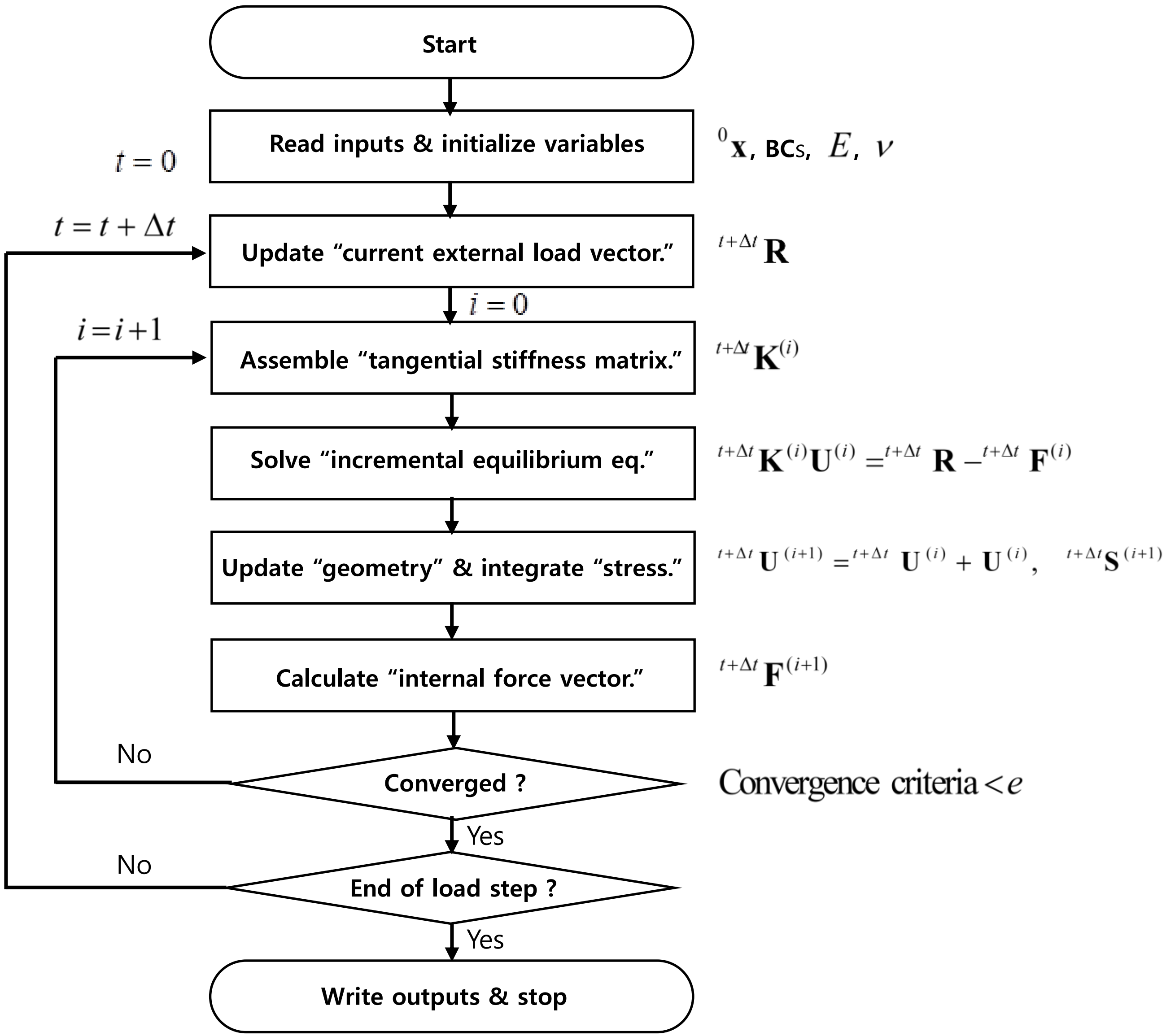
- Force criteria

$$\frac{\left\| {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i)} \right\|_2}{\left\| {}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(0)} \right\|_2} \le e_F$$

- Energy criteria

$$\frac{\mathbf{U}^{(i)T}\left({}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(i)} \right)}{\mathbf{U}^{(0)T}\left({}^{t+\Delta t}\mathbf{R} - {}^{t+\Delta t}\mathbf{F}^{(0)} \right)} \le e_E$$




Start
Read inputs & initialize variables
$t=0$
${}^{0}\mathbf{x}$, BCs, E, ν
$t=t+\Delta t$
Update "current external load vector."
${}^{t+\Delta t}\mathbf{R}$
$i=0$
$i=i+1$
Assemble "tangential stiffness matrix."
${}^{t+\Delta t}\mathbf{K}^{(i)}$
Solve "incremental equilibrium eq."
${}^{t+\Delta t}\mathbf{K}^{(i)}\mathbf{U}^{(i)}={}^{t+\Delta t}\mathbf{R}-{}^{t+\Delta t}\mathbf{F}^{(i)}$
Update "geometry" & integrate "stress."
${}^{t+\Delta t}\mathbf{U}^{(i+1)}={}^{t+\Delta t}\mathbf{U}^{(i)}+\mathbf{U}^{(i)}$, ${}^{t+\Delta t}\mathbf{S}^{(i+1)}$
Calculate "internal force vector."
${}^{t+\Delta t}\mathbf{F}^{(i+1)}$
No
Converged ?
Convergence criteria $<e$
Yes
No
End of load step ?
Yes
Write outputs & stop

11. Dynamic Analysis

Static equilibrium equations

$$\mathbf{KU} = \mathbf{R}$$

Dynamic equilibrium equations (equations of motion)

$$\boxed{\mathbf{M}\ddot{\mathbf{U}}(t) + \mathbf{C}\dot{\mathbf{U}}(t) + \mathbf{KU}(t) = \mathbf{R}(t)}$$

with $\mathbf{M}$: mass matrix, $\mathbf{C}$: damping matrix, $\mathbf{K}$: stiffness matrix

Nodal DOF vectors:

$\mathbf{U}$: displacement vector, $\dot{\mathbf{U}}$: velocity vector, $\ddot{\mathbf{U}}$: acceleration vector

Initial conditions: $\mathbf{U}\big|_{t=0} = {}^{0}\mathbf{U}$ and $\dot{\mathbf{U}}\big|_{t=0} = {}^{0}\dot{\mathbf{U}}$

Mass matrix

- Consistent mass matrix: $\mathbf{M} = \sum_m \mathbf{M}^{(m)}$, $\mathbf{M}^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)^{\mathrm{T}}} \rho^{(m)} \mathbf{H}^{(m)} dV^{(m)}$

- Lumped mass matrix: $\mathbf{M} = \begin{bmatrix} m_{11} & & & \\ & m_{22} & & \\ & & \ddots & \\ & & & m_{NN} \end{bmatrix}$

Damping matrix

- The damping matrix $\mathbf{C}$ is related to "energy dissipation".

- It can be obtained through experimental tests.

- In engineering practice, mass and stiffness proportional damping is frequently used.

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K}$$

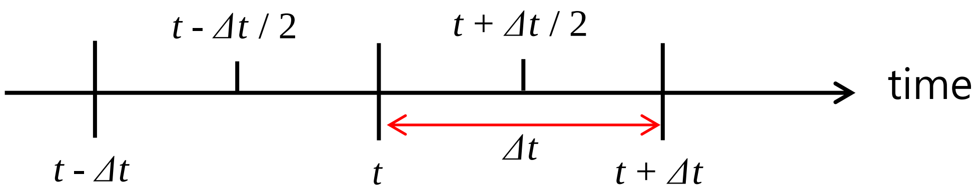
Solution Methods

- Direct integration (linear & nonlinear)
  - (I) Explicit scheme (equilibrium at time t)
    - Ex) Central difference method (CDM), ...
  - (II) Implicit scheme (equilibrium at time $t + \Delta t$)
    - Ex) Houbolt, Wilson-θ, Newmark method, ...
- (III) Mode superposition (linear)

I. Central difference method (Direct integration – explicit scheme)

Let us consider the dynamic equilibrium at time t

$$\mathbf{M}\,{}^{t}\ddot{\mathbf{U}} + \mathbf{C}\,{}^{t}\dot{\mathbf{U}} + \mathbf{K}\,{}^{t}\mathbf{U} = {}^{t}\mathbf{R} \qquad \text{(eq. 11.1)}$$



< Discretization of time>

$${}^{t}\dot{\mathbf{U}} = \frac{1}{2\varDelta t}\left({}^{t+\varDelta t}\mathbf{U} - {}^{t-\varDelta t}\mathbf{U}\right) \qquad \text{(eq. 11.2)}$$

$${}^{t-\varDelta t/2}\dot{\mathbf{U}} = \frac{1}{\varDelta t}\left({}^{t}\mathbf{U} - {}^{t-\varDelta t}\mathbf{U}\right), \quad {}^{t+\varDelta t/2}\dot{\mathbf{U}} = \frac{1}{\varDelta t}\left({}^{t+\varDelta t}\mathbf{U} - {}^{t}\mathbf{U}\right)$$

$${}^{t}\ddot{\mathbf{U}} = \frac{1}{\varDelta t}\left({}^{t+\varDelta t/2}\dot{\mathbf{U}} - {}^{t-\varDelta t/2}\dot{\mathbf{U}}\right) = \frac{1}{(\varDelta t)^2}\left({}^{t+\varDelta t}\mathbf{U} - 2\,{}^{t}\mathbf{U} + {}^{t-\varDelta t}\mathbf{U}\right) \qquad \text{(eq. 11.3)}$$

Knowns: variables at time $t - \Delta t$, t

Unknowns: variables at time $t + \Delta t$

Substituting (eq. 11.2) and (eq. 11.3) into (eq. 11.1), the following equation is obtained

$$\left[\frac{1}{(\Delta t)^2}\mathbf{M} + \frac{1}{2\Delta t}\mathbf{C}\right]{}^{t+\Delta t}\mathbf{U} = {}^{t}\mathbf{R} - \left(\mathbf{K} - \frac{2}{(\Delta t)^2}\mathbf{M}\right){}^{t}\mathbf{U} - \left(\frac{1}{(\Delta t)^2}\mathbf{M} - \frac{1}{2\Delta t}\mathbf{C}\right){}^{t-\Delta t}\mathbf{U}$$

→ $$\boxed{\hat{\mathbf{M}}\,{}^{t+\Delta t}\mathbf{U} = {}^{t}\hat{\mathbf{R}}}$$

- In practice, $\mathbf{M}$ is lumped and $\mathbf{C}$ is mass-proportional. $\hat{\mathbf{M}}$ becomes a diagonal matrix and thus the inverse of $\hat{\mathbf{M}}$ is computationally cheap.

- In CDM, Δt must be smaller than a critical time step Δt_{cr}.

$$\Delta t \le \Delta t_{cr} \quad \text{with} \quad \Delta t_{cr} = \frac{T_N}{\pi} = \frac{2}{\omega_N}$$

T_N: the smallest period of the FE model with N DOFs

ω_N: the highest natural frequency of the FE model obtained solving $\mathbf{K}\boldsymbol{\phi} = \omega^2 \mathbf{M}\boldsymbol{\phi}$

- ω_N is bound by the highest natural frequency of all individual finite elements in the model.

- In practice, for most element types, the critical time step can be estimated by

$$\Delta t_{cr} = \frac{L}{c} \quad \text{with} \quad c = \sqrt{\frac{E}{\rho}}$$

(L: characteristic element length, c: material wave speed), and also

$$\Delta t_{cr} = \frac{L}{c} / \sqrt{1 + \frac{12I}{AL^2}} \quad \text{with } c = \sqrt{\frac{E}{\rho}} \quad \text{for beam elements,}$$

$$\Delta t_{cr} = \frac{L}{c} \quad \text{with } c = \sqrt{\frac{E}{\rho(1-v^2)}} \quad \text{for plate and shell elements,}$$

$$\Delta t_{cr} = \frac{L}{c} \quad \text{with } c = \sqrt{\frac{E(1-v)}{\rho(1+v)(1-2v)}} \quad \text{for 3D solid elements.}$$

II. Newmark method (Direct Integration – implicit scheme)

Consider the dynamic equilibrium at time $t + \Delta t$

$$\mathbf{M}\,{}^{t+\Delta t}\ddot{\mathbf{U}} + \mathbf{C}\,{}^{t+\Delta t}\dot{\mathbf{U}} + \mathbf{K}\,{}^{t+\Delta t}\mathbf{U} = {}^{t+\Delta t}\mathbf{R} \qquad \text{(eq. 11.4)}$$

Using "Taylor series expansion",

$${}^{t+\Delta t}\dot{\mathbf{U}} = {}^{t}\dot{\mathbf{U}} + \Delta t\left[(1-\delta)\,{}^{t}\ddot{\mathbf{U}} + \delta\,{}^{t+\Delta t}\ddot{\mathbf{U}}\right] \qquad \text{(eq. 11.5)}$$

$${}^{t+\Delta t}\mathbf{U} = {}^{t}\mathbf{U} + \Delta t\,{}^{t}\dot{\mathbf{U}} + (\Delta t)^2\left[\left(\frac{1}{2}-\delta\right){}^{t}\ddot{\mathbf{U}} + \alpha\,{}^{t+\Delta t}\ddot{\mathbf{U}}\right] \qquad \text{(eq. 11.6)}$$

When $\alpha = \frac{1}{4}$ and $\delta = \frac{1}{2}$, it is called "trapezoidal rule"

Knowns: variables at time $t - \Delta t$, t

Unknowns: variables at time $t + \Delta t$ (External force at $t + \Delta t$ is known)

Substituting (eq. 11.5) and (eq. 11.6) into (eq. 11.4), we obtain

$$\left[\mathbf{K}+\frac{1}{\alpha(\Delta t)^2}\mathbf{M}+\frac{\delta}{\alpha\Delta t}\mathbf{C}\right]{}^{t+\Delta t}\mathbf{U}={}^{t+\Delta t}\mathbf{R}+\mathbf{M}\left(\frac{1}{\alpha(\Delta t)^2}{}^{t}\mathbf{U}+\frac{1}{\alpha\Delta t}{}^{t}\dot{\mathbf{U}}+\frac{1-2\alpha}{2\alpha}{}^{t}\ddot{\mathbf{U}}\right)$$

$$+\mathbf{C}\left(\frac{\delta}{\alpha\Delta t}{}^{t}\mathbf{U}+\frac{\delta-\alpha}{\alpha}{}^{t}\dot{\mathbf{U}}+\frac{\Delta t}{2}\left(\frac{\delta}{\alpha}-2\right){}^{t}\ddot{\mathbf{U}}\right)$$

→ $\boxed{\hat{\mathbf{K}}\,{}^{t+\Delta t}\mathbf{U}={}^{t+\Delta t}\hat{\mathbf{R}}}$

- $\hat{\mathbf{K}}$ is not a diagonal matrix in general.

- Unconditionally stable: no condition on Δt for stability.

TABLE 9.1 *Step-by-step solution using central difference method (general mass and damping matrices)*

A. *Initial calculations:*

1. Form stiffness matrix $\mathbf{K}$, mass matrix $\mathbf{M}$, and damping matrix $\mathbf{C}$.
2. Initialize ${}^0\mathbf{U}$, ${}^0\dot{\mathbf{U}}$, and ${}^0\ddot{\mathbf{U}}$.
3. Select time step Δt, $\Delta t \le \Delta t_{cr}$, and calculate integration constants:

$$a_0 = \frac{1}{\Delta t^2}; \qquad a_1 = \frac{1}{2\,\Delta t}; \qquad a_2 = 2a_0; \qquad a_3 = \frac{1}{a_2}$$

4. Calculate ${}^{-\Delta t}\mathbf{U} = {}^0\mathbf{U} - \Delta t\,{}^0\dot{\mathbf{U}} + a_3\,{}^0\ddot{\mathbf{U}}$.
5. Form effective mass matrix $\hat{\mathbf{M}} = a_0\mathbf{M} + a_1\mathbf{C}$.
6. Triangularize $\hat{\mathbf{M}}$: $\hat{\mathbf{M}} = \mathbf{LDL}^T$.

B. *For each time step:*

1. Calculate effective loads at time t:

$${}^t\hat{\mathbf{R}} = {}^t\mathbf{R} - (\mathbf{K} - a_2\mathbf{M})\,{}^t\mathbf{U} - (a_0\mathbf{M} - a_1\mathbf{C})\,{}^{t-\Delta t}\mathbf{U}$$

2. Solve for displacements at time $t + \Delta t$:

$$\mathbf{LDL}^T\,{}^{t+\Delta t}\mathbf{U} = {}^t\hat{\mathbf{R}}$$

3. If required, evaluate accelerations and velocities at time t:

$${}^t\ddot{\mathbf{U}} = a_0({}^{t-\Delta t}\mathbf{U} - 2\,{}^t\mathbf{U} + {}^{t+\Delta t}\mathbf{U})$$

$${}^t\dot{\mathbf{U}} = a_1(-{}^{t-\Delta t}\mathbf{U} + {}^{t+\Delta t}\mathbf{U})$$

TABLE 9.3 *Step-by-step solution using Newmark integration method*

A. *Initial calculations:*

1. Form stiffness matrix $\mathbf{K}$, mass matrix $\mathbf{M}$, and damping constant $\mathbf{C}$.
2. Initialize ${}^{0}\mathbf{U}$, ${}^{0}\dot{\mathbf{U}}$, and ${}^{0}\ddot{\mathbf{U}}$.
3. Select time step Δt and parameters α and δ and calculate integration constants:

$$\delta \geq 0.50; \qquad \alpha \geq 0.25(0.5+\delta)^2$$

4. $$a_0 = \frac{1}{\alpha \Delta t^2}; \quad a_1 = \frac{\delta}{\alpha \Delta t}; \quad a_2 = \frac{1}{\alpha \Delta t}; \quad a_3 = \frac{1}{2\alpha} - 1;$$

$$a_4 = \frac{\delta}{\alpha} - 1; \quad a_5 = \frac{\Delta t}{2}\left(\frac{\delta}{\alpha} - 2\right); \quad a_6 = \Delta t(1-\delta); \quad a_7 = \delta \Delta t$$

5. Form effective stiffness matrix $\hat{\mathbf{K}} : \hat{\mathbf{K}} = \mathbf{K} + a_0\mathbf{M} + a_1\mathbf{C}$.
6. Triangularize $\hat{\mathbf{K}} : \hat{\mathbf{K}} = \mathbf{LDL}^T$.

B. *For each time step:*

1. Calculate effective loads at time $t + \Delta t$:

$${}^{t+\Delta t}\hat{\mathbf{R}} = {}^{t+\Delta t}\mathbf{R} + \mathbf{M}\left(a_0\,{}^{t}\mathbf{U} + a_2\,{}^{t}\dot{\mathbf{U}} + a_3\,{}^{t}\ddot{\mathbf{U}}\right) + \mathbf{C}\left(a_1\,{}^{t}\mathbf{U} + a_4\,{}^{t}\dot{\mathbf{U}} + a_5\,{}^{t}\ddot{\mathbf{U}}\right)$$

2. Solve for displacements at time $t + \Delta t$:

$$\mathbf{LDL}^T\;{}^{t+\Delta t}\mathbf{U} = {}^{t+\Delta t}\hat{\mathbf{R}}$$

3. Calculate accelerations and velocities at time $t + \Delta t$:

$${}^{t+\Delta t}\ddot{\mathbf{U}} = a_0\left({}^{t+\Delta t}\mathbf{U} - {}^{t}\mathbf{U}\right) - a_2\,{}^{t}\dot{\mathbf{U}} - a_3\,{}^{t}\ddot{\mathbf{U}}$$

$${}^{t+\Delta t}\dot{\mathbf{U}} = {}^{t}\dot{\mathbf{U}} + a_6\,{}^{t}\ddot{\mathbf{U}} + a_7\,{}^{t+\Delta t}\ddot{\mathbf{U}}$$

III. Mode superposition

Dynamic equilibrium

$$\mathbf{M}\ddot{\mathbf{U}}(t)+\mathbf{C}\dot{\mathbf{U}}(t)+\mathbf{K}\mathbf{U}(t)=\mathbf{R}(t)$$

Coordinate transformation

$\mathbf{U}(t)=\mathbf{P}\mathbf{X}(t)$ with $\mathbf{P}$: transformation matrix

$$\mathbf{X}(t)=\begin{Bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_N(t) \end{Bmatrix}$$: time dependent vector (generalized coordinates)

Rayleigh-Ritz procedure

$$\mathbf{P}^{\mathrm{T}}\mathbf{M}\mathbf{P}\ddot{\mathbf{X}}+\mathbf{P}^{\mathrm{T}}\mathbf{C}\mathbf{P}\dot{\mathbf{X}}+\mathbf{P}^{\mathrm{T}}\mathbf{K}\mathbf{P}\mathbf{X}=\mathbf{P}^{\mathrm{T}}\mathbf{R}$$

$\rightarrow$ $\hat{\mathbf{M}}\ddot{\mathbf{X}}+\hat{\mathbf{C}}\dot{\mathbf{X}}+\hat{\mathbf{K}}\mathbf{X}=\hat{\mathbf{R}}$ (eq. 11.7)

What $\mathbf{P}$ is used?

$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{0}$: free vibration equilibrium without damping

Solution: $\mathbf{U}(t) = \boldsymbol{\phi} \sin \omega (t - t_0)$

$\mathbf{K}\boldsymbol{\phi} = \omega^2 \mathbf{M}\boldsymbol{\phi}$: eigenvalue problem

Eigen solutions: eigenvalues (natural frequencies); $\omega_1 < \omega_2 < \cdots\cdots < \omega_N$

eigenvectors (mode shapes); $\boldsymbol{\phi}_1, \boldsymbol{\phi}_2, \cdots\cdots \boldsymbol{\phi}_N$

$\mathbf{P} = \boldsymbol{\Phi} = [\boldsymbol{\phi}_1 \quad \boldsymbol{\phi}_2 \quad \cdots \quad \boldsymbol{\phi}_N]$: eigenvector matrix

Let us use $\mathbf{\Phi}$ in (eq. 11.7).

$$\hat{\mathbf{M}} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{M}\mathbf{\Phi} = \mathbf{I} = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}$$ (mass orthonormality)

$$\hat{\mathbf{K}} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{K}\mathbf{\Phi} = \mathbf{\Omega}^2 = \begin{bmatrix} \omega_1^2 & & & \\ & \omega_2^2 & & \\ & & \ddots & \\ & & & \omega_N^2 \end{bmatrix}$$ (stiffness orthogonality)

→ $\boxed{\ddot{\mathbf{X}} + \mathbf{\Phi}^{\mathrm{T}}\mathbf{C}\mathbf{\Phi}\dot{\mathbf{X}} + \mathbf{\Omega}^2\mathbf{X} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{R}}$ with initial conditions ${}^0\mathbf{X} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{M}\,{}^0\mathbf{U}$ and ${}^0\dot{\mathbf{X}} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{M}\,{}^0\dot{\mathbf{U}}$

Assume that

$$\hat{\mathbf{C}} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{C}\mathbf{\Phi} = \begin{bmatrix} 2\omega_1\xi_1 & & & \\ & 2\omega_2\xi_2 & & \\ & & \ddots & \\ & & & 2\omega_N\xi_N \end{bmatrix}$$

, ξ_i: modal damping parameters.

$$\hat{\mathbf{R}} = \mathbf{\Phi}^{\mathrm{T}}\mathbf{R} = \begin{bmatrix} r_1(t) \\ r_2(t) \\ \vdots \\ r_N(t) \end{bmatrix}$$

$$\rightarrow \left.\begin{matrix} \ddot{x}_1(t) + 2\omega_1\xi_1\dot{x}_1(t) + \omega_1^2 x_1(t) = r_1(t) \\ \ddot{x}_2(t) + 2\omega_2\xi_2\dot{x}_2(t) + \omega_2^2 x_2(t) = r_2(t) \\ \vdots \\ \ddot{x}_N(t) + 2\omega_N\xi_N\dot{x}_N(t) + \omega_N^2 x_N(t) = r_N(t) \end{matrix}\right\}$$

"decoupled" N single DOF problems

Using "Duhamel integral", we can solve each single DOF problem.

$$x_i(t) = \frac{1}{\bar{\omega}_i}\int_0^t r_i(\tau) e^{-\xi_i \omega_i (t-\tau)} \sin \bar{\omega}_i (t-\tau) d\tau + e^{-\xi_i \omega_i t}\left(\alpha_i \sin \bar{\omega}_i t + \beta_i \cos \bar{\omega}_i t\right) \quad \text{with} \quad \bar{\omega}_i = \omega_i \sqrt{1-\xi_i^2}.$$

Then, $\mathbf{U}(t) = \mathbf{\Phi}\ \mathbf{X}(t)$.

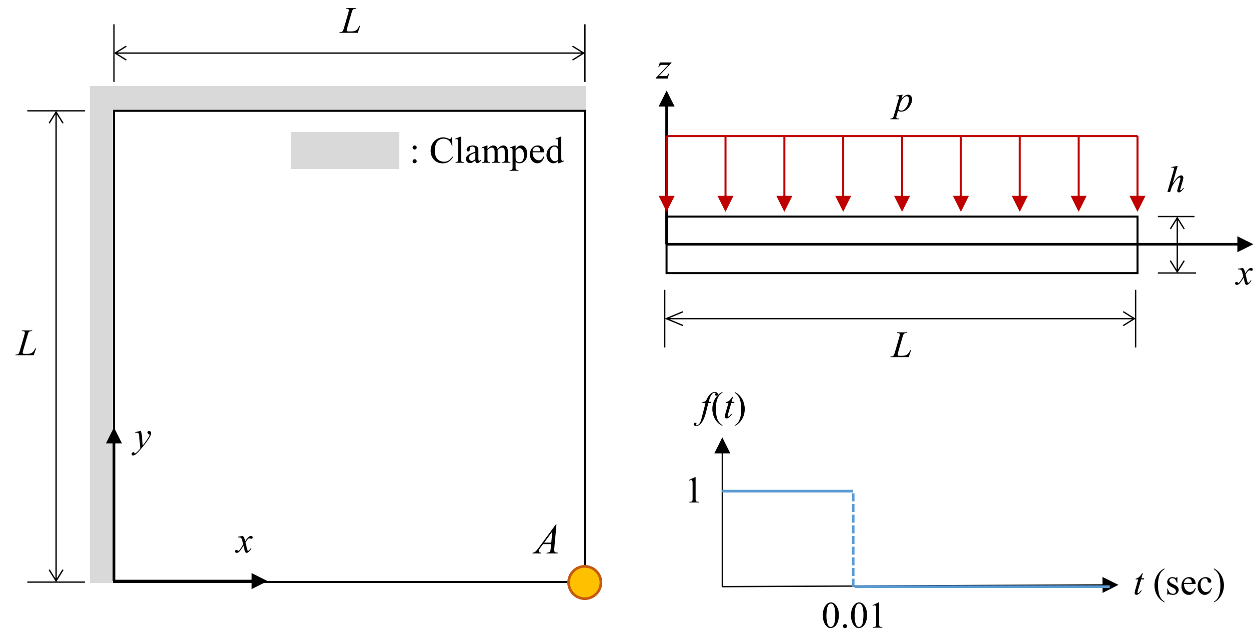
Note that, in this method, we do not need to include all eigenvectors in the transformation matrix $\mathbf{\Phi}$.

$$\mathbf{U}(t) = \mathbf{\Phi}\ \mathbf{X}(t),$$

$$\mathbf{U}(t) = \begin{bmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_N \end{bmatrix} \mathbf{X}(t) = \sum_{i=1}^{N} \boldsymbol{\phi}_i x_i(t) \quad \text{(when all eigenvectors are included)}$$

$$\mathbf{U}(t) \approx \begin{bmatrix} \boldsymbol{\phi}_1 & \boldsymbol{\phi}_2 & \cdots & \boldsymbol{\phi}_M \end{bmatrix} \mathbf{X}(t) = \sum_{i=1}^{M} \boldsymbol{\phi}_i x_i(t), \quad M \ll N \quad \text{(for many practical cases)}$$

Example – Partly clamped plate subjected to a pressure



Young's modulus : $E = 200\text{GPa}$

Poisson's ratio : $\nu = 0.3$

Density : $\rho = 7850\text{kg/m}^3$

Thickness : $h = 0.01\text{m}$

Length : $L = 1\text{m}$

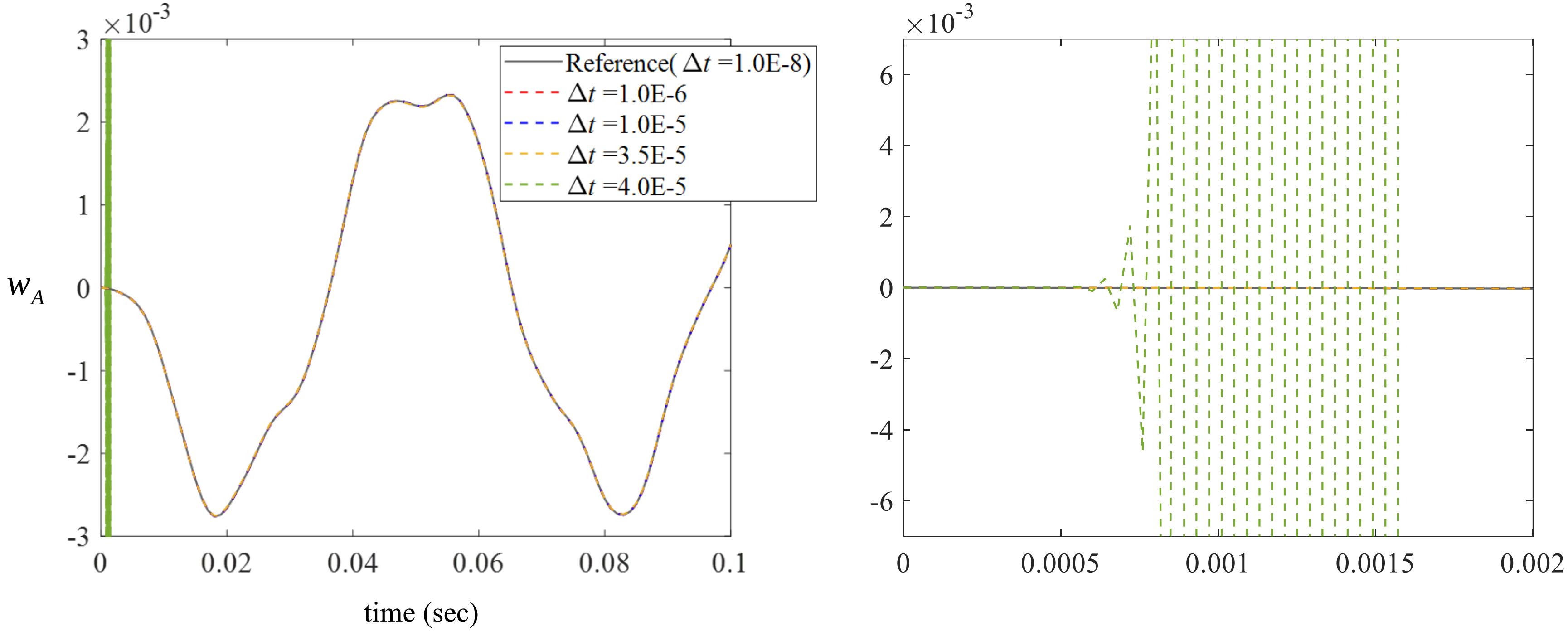
Pressure : $p(t) = 10^3 f(t)\text{N/m}^2$

5x5 mesh of 4-node shell elements

Dynamic analysis is performed using the following three methods.

- Direct integration (explicit, CDM)
- Direct integration (implicit, Newmark)
- Mode superposition

Direct integration (Explicit, Central difference method)

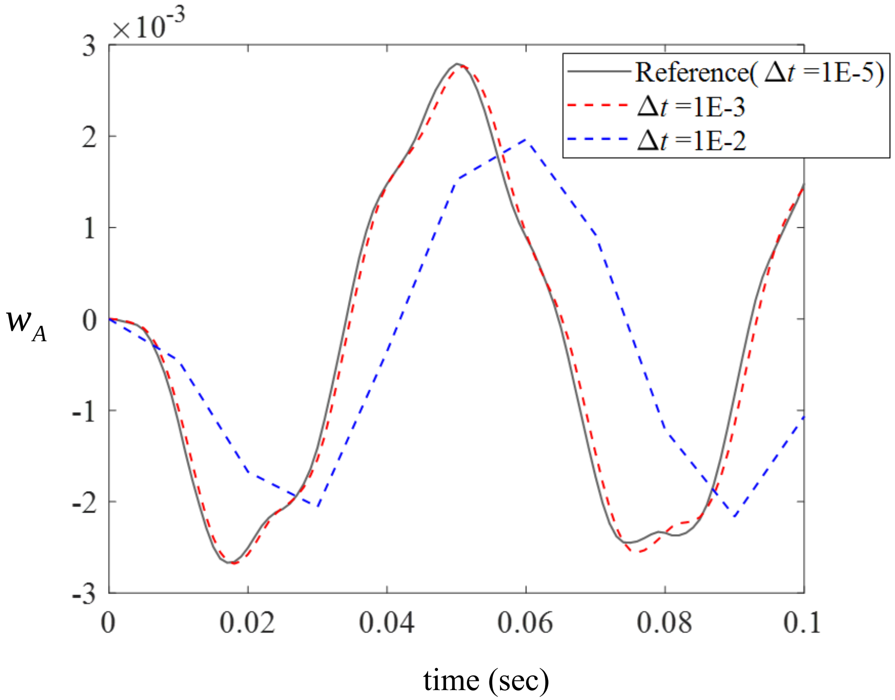


Lumped mass is used. (Using a consistent mass takes too much computation time)

Critical time step: $\Delta t_{cr} \approx 3.8\text{E}-5$

- The time step should be smaller than the critical time step: $\Delta t < \Delta t_{cr}$
- For the plate and shell elements, $\Delta t_{cr} = L/c$

Direct integration (Implicit, Newmark method)



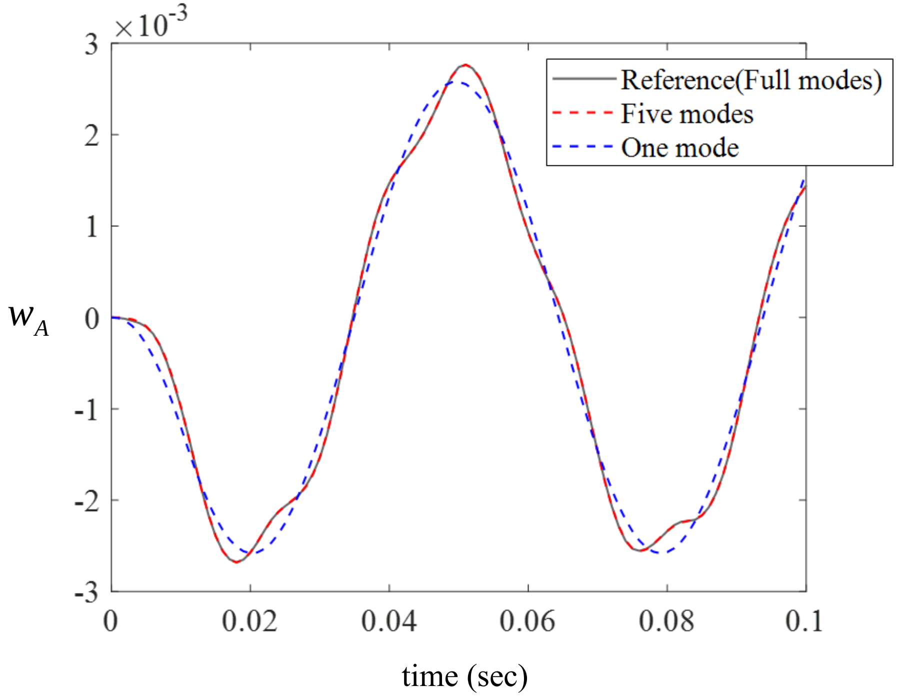
Consistent mass is used.

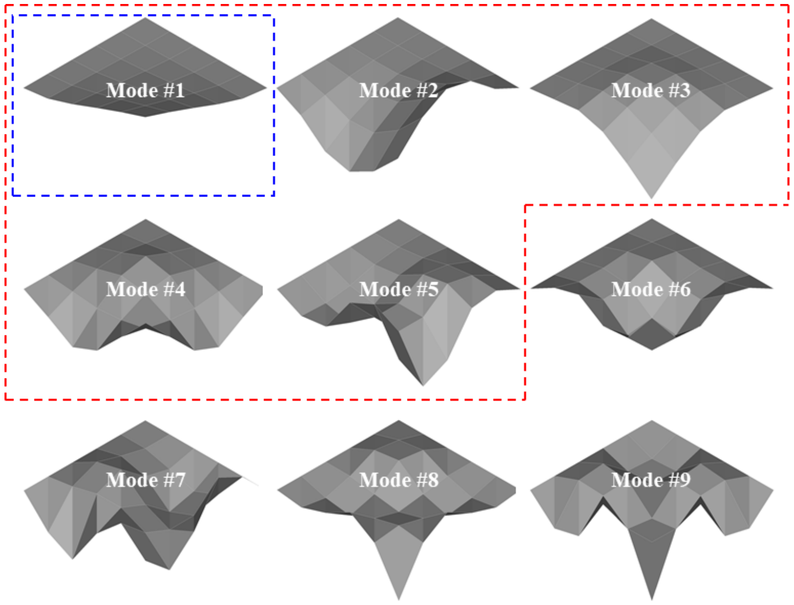
Newmark parameters: $\alpha = 0.25$ and $\delta = 0.5$

Unconditionally stable

- More accurate solution with a smaller time step

Mode superposition





Degrees of freedom (DOFs) : 125

Consistent mass is used.

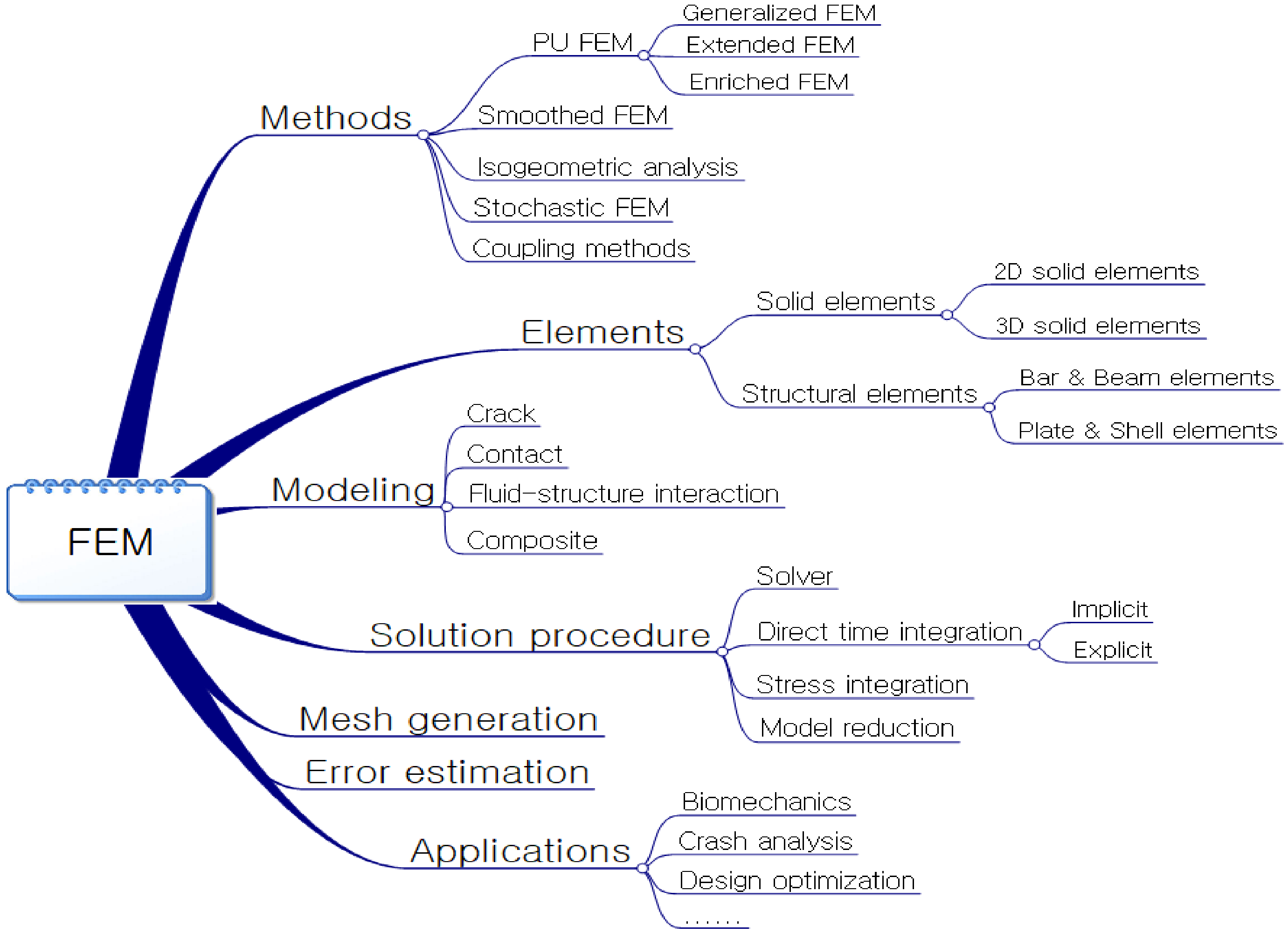
We can predict the response by using only a few modes

- More accurate solution with more modes

References

- KJ Bathe, Finite element procedures, 1996, 2014
- RW Clough, "Thoughts about the origin of the finite element methods", Computers & Structure, 2001
- J Fish, T Belytschko, A first course in finite elements, 2007
- KJ Bathe, Lecture notes, Finite element analysis of solids and fluids
- KJ Bathe, Lecture notes, Computer methods in dynamics
- PS Lee, Lecture notes, Advanced analysis of solids and structures
- PS Lee, Lecture notes, Finite element analysis of structures
- PS Lee, Lecture notes, Engineering design via FEM
- PS Lee, "Past, present and future of the finite element method", KSME 2018.
- ADINA structures – Theory and modeling guide

A1. Research Area Map



A2. Implementation of FE Code

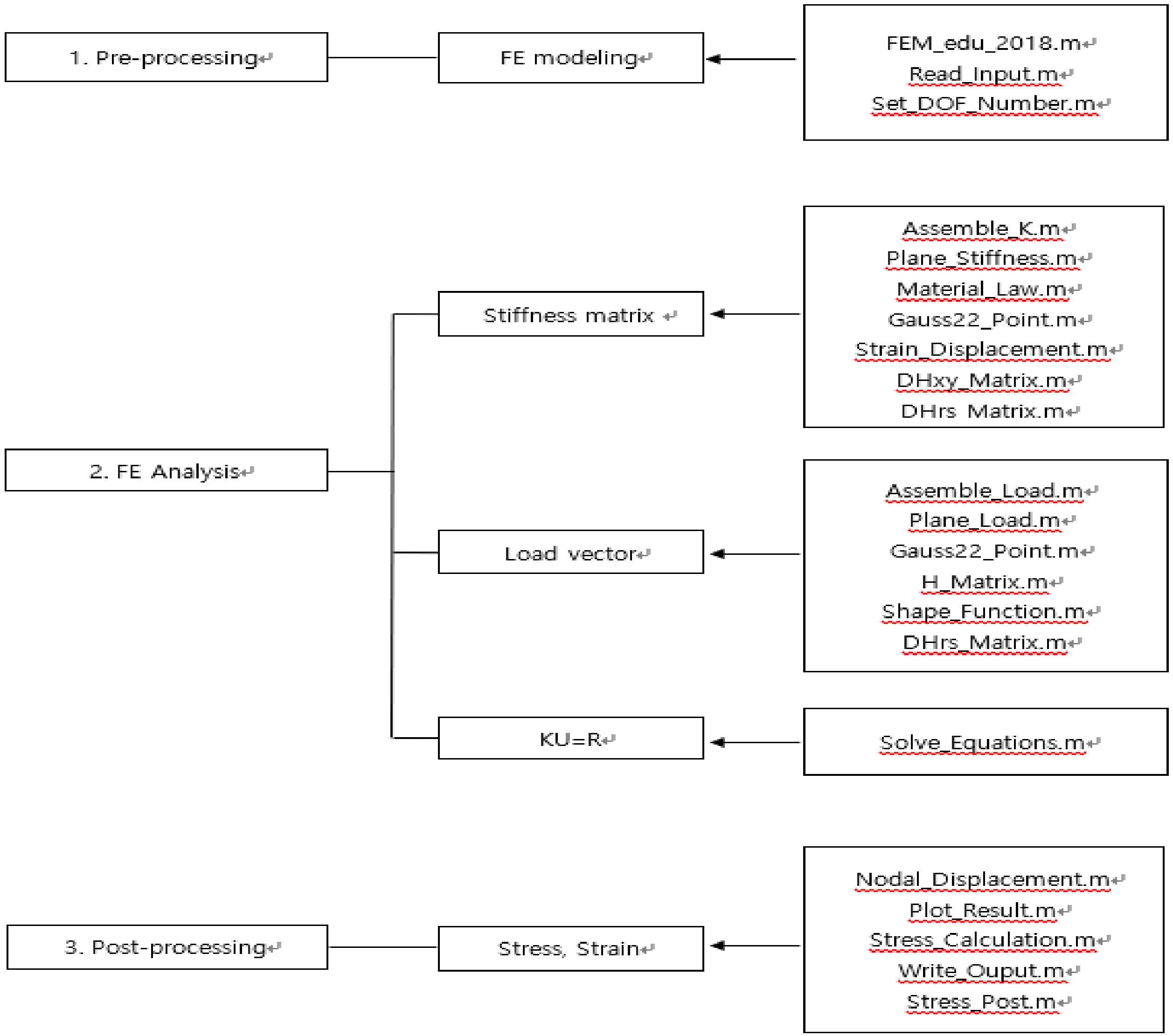
FEM_edu_2018.m : Main program

- Read_Input.m : Read the input file
- Set_DOF_Number.m : Set DOF numbers (numbering)
- Assemble_K.m : Assemble the total stiffness matrix using Direct Stiffness Method
- Assemble_Load.m : Assemble the load vector using Direct Stiffness Method
- Solve_Equations.m : Solve the linear equations KU=R

- Plane_Stiffness.m : Calculate the element stiffness matrix
- Plane_Load.m : Calculate the element load vector
  - Gauss22_Point.m : Give positions and weight factors at Gauss points
  - Material_Law.m : Calculate the material law matrix
  - DHrs_Matrix.m : Calculate the derivatives of displacements with respect to r and s
  - DHxy_Matrix.m : Calculate the derivatives of displacements with respect to x and y
  - H_Matrix.m : Calculate the displacement interpolation matrix
  - Shape_Function.m : Evaluate the shape functions
  - Strain_Displacement.m : Calculate the strain-displacement matrix

- Nodal_Displacement.m : Find the nodal displacements
- Stress_Calculation.m : Calculate the element stresses
- Write_Ouput.m : Write the output file
- Stress_Post.m : Write the output file for plotting deformed shapes and stress contours
- Plot_Result.m : Plot the deformed shapes and stress contours

FE code flow chart

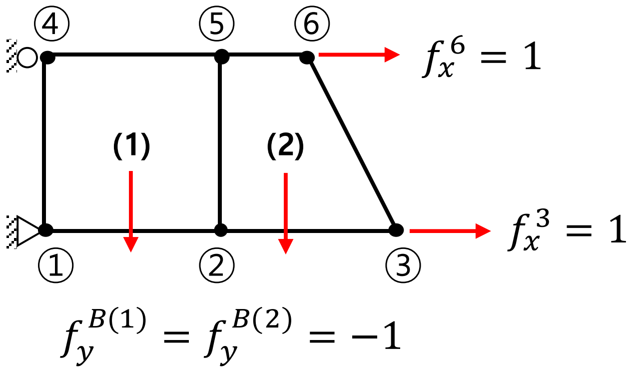


input.txt

```
Nodes
6
#    x   y    u v   px  py
1    0.0 0.0   1 1   0.0 0.0
2    1.0 0.0   0 0   0.0 0.0
3    2.0 0.0   0 0   1.0 0.0
4    0.0 1.0   1 0   0.0 0.0
5    1.0 1.0   0 0   0.0 0.0
6    1.5 1.0   0 0   1.0 0.0

Elements
2
#   connectivity   fx   fy
1    5  4  1  2    0.0  -1.0
2    6  5  2  3    0.0  -1.0

Thickness
1.0
Young
1.0e5
Poisson
0.3e0
```



FEM_edu_2018.m

```
% ----------------------------------------------------------------------------
% KAIST ME535, Finite Element Analysis of Structures
% FEM code for educational purpose (2D plane stress analysis)
% PS Lee  1998-02-19
% HD Seo  2017-10-01
% HT Jung 2018-10-30
% ----------------------------------------------------------------------------
% node_n                : Number of nodes
% element_n             : Number of elements

% node_x(node_n,2)        : Nodal postions
% node_bc(node_n,6)       : Nodal displacement BC
% node_pm(node_n,6)       : Nodal force BC
% element_cn(element_n,4)  : Element connectivity
% element_q(element_n,2)   : Uniformly distributed load on element
% element_thickness       : Thickness
% material_young          : Young's modulus
% material_poisson        : Poisson's ratio

% total_dof_n            : Number of total DOFs
% free_dof_n             : Number of free DOFs
% fixed_dof_n            : Number of fixed DOFs
% node_eqn(node_n,6)      : Equation numbers assigned at nodes

% K(free_dof_n,free_dof_n) : Stiffness matrix
% R(free_dof_n)           : Load vector
% U(free_dof_n)           : Displacement vector

% n_displace(node_n,2)    : Nodal displacement matrix (obtained from U)
% ----------------------------------------------------------------------------

clc;
clear all;

% [1] Read "input.txt" file
[node_x, node_bc, node_pm, element_cn, element_q, element_thickness, material_young, material_poisson, node_n, element_n] = Read_Input();
disp('[1/6] Read input')

% [2] Calculate # of total DOFs, # of free DOFs, # of fixed DOFs
[total_dof_n, free_dof_n, fixed_dof_n, node_eqn] = Set_DOF_Number(node_n, node_bc);
disp('[2/6] Assign equation numbers')

% [3] Calculate K (stiffness matirx)
[K]=Assemble_K(free_dof_n, element_n, node_x, element_cn, material_young, material_poisson, element_thickness, node_eqn);
disp('[3/6] Caluate K (stiffness matrix)')

% [4] Calculate R (load vector)
```

```
[R]=Assemble_Load(node_n, element_n, node_x, element_cn, element_q, node_eqn, node_pm);
disp('[4/6] Caculate R (load vector)')

% [5] Solve KU=R (linear equations) and find U (nodal DOFs vector)
[U] = Solve_Equations(K, R, free_dof_n);
disp('[5/6] Calculate U (displacement)')

% [6] Post process
[n_displace] = Nodal_Displacement(node_n, node_eqn, free_dof_n, U);
Plot_Result(n_displace, node_x, element_n, element_cn, material_young, material_poisson);
Write_Output(U, K, node_n, node_bc, n_displace, element_n, node_x, element_cn, material_young, material_poisson)
disp('[6/6] Post process')
```

Read_Input.m

```
% -------------------------------------------------------------------------
% Read "input.txt" file
% -------------------------------------------------------------------------
% node_n               : # of nodes
% element_n             : # of elements

% node_x(node_n,2)       : Nodal positions
% node_bc(node_n,6)      : Nodal displacement BC
% node_pm(node_n,6)      : Nodal force BC
% element_cn(element_n,4) : Element connectivity
% element_q(element_n,2)  : Uniformly distributed load on element
% element_thickness      : Element thickness
% material_young         : Young's modulus
% material_poisson       : Poisson's ratio
% -------------------------------------------------------------------------

function [node_x, node_bc, node_pm, element_cn, element_q, element_thickness, material_young, material_poisson, node_n, element_n]=Read_Input()

% read input file : input.txt
fid = fopen('input.txt','r');
bufs = fscanf(fid,'%s',1);
node_n = fscanf(fid,'%g',1);
bufs = fscanf(fid,'%s',7);

% read node information
for i=1:1:node_n
   nodes(:,i) = fscanf(fid,'%*g %g %g %g %g %g %g\n',6);
   node_x(i,1) = nodes(1,i);
   node_x(i,2) = nodes(2,i);
   node_bc(i,1) = nodes(3,i);
   node_bc(i,2) = nodes(4,i);
   node_bc(i,3:6) = 1;
   node_pm(i,1) = nodes(5,i);
   node_pm(i,2) = nodes(6,i);
   node_pm(i,3:6) = 0;
end

% read element information
bufs = fscanf(fid,'%s',1);
element_n = fscanf(fid,'%g',1);
bufs = fscanf(fid,'%s',4);
for i=1:1:element_n
   elements(:,i) = fscanf(fid,'%*g %g %g %g %g %g %g \n',6);
   element_cn(i,1) = elements(1,i);
   element_cn(i,2) = elements(2,i);
```

```
    element_cn(i,3) = elements(3,i);
    element_cn(i,4) = elements(4,i);
    element_q(i,1) = elements(5,i);
    element_q(i,2) = elements(6,i);
end

% read material information
bufs = fscanf(fid,'%s',1);
element_thickness = fscanf(fid,'%g',1);
bufs = fscanf(fid,'%s',1);
material_young = fscanf(fid,'%g',1);
bufs = fscanf(fid,'%s',1);
material_poisson = fscanf(fid,'%g',1);

fclose(fid);
end
```

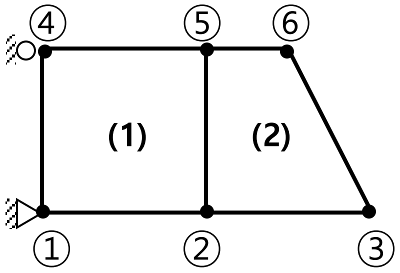
Set_DOF_Number.m

```
% -------------------------------------------------------------------------
% Assign equation number
% -------------------------------------------------------------------------
% tn                 : # of total DOF
% fn                 : # of free DOF
% cn                 : # of fixed DOF
% node_eqn(node_n,6) : Equation numbers assigned at nodes
% -------------------------------------------------------------------------

function [tn, fn, cn, node_eqn]=Set_DOF_Number(node_n,Node_bc)

tn = node_n * 6;
fn = 0; cn = 0;
for i=1:1:node_n
    for j=1:1:6
        if  Node_bc(i,j) == 0
            fn = fn + 1;
            node_eqn(i,j) = fn;
        else
            cn = cn + 1;
            node_eqn(i,j) = tn - cn + 1;
        end
    end
end

end
```



Node	u	v	w	θ_x	θ_y	θ_z
1	36	35	34	33	32	31
2	1	2	30	29	28	27
3	3	4	26	25	24	23
4	22	5	21	20	19	18
5	6	7	17	16	15	14
6	8	9	13	12	11	10

Gauss22_Point.m

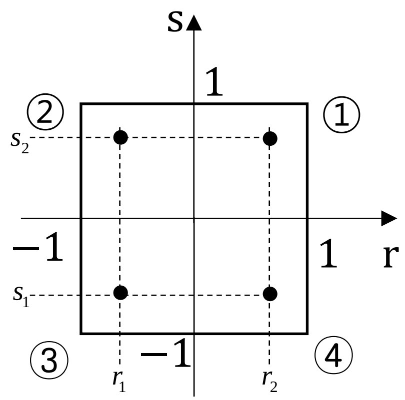
```
% ---------------------------------------------------------------------------
% Gauss integration
% ---------------------------------------------------------------------------

function [r, s, weight]=Gauss22_Point(i,j)

   gauss_point(1) = -0.577350269;
   gauss_point(2) = 0.577350269;
   w(1) = 1.0;
   w(2) = 1.0;

   weight = w(i)*w(j);
   r = gauss_point(i);
   s = gauss_point(j);

end
```



Shape_Function.m

```
% ---------------------------------------------------------------------------
% Construct H matrix
% ---------------------------------------------------------------------------
% Position of node & shape function
%              s
%         2----|----1
%         |    |    |
%        ------+-------r
%         |    |    |
%         3----|----4
% ---------------------------------------------------------------------------

function [H]=Shape_Function(r, s)

   H(1) = 0.25 * (1.0 + r) * (1.0 + s);
   H(2) = 0.25 * (1.0 - r) * (1.0 + s);
   H(3) = 0.25 * (1.0 - r) * (1.0 - s);
   H(4) = 0.25 * (1.0 + r) * (1.0 - s);

end
```

DHrs_Matrix.m

```
% -------------------------------------------------------------------------
% Calculate dH/dr and dH/ds
% -------------------------------------------------------------------------
% dHrs(2,4) : dH/dr(1,:) and dH/ds(2,:)
% -------------------------------------------------------------------------

function [dHrs]=DHrs_Matrix(r, s)

   dHrs(1,1) =  0.25 * (1.0 + s);
   dHrs(1,2) = -0.25 * (1.0 + s);
   dHrs(1,3) = -0.25 * (1.0 - s);
   dHrs(1,4) =  0.25 * (1.0 - s);

   dHrs(2,1) =  0.25 * (1.0 + r);
   dHrs(2,2) =  0.25 * (1.0 - r);
   dHrs(2,3) = -0.25 * (1.0 - r);
   dHrs(2,4) = -0.25 * (1.0 + r);

end
```

$$\frac{\partial h_i}{\partial r}$$

$$\frac{\partial h_i}{\partial s}$$

DHxy_Matrix.m

```
% -------------------------------------------------------------------------
% Calculate dH/dx and dH/dy
% -------------------------------------------------------------------------
% dHrs(2,4)  : dH/dr(1,:) and dH/ds(2,:)
% jacob(2,2) : Jacobian matrix
% det_j      : Determinant of Jacobian
% dHxy(2,4)  : dH/dx(1,:) and dH/dy(2,:)
% -------------------------------------------------------------------------

function [det_j, dHxy]=DHxy_Matrix(r, s, node)

   [dHrs] = DHrs_Matrix(r,s);
   jacob = dHrs * node;
   det_j = jacob(1,1) * jacob(2,2) - jacob(1,2) * jacob(2,1);

   jacob(1,2) = -jacob(1,2);
   jacob(2,1) = -jacob(2,1);
   buf = jacob(1,1);
   jacob(1,1) = jacob(2,2);
   jacob(2,2) = buf;
   jacob = jacob / det_j;

   dHxy = jacob * dHrs;

end
```

$$\begin{pmatrix} \dfrac{\partial h_i}{\partial x} \\ \dfrac{\partial h_i}{\partial y} \end{pmatrix} = \mathbf{J}^{-1} \begin{pmatrix} \dfrac{\partial h_i}{\partial r} \\ \dfrac{\partial h_i}{\partial s} \end{pmatrix}$$

H_Matrix.m

```
% ---------------------------------------------------------------------------
% H matrix and determinant of Jacobian
% ---------------------------------------------------------------------------
% jacob(2,2) : Jacobian matrix
% ---------------------------------------------------------------------------

function [H, det_j]=H_Matrix(r, s, node)

   % shape function (H)
   [H]=Shape_Function(r, s);

   % dHrs = deriavtive of shape functions (dH/dr, dH/ds)
   [dHrs]=DHrs_Matrix(r, s);

   % Jacob = Jacobian matrix
   jacob = dHrs * node;

   % determinant of Jacobian matrix
   det_j = jacob(1,1) * jacob(2,2) - jacob(1,2) * jacob(2,1);

end
```

Strain_Displacement.m

```
% ---------------------------------------------------------------------------
% Calculate B matrix
% ---------------------------------------------------------------------------
% det_j     : Determinant of Jacobian
% dHxy(2,4) : dH/dx(1,:) and dH/dy(2,:)
% node(4,2) : Nodal positions of element
% ---------------------------------------------------------------------------

function [det_j, B]=Strain_Displacement(r, s, node)

   [det_j, dHxy] = DHxy_Matrix(r, s, node);

   B = 0;
   B(1,1:4)=dHxy(1,1:4);

   B(2,5:8)=dHxy(2,1:4);

   B(3,1:4)=dHxy(2,1:4);
   B(3,5:8)=dHxy(1,1:4);


end
```

$$\mathbf{B}^{(m)} = \begin{bmatrix} \dfrac{\partial h_i}{\partial x} & 0 \\ 0 & \dfrac{\partial h_i}{\partial y} \\ \dfrac{\partial h_i}{\partial y} & \dfrac{\partial h_i}{\partial x} \end{bmatrix}$$

Material_Law.m

```
% --------------------------------------------------------------------------
% Construct Material_law matrix
% --------------------------------------------------------------------------
% young   : Young's modulus
% poisson : Poisson's ratio
% --------------------------------------------------------------------------

function [C]=Material_Law(young, poisson)

   C(1,1) = young / (1.0 - poisson^2);
   C(2,2) = C(1,1);
   C(1,2) = poisson * C(1,1);
   C(2,1) = C(1,2);
   C(3,3) = 0.5 * young / (1.0 + poisson);

End
```

$$\mathbf{C}^{(m)} = \begin{bmatrix} \frac{E}{1-v^2} & \frac{vE}{1-v^2} & \\ \frac{vE}{1-v^2} & \frac{E}{1-v^2} & \\ & & \frac{E}{2(1+v)} \end{bmatrix}$$

Plane_Stiffness.m

```
% --------------------------------------------------------------------------
% Calculate local stiffness matrix
% --------------------------------------------------------------------------
% C(3,3)    : Material law matrix
% det_j     : Determinant of Jacobian
% Ke(8,8)   : Stiffness of element
% node(4,2) : Nodal positions of element
% --------------------------------------------------------------------------

function [Ke]=Plane_Stiffness(young, poisson, thick, node)
% Calculate material matrix
[C] = Material_Law(young, poisson);
Ke  = zeros(8,8);

% 2 by 2 Gauss integration
for i=1:1:2
   for j=1:1:2

      % Gauss interation point and weight factor
      [r, s, weight]=Gauss22_Point(i,j);

      % calculate determinant of Jacobian and B-matrix
      [det_j, B]=Strain_Displacement(r, s, node);

      % integration
      Ke = Ke + weight * thick * transpose(B)*C*B * det_j;
   end
end
end
```

$$\mathbf{K}^{(m)} = \int_{V^{(m)}} \mathbf{B}^{(m)T} \mathbf{C}^{(m)} \mathbf{B}^{(m)} dV$$

$$\mathbf{B}_{ij} = \mathbf{B}\left(r_i, r_j\right)$$

$$\left(\det \mathbf{J}\right)_{ij} = \det \mathbf{J}\left(r_i, r_j\right)$$

$$\mathbf{K}^{(m)} = t \sum w_{ij} \mathbf{B}_{ij}^{(m)T} \mathbf{C}^{(m)} \mathbf{B}_{ij}^{(m)} \left(\det \mathbf{J}\right)_{ij}$$

Plane_Load.m

```
% ----------------------------------------------------------------------------
% Calculate load vector
% ----------------------------------------------------------------------------
% nodal_load(8,1) : Nodal load vector of element
% det_j           : Determinant of Jacobian
% q(2)            : Nodal load of element
% node(4,2)       : Nodal positions of element
% ----------------------------------------------------------------------------

function [nodal_load]=Plane_Load(node, q)

   nodal_load=zeros(8,1);

   % numerical integration
   for i=1:1:2
     for j=1:1:2

        % Gauss points
        [r, s, weight]=Gauss22_Point(i,j);

        % determinant of Jacobian and shape function (H)
        [H, det_j]=H_Matrix(r, s, node);

        % equivalent nodal load vector
        nodal_load(1:4) = nodal_load(1:4) + weight*abs(det_j)*H(:)*q(1);
        nodal_load(5:8) = nodal_load(5:8) + weight*abs(det_j)*H(:)*q(2);
     end
   end
end
```

$$\mathbf{R}^{(m)} = \int_{V^{(m)}} \mathbf{H}^{(m)T}\mathbf{f}^{(m)} dV$$

$$= \sum w_{ij} \mathbf{H}_{ij}^{(m)T}\mathbf{f}^{(m)} \left(\det \mathbf{J}\right)_{ij}$$

Assemble_K.m

```
% -------------------------------------------------------------------------
% Assemble stiffness matrix
% -------------------------------------------------------------------------
% K(free_dof_n,free_dof_n) : Total stiffness matrix
% free_dof_n              : # of free DOFs
% element_n               : # of elements
% node_x(node_n,2)         : Nodal positions
% element_cn(element_n,4)  : Element connectivity
% material_young           : Young's modulus
% material_poisson         : Poisson's ratio
% element_thickness        : Element thickness
% node_eqn(node_n,6)       : Equation numbers assigned at nodes
% eqn(8)                  : Equation number vector of element
% enode(4,2)              : Nodal positions of element
% -------------------------------------------------------------------------
function [K]=Assemble_K(free_dof_n, element_n, node_x, element_cn, material_young, material_poisson, element_thickness, node_eqn)
K = zeros(free_dof_n);
fid = fopen('stiffness.txt','w');
for i=1:element_n
   % extract element nodal position
   for j=1:1:4
      for k=1:1:2
         enode(j,k) = node_x(element_cn(i,j),k);
      end
   end
   % calculate Ke (stiffness of each element)
   [Ke]=Plane_Stiffness(material_young, material_poisson, element_thickness, enode);
   fprintf(fid,'\r\n Siffness matrix of element : ');
   fprintf(fid,'%g \r\n',i);
   fprintf(fid, '----------------------------------------------\r\n');
   for j=1:1:8
      fprintf(fid,'%+e %+e %+e %+e %+e %+e %+e %+e \r\n',Ke(j,:));
   end
   eqn(1:4) = node_eqn(element_cn(i,1:4),1);
   eqn(5:8) = node_eqn(element_cn(i,1:4),2);       Local DOF → Global DOF

   % assemble total stiffness matrix
   for j=1:8
      for k=1:8
         if (eqn(j)<=free_dof_n) && (eqn(k)<=free_dof_n)
            K(eqn(j),eqn(k))=K(eqn(j),eqn(k))+Ke(j,k);        ← Direct stiffness method
         end
      end
   end
end
fclose(fid);
end
```

Assemble_Load.m

```
% -------------------------------------------------------------------------
% Assemble Load vector
% -------------------------------------------------------------------------
% node_n                 : # of nodes
% element_n              : # of elements
% node_x(node_n,2)        : Nodal postions
% element_cn(element_n,4)  : Element connectivity
% element_q(element_n,2)   : Uniformly distributed load on element
% node_eqn(node_n,6)      : Equation numbers assigned at nodes
% node_pm(node_n,6)       : Nodal force BC
% enode(4,2)             : Nodal positions of element
% -------------------------------------------------------------------------

function [R]=Assemble_Load(node_n, element_n, node_x, element_cn, element_q, node_eqn, node_pm)

   % assemble load vector for nodal load
   for i=1:1:node_n
     for j=1:1:2
        R(node_eqn(i,j)) = node_pm(i,j);
     end
   end

   % assemble load vector for body force
   for i=1:1:element_n

     % nodal position of element
     for j=1:1:4
        for k=1:1:2
            enode(j,k) = node_x(element_cn(i,j),k);
        end
     end

     % calculate equivalent nodal load from body force
     [nodal_load]=Plane_Load(enode,element_q(i,:));

     % assemble load vector
     for j=1:1:2
        for k=1:1:4
          R(node_eqn(element_cn(i,k),j)) = R(node_eqn(element_cn(i,k),j)) + nodal_load(4*j+k-4);
        end
     end
   end

end
```

Solve_Equations.m

```
% -------------------------------------------------------------------------
% Solve equations
% -------------------------------------------------------------------------
% K(free_dof_n,free_dof_n) : Stiffness matrix
% R(free_dof_n)          : Load vector
% U(free_dof_n)          : Displacement vector
% -------------------------------------------------------------------------

function [U]=Solve_Equations(K, R, free_dof_n)
   % solve equations
   U = K\transpose(R(1:free_dof_n));
end
```

$$\mathbf{U} = \mathbf{K}^{-1}\mathbf{R}$$

Nodal_Displacement.m

```
% -------------------------------------------------------------------------
% Obtain nodal displacement
% -------------------------------------------------------------------------
% n_displace(node_n,2) : Displacement for post process
% -------------------------------------------------------------------------

function [n_displace] = Nodal_Displacement(node_n, node_eqn, free_dof_n, U)
n_displace = zeros(node_n,2);
for i =1:node_n
   for j= 1:2
      if node_eqn(i,j) <= free_dof_n
         n_displace(i,j) = U(node_eqn(i,j));
      end
   end
end

end
```

Stress_Calculation.m

```
% ------------------------------------------------------------------------
% Calculate stress
% ------------------------------------------------------------------------
% C(3,3)     : Material law matrix
% enode(4,2) : Nodes of element
% ele_U(8)   : Element nodal displacement
% ------------------------------------------------------------------------

function [stress]=Stress_Calculation(young, poisson, row_U, enode)

r4(1) =  1.0;
r4(2) = -1.0;
r4(3) = -1.0;
r4(4) =  1.0;

s4(1) =  1.0;
s4(2) =  1.0;
s4(3) = -1.0;
s4(4) = -1.0;

% calculate a material matrix
[C]=Material_Law(young, poisson);

for i=1:1:2

   for j=1:1:2

      % set positions where stresses are out
      r = r4((i-1)*2+j); % at nodal positions of element
      s = s4((i-1)*2+j);

      % calculate B-matrix
      [buf_det, B]=Strain_Displacement(r, s, enode);

      % calculate stresses
      stress(i*2+j-2,:)=C*B*transpose(ele_U);
   end

end

end
```

$$\boldsymbol{\tau}^{(m)}\left(r,s\right)=\mathbf{C}^{(m)}\mathbf{B}^{(m)}\left(r,s\right)\mathbf{U}^{(m)}$$

Stress_Post.m

```
% --------------------------------------------------------------------------
%  Stress post processing
% --------------------------------------------------------------------------
% element_n            : # of elements
% node_x(node_n,2)       : Nodal postions
% element_cn(element_n,4) : Element connectivity
% material_young        : Young's modulus
% material_poisson       : Poisson's ratio
% displ(node_n,2)        : Nodal displacement matrix
% enode(4,2)            : Nodal positions of element
% ele_U(8)   : Element nodal displacement
% --------------------------------------------------------------------------

function Stress_Post(element_n, node_x, element_cn, material_young, material_poisson, displ, fid)

for i=1:1:element_n

   % nodal position of element
   for j=1:1:4
      for k=1:1:2
         enode(j,k) = node_x(element_cn(i,j),k);
      end
   end

   eqn(1:4) = displ(element_cn(i,1:4),1);
   eqn(5:8) = displ(element_cn(i,1:4),2);

   % calculate stress of element
   [stress]=Stress_Calculation(material_young, material_poisson, ele_U, enode);

   fprintf(fid, '\r\n Stress of Element : %g\r\n',i);
   fprintf(fid, '---------------------------------------------------------\r\n');
   fprintf(fid, 'Node      Sxx              Syy              Sxy\r\n');
   for j=1:4
      fprintf(fid,'%g    %+e     %+e     %+e\r\n',element_cn(i,j),stress(j,1),stress(j,2),stress(j,3));
   end
end

fclose(fid);

end
```

Write_Output.m

```
% ------------------------------------------------------------------------
% Write the ouput file
% ------------------------------------------------------------------------
% free_dof_n             : # of free DOFs
% element_n              : # of elements

% U(free_dof_n)           : Displacement vector
% K(free_dof_n,free_dof_n) : Stiffness matrix
% n_displace(node_n,2)     : Displacement for post process
% node_bc(node_n,6)       : Nodal displacement BC
% node_x(node_n,2)        : Nodal postions

% element_cn(element_n,4)  : Element connectivity
% material_young          : Young's modulus
% material_poisson        : Poisson's ratio
% ------------------------------------------------------------------------

function Write_Output(U, K, node_n, node_bc, n_displace, element_n, node_x, element_cn, material_young, material_poisson)
 fid = fopen('results.txt','w');
   fprintf(fid, '*****************************************************\r\n');
   fprintf(fid, '*             Plane Stress Element             *\r\n');
   fprintf(fid, '*****************************************************\r\n\r\n');

   fprintf(fid, 'EQUATION NUMBER\r\n');
   fprintf(fid, '------------------\r\n');
   fprintf(fid, 'node   DOF   eqn\r\n');
   fn = 0;
      for i=1:1:node_n
         for j=1:1:6
            if  node_bc(i,j) == 0
               fn = fn + 1;
               fprintf(fid,'%g     %g     %g \r\n',i,j,fn);
            end
         end
      end
   fprintf(fid, '\r\n');
   fprintf(fid, 'Strain Energy : ');
   fprintf(fid,'%e\r\n\r\n',0.5*U'*K*U);

   fprintf(fid, '                 Displacement\r\n');
   fprintf(fid, '---------------------------------------------------------\r\n');
   fprintf(fid, 'Node       Dx              Dy\r\n');
   for i=1:node_n
      fprintf(fid, '%g     %+e     %+e\r\n',i,n_displace(i,1),n_displace(i,2));
   end
   Stress_Post(element_n, node_x, element_cn, material_young, material_poisson, n_displace, fid);
end
```

Plot_Result.m

```
% ------------------------------------------------------------------------
% Plot initial and deformed configurations with stress
% ------------------------------------------------------------------------
% n_displace(node_n,2)    : Nodal displacement for post process
% node_x(node_n,2)       : Nodal postions
% element_n             : # of elements
% element_cn(element_n,4) : Element connectivity
% material_young         : Young's modulus
% material_poisson       : Poisson's ratio
% ------------------------------------------------------------------------

function Plot_Result(n_displace, node_x, element_n, element_cn, material_young, material_poisson)

% m : scale factor
m = 2000;

current_node(:,:)=m*n_displace(:,:)+node_x(:,:);
% Deformed configuration plot
for i=1:element_n
   for j=1:4
      for k=1:2
         enode(j,k) = current_node(element_cn(i,j),k);
      end
   end
   for j=1:4
      for k=1:2
         enode2(j,k) = node_x(element_cn(i,j),k);
      end
   end

   row_U(1:4) = n_displace(element_cn(i,1:4),1);
   row_U(5:8) = n_displace(element_cn(i,1:4),2);

   % Nodal position information
   xx = [enode(1,1) enode(2,1) enode(3,1) enode(4,1)];
   yy = [enode(1,2) enode(2,2) enode(3,2) enode(4,2)];
   zz = [1 1 1 1]';
   [stress]=Stress_Calculation(material_young, material_poisson, row_U, enode2);

   % Stress components
   for j=1:4
      sxx(j) = stress(j,1);
      syy(j) = stress(j,2);
      sxy(j) = stress(j,3);
   end
   % choose the component (sxx or syy or sxy)
   %S = [sxx(1) sxx(2) sxx(3) sxx(4)];
```

```
    S = [syy(1) syy(2) syy(3) syy(4)];
    S = [sxy(1) sxy(2) sxy(3) sxy(4)];
    patch(xx,yy,S,'Marker','o','EdgeColor','g','LineWidth',1);
    hold on;
end

% Initial configuration plot
for i=1:element_n
    for j=1:4
        for k=1:2
            enode(j,k) = node_x(element_cn(i,j),k);
        end
    end

    xx = [enode(1,1) enode(2,1) enode(3,1) enode(4,1)]';
    yy = [enode(1,2) enode(2,2) enode(3,2) enode(4,2)]';
    colormap jet;
    patch(xx,yy,zeros(size(xx)),'Marker','o','EdgeColor','black','LineWidth',3,'FaceColor','none');
    colorbar
    caxis([-3 3])

    set(gca,'xlim',[-0.5 2.5]);
    set(gca,'ylim',[-0.5 1.5]);
    pbaspect([3, 2, 1]);
end

end
```

stiffness.txt

```
 Siffness matrix of element : 1
-----------------------------------------------
+4.945055e+04 -3.021978e+04 -2.472527e+04 +5.494506e+03 +1.785714e+04 -1.373626e+03 -1.785714e+04 +1.373626e+03
-3.021978e+04 +4.945055e+04 +5.494506e+03 -2.472527e+04 +1.373626e+03 -1.785714e+04 -1.373626e+03 +1.785714e+04
-2.472527e+04 +5.494506e+03 +4.945055e+04 -3.021978e+04 -1.785714e+04 +1.373626e+03 +1.785714e+04 -1.373626e+03
+5.494506e+03 -2.472527e+04 -3.021978e+04 +4.945055e+04 -1.373626e+03 +1.785714e+04 +1.373626e+03 -1.785714e+04
+1.785714e+04 +1.373626e+03 -1.785714e+04 -1.373626e+03 +4.945055e+04 +5.494506e+03 -2.472527e+04 -3.021978e+04
-1.373626e+03 -1.785714e+04 +1.373626e+03 +1.785714e+04 +5.494506e+03 +4.945055e+04 -3.021978e+04 -2.472527e+04
-1.785714e+04 -1.373626e+03 +1.785714e+04 +1.373626e+03 -2.472527e+04 -3.021978e+04 +4.945055e+04 +5.494506e+03
+1.373626e+03 +1.785714e+04 -1.373626e+03 -1.785714e+04 -3.021978e+04 -2.472527e+04 +5.494506e+03 +4.945055e+04

 Siffness matrix of element : 2
-----------------------------------------------
+7.692308e+04 -5.769231e+04 -3.571429e+04 +1.648352e+04 +2.747253e+04 -1.098901e+04 -2.197802e+04 +5.494505e+03
-5.769231e+04 +6.730769e+04 +1.648352e+04 -2.609890e+04 -8.241758e+03 -8.241758e+03 +2.747253e+03 +1.373626e+04
-3.571429e+04 +1.648352e+04 +5.631868e+04 -3.708791e+04 -2.197802e+04 +5.494505e+03 +2.472527e+04 -8.241758e+03
+1.648352e+04 -2.609890e+04 -3.708791e+04 +4.670330e+04 +2.747253e+03 +1.373626e+04 -5.494505e+03 -1.098901e+04
+2.747253e+04 -8.241758e+03 -2.197802e+04 +2.747253e+03 +7.142857e+04 -1.648352e+04 -3.846154e+04 -1.648352e+04
-1.098901e+04 -8.241758e+03 +5.494505e+03 +1.373626e+04 -1.648352e+04 +4.395604e+04 -1.648352e+04 -1.098901e+04
-2.197802e+04 +2.747253e+03 +2.472527e+04 -5.494505e+03 -3.846154e+04 -1.648352e+04 +5.494505e+04 +3.748069e-06
+5.494505e+03 +1.373626e+04 -8.241758e+03 -1.098901e+04 -1.648352e+04 -1.098901e+04 +3.748070e-06 +2.747253e+04
```

results.txt

```
****************************************************
*              Plane Stress Element              *
****************************************************

EQUATION NUMBER
------------------
node   DOF    eqn
2      1      1
2      2      2
3      1      3
3      2      4
4      2      5
5      1      6
5      2      7
6      1      8
6      2      9

Strain Energy : 1.076086e-04

Displacement
----------------------------------------------------------
Node       Dx               Dy
1    +0.000000e+00     +0.000000e+00
2    -1.409486e-05     -8.488085e-05
3    +1.148395e-07     -1.710452e-04
4    +0.000000e+00     -3.123749e-05
5    +6.005329e-05     -8.550479e-05
6    +7.488516e-05     -1.334563e-04

Stress of Element : 1
----------------------------------------------------------
Node      Sxx              Syy              Sxy
5    +6.578693e+00     +1.911214e+00     +7.646482e-01
4    +5.569455e+00     -1.452913e+00     -2.087204e+00
1    -2.578693e+00     -3.897357e+00     -3.264648e+00
2    -1.569455e+00     -5.332300e-01     -4.127963e-01

Stress of Element : 2
----------------------------------------------------------
Node      Sxx              Syy              Sxy
6    +2.918128e+00     -1.608224e-01     -2.423409e-01
```

5	+3.239183e+00	+9.093613e-01	-8.367272e-01
2	+1.540936e+00	+3.998873e-01	-4.621629e-01
3	+1.380409e+00	-1.352045e-01	-1.649697e-01