Galerkin and Finite Element Methods

Stiffness and Mass matrices

D. Clouteau

Department of Mechanical and Civil Engineering
Ecole Centrale Paris, France

September 11, 2008

Outline

Introduction

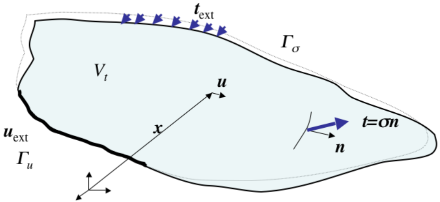
History of the FEM:

- B. Galerkin 1871-1945: St Petersburg, Civil and Mechanical Engineer. He created the "Galerkin's method" to compute approximate solution,
- W. Ritz 1878-1909: Göttingen, similar method in mathematical physics, with D. Hilbert.
- R. Courant 1888-1972: Göttingen → New-York (triangular elements 1943)
- R. Clough 1920-: Berkeley, (FEM 1959) Seismic response of a Dam, *Dynamics of Structures*, 1975,
- O. Zienkiewicz 1921-: Swansea (FEM 1965), *The Finite Element Method* 1970.
- J. Argyris 1913-2004: Stuttgart, Civil Eng. → Aerospace.
- ...

Outline

The Galerkin's Method

The basic idea



- Take a set of displacement fields $\boldsymbol{w}_n(\boldsymbol{x})$,
- Look for an approximate solution $\boldsymbol{u}_h$ as :

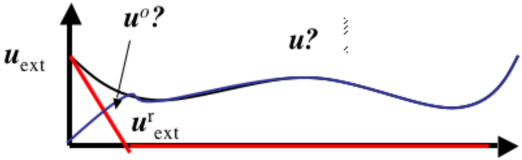
$$\boldsymbol{u}(\boldsymbol{x}, t) = \sum_{n=1}^{N} q_n(t)\boldsymbol{w}_n(\boldsymbol{x})$$

- Use these fields $\boldsymbol{w}_n(\boldsymbol{x})$ in the *Virtual Power Principle* to obtain a discrete dynamical system :

$$\mathbf{Kq} + \mathbf{C\dot{q}} + \mathbf{M\ddot{q}} = \mathbf{f}$$

to be solved for the amplitudes $q_n(t)$.

Outline



- On Γ_u : $\boldsymbol{u} = \boldsymbol{u}_{\rm ext} \neq 0$ but $\boldsymbol{w} = 0$ for the VPP.
- An auxiliary field $\boldsymbol{u}^r_{\rm ext}(\boldsymbol{x}, t)$ on V such that

$$\boldsymbol{u}^r_{\rm ext}(\boldsymbol{x}, t) = \boldsymbol{u}_{\rm ext}(\boldsymbol{x}, t) \qquad \boldsymbol{x} \in \Gamma_u$$

- A new unknown field $\boldsymbol{u}^o = \boldsymbol{u} - \boldsymbol{u}^r_{\rm ext} \in \mathbb{V}_o$ satisfying

$$\mathcal{P}^o_{\rm kin}(\boldsymbol{w}) = \mathcal{P}^o_{\rm int}(\boldsymbol{w}) + \mathcal{P}^o_{\rm ext}(\boldsymbol{w}) + \mathcal{P}^r_{\rm int}(\boldsymbol{w}) - \mathcal{P}^r_{\rm kin}(\boldsymbol{w}) \qquad \forall \boldsymbol{w} \in \mathbb{V}_o$$

Outline

- $\{\boldsymbol{w}_n\}_{n=1,N}$ a basis of $\mathbb{V}_h \in \mathbb{V}_o$ (no need to be orthogonal for $(.,.)_{\mathbb{V}_o}$ but could be for others)
- $\boldsymbol{u}_h$ the approximate solution on this basis

$$\boldsymbol{u}_h(\boldsymbol{x},t) = \sum_{n'=1}^{N} \boldsymbol{w}_{n'}(\boldsymbol{x})q_{n'}(t) = \boldsymbol{W}(\boldsymbol{x})\mathbf{q}(t),$$

- The VPP with $\boldsymbol{w} = \boldsymbol{w}_n, \forall n \leq N$:

$$\mathcal{K}(\boldsymbol{u},\boldsymbol{w}_n) + \mathcal{C}(\dot{\boldsymbol{u}},\boldsymbol{w}_n) + \mathcal{M}(\ddot{\boldsymbol{u}},\boldsymbol{w}_n) = \mathcal{P}(\boldsymbol{w}_n)$$

$$\sum_{n'=1}^{N} \mathcal{K}(\boldsymbol{w}_{n'},\boldsymbol{w}_n)q_{n'} + \mathcal{C}(\boldsymbol{w}_{n'},\boldsymbol{w}_n)\dot{q}_{n'} + \mathcal{M}(\boldsymbol{w}_{n'},\boldsymbol{w}_n)\ddot{q}_{n'} = \mathcal{P}(\boldsymbol{w}_n)$$

$$\mathbf{K}\mathbf{q} + \mathbf{C}\dot{\mathbf{q}} + \mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}$$

The Stiffness and Mass matrices
in the Galerkin method

- The symmetric positive *mass matrix* $\mathbf{M}$:

$$[\mathbf{M}]_{n'n} = \mathcal{M}(\boldsymbol{w}_{n'}, \boldsymbol{w}_n) = \int_V \rho \boldsymbol{w}_n \cdot \boldsymbol{w}_{n'} dV$$

- The *stiffness matrix* $\mathbf{K}$:

$$[\mathbf{K}]_{n'n} = \overbrace{(\mathcal{K}_e + \mathcal{K}_g - \mathcal{K}_i)}^{\mathcal{K}}(\boldsymbol{w}_{n'}, \boldsymbol{w}_n) = [\mathbf{K}_e + \mathbf{K}_g - \mathbf{K}_i]_{n'n}$$

- The symetric positive *elastic stiffness matrix* $\mathbf{K}_e$:

$$\begin{aligned}[\mathbf{K}_e]_{n'n} &= \int_V \boldsymbol{\sigma}(\boldsymbol{w}_{n'}) : \boldsymbol{\epsilon}(\boldsymbol{w}_n) dV = \int_V \mathrm{tr}(\boldsymbol{\epsilon}(\boldsymbol{w}_n)\boldsymbol{C}_e\boldsymbol{\epsilon}(\boldsymbol{w}_{n'}))dV \\ &= \int_V (\lambda \mathrm{div}(\boldsymbol{w}_n)\mathrm{div}(\boldsymbol{w}_{n'}) + 2\mu\boldsymbol{\epsilon}(\boldsymbol{w}_n) : \boldsymbol{\epsilon}(\boldsymbol{w}_{n'}))\, dV\end{aligned}$$

- The positive symmetric *damping* matrix $\mathbf{D}$ and the skew-symmetric *gyroscopic* matrix $\mathbf{C}_a$:

$$[\mathbf{C}]_{n'n} = (\mathcal{C}_a + \mathcal{D}_g)(\boldsymbol{w}_{n'}, \boldsymbol{w}_n) = [\mathbf{C}_a + \mathbf{D}]_{n'n}$$

Outline

The approximate eigenmodes

Without gyroscopic terms

- The approximate eigenvectors $\hat{\mathbf{q}}_k$ and eigenfrequencies $\tilde{\omega}_k$:

$$\mathbf{K}\mathbf{q}_k = \tilde{\omega}_k^2 \mathbf{M}\mathbf{q}_k$$

- Orthogonality :

$$\mathbf{q}_l^T \mathbf{M}\mathbf{q}_k = m_k \delta_{kl} \qquad \mathbf{q}_l^T \mathbf{K}\mathbf{q}_k = m_k \tilde{\omega}_k^2 \delta_{kl}$$

- link with the true eigenfrequencies :

$$\tilde{\omega}_k \geq \omega_k$$

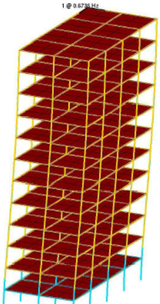
Which basis fields for the Galerkin Method?
and foreseen difficulties

- Two contradictory goals for a given accuracy :
  - Reducing the number of fields in the basis,
  - Using easy to compute displacement fields.
- Two remaining difficulties:
  - computing strains and stresses,
  - integrating over the domain.
- Two solutions:
  - Localized masses and coupling stiffnesses
  - The FE Method

Outline

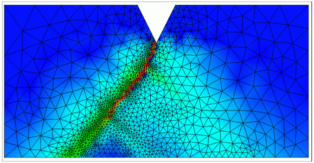
- Hypotheses:
  - Masses localized on the floors $m_i = m$,
  - Horizontal displacement of the floors: N Degrees Of Freedom $q_i(t)$,
  - One static mode due to inertial forces $\hat{\mathbf{q}}$
- Storey stiffness brought by columns: $k_i = k = n_c \frac{12EI}{h^3}$,
- The static solution : $\Delta \hat{q}_n = \hat{q}_n - \hat{q}_{n-1} = \frac{m}{k}(N+1-n)$,
- The approximate natural frequency:

$$\omega_o = \frac{\mathcal{E}_e(\hat{\mathbf{q}})}{\mathcal{E}_{\mathrm{kin}}(\hat{\mathbf{q}})} = \frac{k\|\Delta\hat{\mathbf{q}}\|^2}{m\|\hat{\mathbf{q}}\|^2}$$



Mode 1 : 0,674 Hz

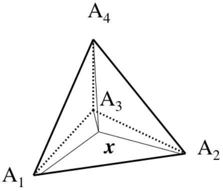
Outline



- Parametrize the geometry,
  - to perform integrations,
  - to define *Basis function*,
  - **Solution:** Divide the structure into simple cells (tetrahedrons, prisms, bricks, plates, beams...)
- Build *basis functions*
  - being continuous,
  - with simple derivatives,
  - Having given values on the boundary,
  - easy to integrate over the cells.
  - **Solution:** Polynomials on the cells
- *Interpolating basis functions* $\Leftrightarrow$ Degrees of freedom q_n being the displacement field at some given *nodes* (not mandatory),

Outline

- The *nodes* with coordinates $\boldsymbol{x}_{A=1,N_d}$
- The elements (tetrahedrons) $V_{E=1,N_e}$ with their local *nodes* $(A_i)_{i=1,n_E}$,
- Local parametrization of $\boldsymbol{x} \in V_E$:



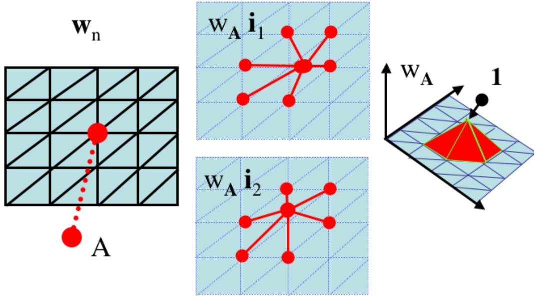
$$\boldsymbol{x} = \sum_{i=1}^{4} w_i(\boldsymbol{x})\boldsymbol{x}_{A_i} \qquad 0 \leq w_i \leq 1 \qquad \sum_{i=1}^{4} w_i = 1$$

- $w_i(\boldsymbol{x})$ are linear *interpolating* shape functions $(w_i(\boldsymbol{x}_j) = \delta_{ij})$

$$w_i(\boldsymbol{x}) = \frac{(\boldsymbol{x}_{A_k} - \boldsymbol{x}_{A_j}) \wedge (\boldsymbol{x}_{A_l} - \boldsymbol{x}_{A_j}) \cdot (\boldsymbol{x} - \boldsymbol{x}_{A_j})}{6V_E} = \frac{h_i(\boldsymbol{x})}{H_i}$$

Outline

Nodal shape functions
Parametrizing the geometry



- $V_A = \cup_{e=1}^{N_A} V_{E_A^e}$ the support the shape function,
- The scalar shape function

$$w_A(\boldsymbol{x}) = w_{E_A^e i}(\boldsymbol{x})\delta_{AA_i}, \qquad \boldsymbol{x} \in V_{E_A^e}$$

- The displacement basis function $\boldsymbol{w}_{Aa}(\boldsymbol{x}) = w_A(\boldsymbol{x})\boldsymbol{i}_a$
- Boundary condition $\boldsymbol{x}_A \notin \Gamma_u$
- $\boldsymbol{w}_{Aa}$ are continuous and piecewise differentiable.

Outline

Local matrix assembling
The tetrahedron element

- The strain tensor:

$$\boldsymbol{\epsilon}(\boldsymbol{w}_i(\boldsymbol{x})\boldsymbol{i}_a) = \frac{\boldsymbol{i}_a \otimes_s \boldsymbol{n}_i}{H_i}$$

- The stress tensor:

$$\sigma(\boldsymbol{w}_i(\boldsymbol{x})\boldsymbol{i}_a) = \frac{\lambda \boldsymbol{i}_a \cdot \boldsymbol{n}_i \boldsymbol{I}_d + 2\mu \boldsymbol{i}_a \otimes_s \boldsymbol{n}_i}{H_i}$$

- The local stiffness matrix:

$$\begin{aligned}
[K]^E_{iajb} &= \int_{V_E} \boldsymbol{\sigma}(\boldsymbol{w}_i(\boldsymbol{x})\boldsymbol{i}_a) : \boldsymbol{\epsilon}(\boldsymbol{w}_j(\boldsymbol{x})\boldsymbol{i}_b) dV \\
&= V_E \frac{\lambda \boldsymbol{i}_a \cdot \boldsymbol{n}_i \boldsymbol{i}_b \cdot \boldsymbol{n}_j + \mu(\delta_{ab} \boldsymbol{n}_i \cdot \boldsymbol{n}_j + \boldsymbol{n}_i \cdot \boldsymbol{i}_a \boldsymbol{n}_i \cdot \boldsymbol{i}_b)}{H_i H_j}
\end{aligned}$$

- The mass matrix:

$$[M]^E_{iajb} = \delta_{ab} \int_{V_E} \rho \boldsymbol{w}_i(\boldsymbol{x}) \boldsymbol{w}_j(\boldsymbol{x}) dV = \delta_{ab} \int_{V_E} \rho \frac{h_i h_j}{H_i H_j} dV$$

Outline

- Global assembling, axample of the *mass matrix*:

$$[M]_{AaBb} = \sum_{E=1}^{N_E} \sum_{i \leq n_E; j \leq n_E} \delta_{AA_i} \delta_{BA_j} [M]^E_{iajb}$$

- **Properties:**
  - All matrices but $\mathbf{C}_a$ are symmetric
  - $\mathbf{C}_a$ is skew-symmetric,
  - Huge matrices: 300K nodes leads to 10^{12} terms and 8To of memory in full storage for each,
  - All matrices are sparse: to be accounted in the storage (Profil, sparse, Element by element 50Mo).

Summary

- What's new ?
  - A general procedure leading to a discrete dynamical model:

$$(\mathbf{K}_e + \mathbf{K}_g - \mathbf{K}_i)\mathbf{q} + (\mathbf{C}_a + \mathbf{D})\dot{\mathbf{q}} + \mathbf{M}\ddot{\mathbf{q}} = \mathbf{f}$$

  - an efficient and versatile procedure to assemble this system: **Finite Element Method**
- What's left ?
  - How to solve the discrete model ?
  - Which level of refinement ?
  - What about beams and plates ?