AMS527: Numerical Analysis II

A Brief Overview of Finite Element Methods (II): Computer implementation of FEM

Xiangmin Jiao

SUNY Stony Brook

April 29, 2009

Stiffness Matrix

- To implement finite-element method for Poisson's equation, key is to obtain linear system

$$\boldsymbol{K}\boldsymbol{u} = \boldsymbol{f}$$

where

$$K_{ij} = \iint_S \left(\frac{\partial T_i}{\partial x}\frac{\partial T_j}{\partial x} + \frac{\partial T_i}{\partial y}\frac{\partial T_j}{\partial y} \right) dx\, dy$$

$$f_i = \iint_S f T_i \, dx\, dy$$

- On triangles, each T_j is pyramid over jth vertex v_j, and

$$K_{ij} = \sum_{\{e \mid v_i \in e \wedge v_j \in e\}} \iint_e \left(\frac{\partial T_i}{\partial x}\frac{\partial T_j}{\partial x} + \frac{\partial T_i}{\partial y}\frac{\partial T_j}{\partial y} \right) dx\, dy$$

$$f_i = \sum_{\{e \mid v_i \in e \wedge v_j \in e\}} \iint_e f T_i \, dx\, dy$$

Element Stiffness Matrices

- Within each triangle, let $U = a + bx + cy$, compute

$$(k_e)_{IJ} = \iint_e \left(\frac{\partial N_I}{\partial x}\frac{\partial N_J}{\partial x} + \frac{\partial N_I}{\partial y}\frac{\partial N_J}{\partial y} \right) dx\,dy, \qquad I, J = 1, 2, 3$$

  where N_I is shape function w.r.t. Ith vertex.
- Matrix $k_e = [(k_e)_{IJ}]$, $I, J = 1, 2, 3$ is *element stiffness matrix*
- For Laplacian equation, it can be shown that

$$k_e = \begin{bmatrix} c_2 + c_3 & -c_3 & -c_2 \\ -c_3 & c_1 + c_3 & -c_1 \\ -c_2 & -c_1 & c_1 + c_2 \end{bmatrix}, \quad \text{where } c_i = \frac{1}{2\tan\theta_i}$$

- Question: Is k_e symmetric and positive definite?

Mass Matrix and Element Mass Matrices

- Besides stiffness matrix, mass matrix $[M_{ij}]$ is very common in finite element methods

$$
\begin{aligned}
M_{ij} &= \iint_S T_i T_j \, dx \, dy \\
&= \sum_{\{e | v_i \in e \wedge v_j \in e\}} \iint_e T_i T_j \, dx \, dy
\end{aligned}
$$

- Its computation involves element mass matrix m_e with

$$
(m_e)_{IJ} = \iint_e N_I N_J \, dx \, dy, \qquad I, J = 1, 2, 3
$$

General Procedure for Elemental Computation

- In general, $\partial N_I / \partial x$ and $\partial N_I / \partial y$ is obtained from chain rule

$$\underbrace{\begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}}_{\boldsymbol{J}^T} \begin{bmatrix} \frac{\partial N_I}{\partial x} \\ \frac{\partial N_I}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{\partial N_I}{\partial \xi} \\ \frac{\partial N_I}{\partial \eta} \end{bmatrix}$$

  where $\boldsymbol{J} = \begin{bmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{bmatrix}$. ξ and η are called *natural coordinates*

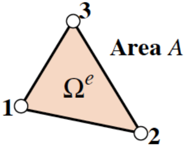
- Requires computing Jacobian matrix and derivatives of shape functions

Example Quadrature Rules Over Triangle

- Requires quadrature rules over elements

Centroid rule: 1 point, degree 1

$$\frac{1}{A}\int_{\Omega^e} F(\zeta_1,\zeta_2,\zeta_3)\,d\Omega \approx F(\tfrac{1}{3},\tfrac{1}{3},\tfrac{1}{3})$$



Three-interior-point rule: 3 points, degree 2

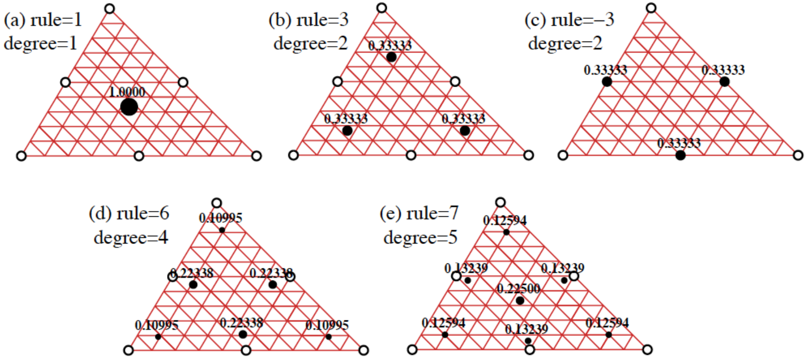
$$\frac{1}{A}\int_{\Omega^e} F(\zeta_1,\zeta_2,\zeta_3)\,d\Omega \approx \tfrac{1}{3}F(\tfrac{2}{3},\tfrac{1}{6},\tfrac{1}{6})+\tfrac{1}{3}F(\tfrac{1}{6},\tfrac{2}{3},\tfrac{1}{6})+\tfrac{1}{3}F(\tfrac{1}{6},\tfrac{1}{6},\tfrac{2}{3})$$

Midpoint rule: 3 points, degree 2

$$\frac{1}{A}\int_{\Omega^e} F(\zeta_1,\zeta_2,\zeta_3)\,d\Omega \approx \tfrac{1}{3}F(\tfrac{1}{2},\tfrac{1}{2},0)+\tfrac{1}{3}F(0,\tfrac{1}{2},\tfrac{1}{2})+\tfrac{1}{3}F(\tfrac{1}{2},0,\tfrac{1}{2})$$

Source: http://www.colorado.edu/engineering/cas/courses.d/IFEM.d

Example Quadrature Rules Over Triangle



http://www.colorado.edu/engineering/cas/courses.d/IFEM.d

Assembling Element Matrices

- For each triangle, k_e is 3×3 matrix, each of whose rows (columns) correspond to a vertex
- Each vertex of triangle has global vertex ID. Example:
  - triangle 1: nodes 3, 1, 2
  - triangle 2: nodes 3, 4, 2
  - ...
  - triangle m: nodes 5, 4, n

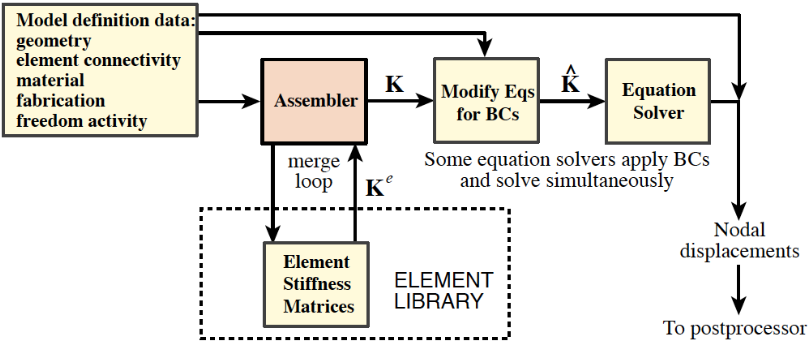
Assembling Element Matrices

- Map k_e for element e to an $n \times n$ matrix K_e, based on local to global mapping of vertex IDs, e.g.,

$$K_1 = \begin{bmatrix} (k_e)_{22} & (k_e)_{23} & (k_e)_{21} & & & \\ (k_e)_{32} & (k_e)_{33} & (k_e)_{31} & & & \\ (k_e)_{12} & (k_e)_{13} & (k_e)_{11} & & & \\ & & & 0 & & \\ & & & & \ddots & \\ & & & & & 0 \end{bmatrix}$$

- Then, $\boldsymbol{K} = \boldsymbol{K}_1 + \boldsymbol{K}_2 + \cdots + \boldsymbol{K}_m$
- Without boundary condition, $\boldsymbol{K}$ is singular! After applying boundary condition, $\boldsymbol{K}$ is SPD
- In iterative linear solvers involving only matrix-vector multiplications, $\boldsymbol{K}\boldsymbol{x} = \boldsymbol{K}_1\boldsymbol{x} + \boldsymbol{K}_2\boldsymbol{x} + \cdots + \boldsymbol{K}_m\boldsymbol{x}$, so $\boldsymbol{K}$ need not be assembled explicitly, leading to matrix-free linear solvers

Role of Assembler



Summary of Key Aspects for Implementing FEM

- Input of finite element methods
  - Vertex coordinates, used for computing Jacobian matrix
  - Element connectivity, used for local to global mapping in assembling matrix
  - Boundary conditions
- Elemental computation
  - Shape functions, first and second derivatives
  - Numerical quadrature rules and Gauss points
- Assemble element matrices into stiffness matrix
- Solution of sparse linear system

Accuracy and Convergence of Finite Elements

- When using degree p piecewise linear polynomials, basic theory of FEM can be stated as:
  - The finite element method converges if $p \geq 1$, and its error is $\mathcal{O}(h^{2p})$, where h is largest edge length.
  - The answers are exact for solutions of degree p (checked by *patch test*)
- In addition, finite-element method gives optimal solution $\boldsymbol{U}$ that is closest to exact solution $\boldsymbol{u}$ in that it minimizes the energy

$$E = (\boldsymbol{U} - \boldsymbol{u})^T \boldsymbol{K}(\boldsymbol{U} - \boldsymbol{u})$$

within all feature solutions in function space of trial functions