2D Finite Element Formulation

The basic steps involved in solving a boundary value problem by finite element method are as follows:

1) Formulation of a variational statement with an appropriate space of admissible functions identified.
2) Construction of a finite element mesh and piecewise-polynomial basis functions defined on the mesh.
3) Construction of an approximation of the variational boundary value problem on a finite element subspace H^h. This generates a system of algebraic equations (or ordinary differential equations).
4) Solution of a system of equations.

Example 1.

$$
\begin{aligned}
-\nabla \cdot [k\nabla u] + bu &= f(x,y) && x, y \in \Omega, \\
u(s) = \hat{u}(s) & && s \in \partial\Omega_1, \\
-k(s)\frac{\partial u(s)}{\partial n} = p(s)[u(s) - \hat{u}(s)] &= \hat{\sigma}(s) && s \in \partial\Omega_2,
\end{aligned} \tag{1}
$$

where ∇ is the gradient operator, $\nabla\cdot$ is the divergence operator and $\triangle = \nabla^2$ is the Laplace operator.

Variational Statement

To construct the variational statement of the boundary value problem (1), we define the residual function

$$r = -\nabla \cdot [k\nabla u] + bu - f.$$

To test the residual over an arbitrary subregion, we multiply r by a sufficiently smooth test function v, integrate over Ω and set the resulting overall weighted residual to zero, we thus have

$$\int_\Omega [-\nabla \cdot (k\nabla u) + bu - f]v \, d\Omega \; = \; 0. \tag{2}$$

Then, as is typical in finite element work, we proceed to reduce the 2nd order terms to the 1st order by integration by parts. Using the product rule for differentiation

$$\begin{aligned} \nabla \cdot (vk\nabla u) &= k\nabla u \cdot \nabla v + v\nabla \cdot (k\nabla u) \\ \Rightarrow \quad v\nabla \cdot (k\nabla u) &= \nabla \cdot (vk\nabla u) - k\nabla u \cdot \nabla v, \end{aligned} \tag{3}$$

we have from (2)

$$\int_{\Omega}[k\nabla u\cdot\nabla v-\nabla\cdot(vk\nabla u)+buv-fv]d\Omega=0. \tag{4}$$

From the divergence theorem

$$\int_{\Omega}\nabla\cdot(vk\nabla u)\ d\Omega\ =\ \int_{\partial\Omega}vk\nabla u\cdot n\ ds\ =\ \int_{\partial\Omega}vk\frac{\partial u}{\partial n}\ ds, \tag{5}$$

equation (4) becomes

$$\int_{\Omega}[k\nabla u\cdot\nabla v+buv-fv]\ d\Omega-\int_{\partial\Omega}k\frac{\partial u}{\partial n}v\ ds\ =\ 0. \tag{6}$$

Choosing v such that $v = 0$ on $\partial\Omega_1$ and using the boundary condition $(1)_3$, we obtain

$$\int_\Omega [k\nabla u \cdot \nabla v + buv - fv] \, d\Omega + \int_{\partial\Omega_2} puv \, ds - \int_{\partial\Omega_2} p\hat{u}v \, ds \; = \; 0. \quad (7)$$

To specify the appropriate class of admissible functions for problem (7),

- we examine the integrals in (7) and observe that the area integrals are well defined whenever u and v and their 1st order partial derivatives are smooth enough to be square-integrable over Ω.
- Thus, we need to choose u and v from $H^1(\Omega)$.

Hence, our variational boundary value problem can now be stated concisely in the following form:
Find $u \in H^1(\Omega)$ such that $u = \hat{u}$ on $\partial\Omega_1$ and

$$a(u, v) = L(v) \quad \forall v \in H^1(\Omega), \tag{8}$$

where $\quad H_0^1 \; = \; \{v : v \in H^1 \text{ and } v = 0 \text{ on } \partial\Omega_1\},$

$a(u, v) = \int_\Omega (k\nabla u \cdot \nabla v + buv)\, d\Omega + \int_{\partial\Omega_2} puv\, ds$ is a bilinear form

$L(v) = \int_{\partial\Omega_2} p\hat{u}v\, ds + \int_\Omega fv\, d\Omega$ is a linear form of v.

The Galerkin Approximation

A Galerkin approximation of (8) is obtained by posing the variational problem on a finite-dimensional subspace H^h of the space of admissible functions. Specifically, we

$$\begin{aligned} &\text{seek} \quad u_h \in H_h^1 \quad \text{such that} \;\; u_h = \hat{u} \text{ on } \;\partial\Omega_1 \;\; \text{and} \\ &a(u_h, v_h) = L(v_h) \qquad\qquad \forall v_h \in H_{0h}^1 \end{aligned} \tag{9}$$

Let $\{\phi_i(x, y)\}_{i=1}^N$ be the basis functions of H_h^1, then

$$u_h = \sum_{j=1}^N \alpha_j \phi_j(x, y), \quad v_h = \sum_{i=1}^N \beta_i \phi_i(x, y). \tag{10}$$

Substituting (10) into (9) yields

$$\begin{aligned} \textstyle\sum_{i=1}^{N} a(u_h, \phi_i)\beta_i &= \textstyle\sum_{i=1}^{N} L(\phi_i)\beta_i \qquad \forall \beta_i \\ \Rightarrow \quad a(u_h, \phi_i) &= L(\phi_i), \quad (i = 1, 2, \ldots, N) \end{aligned} \tag{11}$$

Substituting (10) into (11) yields

$$\begin{aligned} \textstyle\sum_{i=1}^{N} a(\phi_i, \phi_j)\alpha_j &= L(\phi_i), \quad (i = 1, 2, \ldots, N) \\ \Rightarrow \quad \mathbf{A}\alpha &= \mathbf{F}, \end{aligned} \tag{12}$$

where $\mathbf{A} = (a_{ij})$ is an $N \times N$ matrix with $a_{ij} = a(\phi_i, \phi_j)$,
$\mathbf{F} = (F_i) \in \mathcal{R}^N$ with $F_i = L(\phi_i)$ and
$\alpha = (\alpha_i) \in \mathcal{R}^N$.

Therefore, the Galerkin approximation u_h of the solution u is of the form

$$u_h = \sum_{j=1}^{N} \alpha_j \phi_j(x, y), \tag{13}$$

where $\alpha \in \mathcal{R}^n$ is determined by (12) and

$\{\phi_j(x, y)\}_{j=1}^{N}$ are basis functions of H_h^1.

The finite element method provides a general and systematic technique for constructing the basis functions ϕ_i.

The Finite element Interpolation

Consider an open bounded domain Ω in $\mathcal{R}^N$ with boundary $\partial\Omega$.

Let $u \in C^m(\bar{\Omega})$ where $\bar{\Omega}$ is the closure of Ω, then the construction of a finite element interpolation of $u(\phi_i)$ can be accomplished by the following steps.

1) Partitioning of $\bar{\Omega}$

We replace $\bar{\Omega}$ by a collection $\bar{\Omega}_h$ of simple domain (element) $\bar{\Omega}_e$ such that

a) $\bar{\Omega}_h = \cup_{e=1}^{E} \bar{\Omega}_e$

b) $\bar{\Omega}_e \cap \bar{\Omega}_f = \phi$ for distinct $\bar{\Omega}_e$ and $\bar{\Omega}_f \in \bar{\Omega}_h$

c) every $\bar{\Omega}_e$ is closed and consists of a non-empty interior Ω_e and a boundary $\partial\Omega_e$.

2) Local Interpolation Over $\bar{\Omega}_e$-Local Basis ϕ_i^e

Over each $\bar{\Omega}_e$,

- we choose N_e nodes where the values of u and u_i^e are to be used as basic unknowns.
- Then we construct local interpolation function $\{\phi_i^e(x,y)\}_{i=1}^{N_e}$ such that the restriction of u_h to $\bar{\Omega}_e$ is

$$u_h^e(x,y) = \sum_{i=1}^{N_e} u_i^e \phi_i^e(x,y).$$

The form of $\phi_i^e(x,y)$ depends on type of elements.

3) Assembly of Global Basis Functions ϕ_i

The global basis functions ϕ_i can be generated by patching together those local shape functions ϕ_i^e defined over $\bar{\Omega}_e$ which contain the node i.

Suppose node i in the finite element mesh is shared by M elements. Then the local shape functions for point i corresponding to each of these elements are combined to form the global ϕ_i which satisfies

- the proper inter-element continuity
- $\phi_i(x_j, y_j) = \delta_{ij}$
- $\phi_i(x, y)$ is non-zero only over the particular patch of the M elements meeting at node i.

Thus, we can generate N linearly independent functions $\{\phi_i(x, y)\}_{i=1}^{N}$ which form a basis of an $N-\texttt{dimension}$ function space.

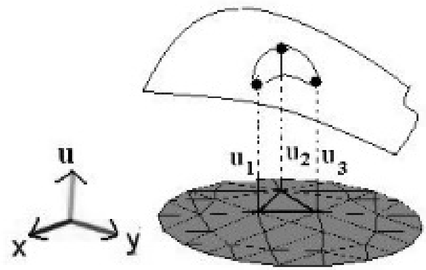
Triangular Elements

Linear 3-point triangular elements

Approximate $u(x)$ over the element Ω_e by

$$u_h^e(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y, \quad \forall (x,y) \in \Omega_e \tag{14}$$

which determines a plane surface. Thus the use of linear interpolation on a triangular element will result in the approximation of a given smooth surface $v(x,y)$ by a plane as shown.



By evaluating (14) at each node, we have

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix}$$

or $$u^e = p(x_i)$$

$\Rightarrow$ $$\alpha = p^{-1}(x_i)u^e$$

Therefore $u_h^e(x,y) = [1,\ x,\ y]p^{-1}(x_i)u^e$ which can be rearranged to yield

$$u_h^e(x,y) = u_1\phi_1^e + u_2\phi_2^e + u_3\phi_3^e \tag{15}$$

with element shape functions being

$$
\begin{cases}
\phi_1^e(x,y) = \frac{1}{2A_e}[(x_2y_3 - x_3y_2) + (y_2 - y_3)x + (x_3 - x_2)y] \\
\phi_2^e(x,y) = \frac{1}{2A_e}[(x_3y_1 - x_1y_3) + (y_3 - y_1)x + (x_1 - x_3)y] \\
\phi_3^e(x,y) = \frac{1}{2A_e}[(x_1y_2 - x_2y_1) + (y_1 - y_2)x + (x_2 - x_1)y]
\end{cases}
\tag{16}
$$

and $A_e = \frac{1}{2}\begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}$ is area of an element.

(2) Higher Order Triangular Element

Let us first display the terms appearing in polynomials of various degrees in two variables in the form as shown in Figure 2.

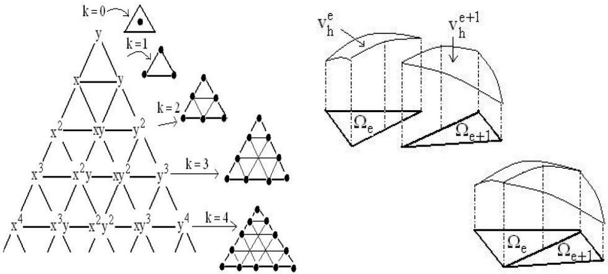


Figure: Pascal's triangle

The above triangular array is called Pascal's triangle.

Rectangular Elements

By taking the product of a set of polynomials in x with a set of polynomials in y, shape functions for a variety of rectangular elements can be obtained.

(1) Bilinear polynomials The product of $(1, x)$ and $(1, y)$ produces a matrix

$$\begin{bmatrix} 1 \\ x \end{bmatrix} \begin{bmatrix} 1 & y \end{bmatrix} = \begin{bmatrix} 1 & y \\ x & xy \end{bmatrix}. \tag{17}$$

A bilinear local interpolant can then be obtained by forming a linear combination of all the four terms in the matrix, i.e

$$v_h^e(x, y) = a_1 + a_2 x + a_2 y + a_4 xy. \tag{18}$$

(2) Higher Order Rectangular Elements

By considering tensor products of polynomials of higher degree, element shape functions can be constructed which contain polynomials of any desired degree and which lead to basis functions that are continuous throughout Ω_h.

eg. For a biquadratic local interpolation, we firstly find the matrix from

$$\begin{bmatrix} 1 \\ x \\ x^2 \end{bmatrix} \begin{bmatrix} 1 & y & y^2 \end{bmatrix} = \begin{bmatrix} 1 & y & y^2 \\ x & xy & xy^2 \\ x^2 & x^2y & x^2y \end{bmatrix}$$

Then the biquadratic local interpolant v_h^e is obtained by forming a linear combination of all the nine terms in the matrix.

To completely determine the interpolant, construct a rectangular element with nine nodes as shown.

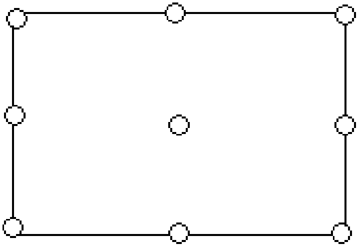


Figure: A rectangular element with nine nodes

Interpolation Error

Let g be a smooth function given,

g_n be the finite element representation which contains a complete polynomial of degree k.

If all partial derivatives of g of order $k+1$ are bounded in the domain Ω_h, then the interpolation error satisfies

$$\|g - g_h\|_{\infty,\Omega_e} = \max_{(x,y)\in\Omega_e} |g(x,y) - g_h(x,y)| \leq C h_e^{k+1}$$

where C is a positive constant and h_e is the *diameter* of Ω_e ; that is, h_e is the largest distance between any two points in Ω_e. Similarly

$$\left\|\frac{\partial g}{\partial x} - \frac{\partial g_h}{\partial x}\right\|_{\infty,\Omega_e} \leq C_1 h_e^k, \quad \left\|\frac{\partial g}{\partial y} - \frac{\partial g_h}{\partial y}\right\|_{\infty,\Omega_e} \leq C_2 h_e^k.$$

Finite Element Approximation

Return to the problem, as we choose $\phi_i(x_j, y_j) = \delta_{ij}$, our finite element approximation of u is

$$u_h(x, y) = \sum_{j=1}^{N} u_j \phi_j(x, y).$$

Thus, our problem now is:

Find $\mathbf{u} \in \mathcal{R}^N$ such that $u_i = \hat{u}$ on $\partial\Omega_1$ and

$$\mathbf{Au} = \mathbf{F}$$

where
$\mathbf{A} = (a_{ij})$ with $a_{ij} = a(\phi_i, \phi_j) = \int_\Omega (k\nabla\phi_i \cdot \nabla\phi_j + b\phi_i\phi_j) + \int_{\partial\Omega_2} p\phi_i\phi_j \, ds$
$\mathbf{F} = (F_i)$ with $F_i = L(\phi_i) = \int_\Omega f\phi_i \, d\Omega + \int_{\partial\Omega_2} p\hat{u}\phi_i \, ds$.

As $\phi_i(x,y)$ are defined piecewisely over each element Ω_e, we have

$$a_{ij} = \sum_{e=1}^{E} \int_{\Omega_e} (k\nabla\phi_i \cdot \nabla\phi_j + b\phi_i\phi_j)\, d\Omega + \sum_{e=1}^{E} \int_{\partial\Omega_{2e}} p\phi_i\phi_j\, ds$$

$$F_i = \sum_{e=1}^{E} \left\{ \int_{\Omega_e} f\phi_i d\Omega + \int_{\partial\Omega_{2e}} p\hat{u}\phi_i\, ds \right\}.$$

To assemble **A**, loop over all elements to calculate a^e and successively add in the contributions from each a^e as follows :

Set $\mathbf{A}(i,j) = 0, \quad b(i) = 0, \quad i,j = 1,2,...N$

For $e = 1,2...E$

calculate a^e

Set $\quad \mathbf{A}_{g(e,\alpha)g(e,\beta)} = \mathbf{A}_{g(e,\alpha)g(e,\beta)} + a^e_{\alpha\beta}$

$$\mathbf{F}_{g(e,\alpha)} = \mathbf{F}_{g(e,\alpha)} + F^e_\alpha \quad \alpha,\beta = 1,2,...,N_e.$$

where $g(e,k)$ is the global node number of the k^{th} node of element e.

Boundary Condition

Suppose at point $\ell, u_\ell = \hat{u}$ and the assembled system is

$$\begin{bmatrix} a_{11} & \cdots & a_{1\ell} & \cdots & a_{1N} \\ \vdots & & \vdots & & \vdots \\ a_{\ell 1} & \cdots & a_{\ell\ell} & \cdots & a_{\ell N} \\ \vdots & & \vdots & & \vdots \\ a_{N1} & \cdots & a_{N\ell} & \cdots & a_{NN} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_\ell \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 \\ \vdots \\ f_\ell \\ \vdots \\ f_N \end{bmatrix}$$

We impose the boundary condition $u_\ell = \hat{u}$ by performing the following steps:

1) Move the known values to the right hand side

$$\left[\begin{array}{cc|c|cc} a_11 & \cdots & 0 & \cdots & a_{1N} \\ \vdots & & \vdots & & \vdots \\ a_{\ell 1} & \cdots & 0 & \cdots & a_{\ell N} \\ \vdots & & \vdots & & \vdots \\ a_{N1} & \cdots & 0 & \cdots & a_{NN} \end{array}\right] \begin{bmatrix} u_1 \\ \vdots \\ u_\ell \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 & - & a_{1\ell}\hat{u} \\ & & \vdots \\ f_\ell & - & a_{\ell\ell}\hat{u} \\ & & \vdots \\ f_N & - & a_{N\ell}\hat{u} \end{bmatrix}$$

2) Impose the restriction $\phi_\ell = 0$ on the system.
Noting that $a_{\ell j} = a(\phi_\ell, \phi_j) = 0,\ f_\ell = L(\phi_\ell) = 0$, we have

$$\begin{bmatrix} a_11 & \cdots & 0 & \cdots & a_{1N} \\ & & \vdots & & \\ \hline 0 & \cdots & \vdots & \cdots & 0 \\ \hline & & \vdots & & \\ a_{N1} & & 0 & & a_{NN} \end{bmatrix} \begin{bmatrix} u_1 \\ \vdots \\ u_\ell \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 & - & a_{1\ell}\hat{u} \\ f_2 & - & a_{2\ell}\hat{u} \\ & & \vdots \\ & & 0 \\ & & \vdots \\ f_N & - & a_{N\ell}\hat{u} \end{bmatrix}$$

This set of equations is rank deficient and need to be modified by one of the following methods.

- Combine with the Dirichlet Condition $u_\ell = \hat{u}_\ell$ to yield

$$\left[\begin{array}{cc|c|cc} a_{11} & \cdots & 0 & \cdots & a_{1N} \\ \vdots & & \vdots & & \\ \hline 0 & \vdots & 1 & \vdots & 0 \\ \hline \vdots & & \vdots & & \\ a_{N1} & \cdots & 0 & \cdots & a_{NN} \end{array}\right] \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_\ell \\ \vdots \\ u_N \end{bmatrix} = \begin{bmatrix} f_1 & - & a_{1\ell}\hat{u} \\ & & \vdots \\ & & \hat{u} \\ & & \vdots \\ f_N & - & a_{N\ell}\hat{u} \end{bmatrix}$$

- Delete row ℓ and column ℓ to form an $(N-1) \times (N-1)$ system.

Example 2

Consider

$$\begin{cases} -\Delta(x,y) = f(x,y) & in \;\; \Omega \\ u = 0 & on \;\; \Gamma_{41} \\ \frac{\partial u}{\partial n} = 0 & on \;\; \Gamma_{12}, \Gamma_{25}, \Gamma_{67}, and \Gamma_{74} \\ \frac{\partial u}{\partial n} + \beta u = \gamma & on \;\; \Gamma_{56} \end{cases}$$

In this case $\partial\Omega_1 = \Gamma_{41}$

$$\partial\Omega_2 = \Gamma_{12} \cup \Gamma_{25} \cup \Gamma_{67} \cup \Gamma_{74} \cup \Gamma_{56}$$

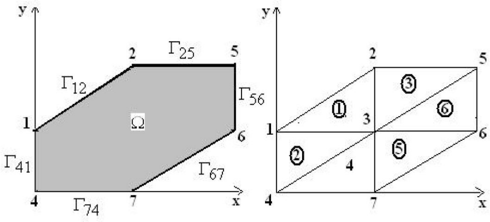


Figure:

Our analysis of this problem proceeds as follows:

- Partition Ω into six triangular elements.
- Compute the element matrices a^e and f^e $(e = 1, 2, ..., 6)$

$$a^e = \begin{bmatrix} a^e_{11} & a^e_{12} & a^e_{13} \\ a^e_{21} & a^e_{22} & a^e_{23} \\ a^e_{31} & a^e_{32} & a^e_{33} \end{bmatrix}, f^e = \begin{bmatrix} f^e_1 \\ f^e_2 \\ f^e_3 \end{bmatrix}$$

- Assemble the element matrices to form the global matrix using the following opology:

ele	node 1	2	3
1	1	2	3
2	1	3	4
3	2	5	3
4	3	4	7
5	3	6	7
6	3	5	6

Hence, we have global matrix and vector

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} & K_{14} & 0 & 0 & 0 \\ K_{21} & K_{22} & K_{23} & 0 & K_{25} & 0 & 0 \\ K_{31} & K_{32} & K_{33} & K_{34} & K_{35} & K_{36} & K_{37} \\ K_{41} & 0 & K_{43} & K_{44} & 0 & 0 & K_{47} \\ 0 & K_{52} & K_{53} & 0 & K_{55}+K_b & K_{56} & 0 \\ 0 & 0 & K_{63} & 0 & K_{65} & K_{66}+K_b & K_{67} \\ 0 & 0 & K_{73} & K_{74} & 0 & K_{76} & K_{77} \end{bmatrix}, \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5+F_b \\ F_6+F_b \\ F_7 \end{bmatrix}$$

- Impose the essential boundary condition.

Question 1

Consider a rectangular element with four nodes, one at each corner. In view of our criteria for acceptable finite element basis functions, why is the following choice of a local test function representation unacceptable ?

$$v_h^e(x, y) = a_1 + a_2 x + a_3 y + a_4 x^2$$

Here a_1, a_2, a_3 *and* a_4 *are constants.*

Question 2

Suppose that Ω_h is a square consisting of eight triangular elements of equal size. Describe by means of sketches, the global basis functions ϕ_i, $i = 1, 2, ...9$ generated by piecewise-linear shape functions on each element.

Question 3 In Example 2,

a) Suppose that all the elements in the mesh shown are equal isoscale triangles, the two equal sides being of length h. Derive the element stiffness matrix a^e and f^e for $f(x, y) = 1$.

b) Suppose the coordinates of the nodes in the mesh is as shown. Use the result in a) to calculate the element stiffness matrices and load vectors for all six elements.

c) Construct the global matrices and load vector.

d) Impose the boundary condition to obtain the final system of equations.