The Finite Element Method

Section 6

Dr. Matthew Santer

Department of Aeronautics, I.C.L.

m.santer@imperial.ac.uk

Section Objectives

In the previously-illustrated finite element solutions, we have approximated the variables and weight functions by means of shape (or basis) functions N. We now consider the form and requirements of these shape functions in greater detail.

In this section we will:

- consider general shape function properties that must be satisfied
- introduce the bilinear quadrilateral element and show how this may be degenerated to form a compatible 3-noded triangular element
- formally define an isoparametric element
- introduce higher-order elements in 1 and 2-dimensions

Shape Function Requirements

We require that element shape function choice means that as the finite element mesh is refined, the approximate Galerkin solution converges to the exact solution. To achieve this convergence, the basic requirements of the shape functions are that they are:

- smooth (at least C^1) on each element interior Ω^e
- continuous across each element boundary Γ^e
- complete

The first two requirements ensure the first derivatives of the shape functions have, at worst, a finite jump across Γ^e. This ensures that all necessary integrals are well defined.

'Completeness' requires that the element interpolation functions are capable of representing an arbitrary linear polynomial when the nodal degrees of freedom are assigned values in accordance with it.

Completeness: Linear Heat Conduction

Completeness can be illustrated for the 2-D linear heat conduction case as follows.

$$u^h = \sum_{a=1}^{n_{en}} N_a d_a^e \qquad d_a^e = u^h(\mathbf{x}_a^e)$$

The shape functions are complete if

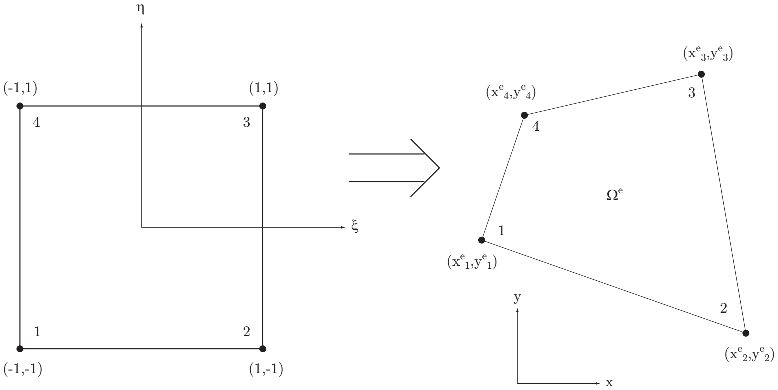
$$d_a^e = c_0 + c_1 x_a^e + c_2 y_a^e$$

implies

$$u^h(\mathbf{x}) = c_0 + c_1 x + c_2 y$$

Bilinear Quadrilateral Element

We first illustrate shape functions for the bilinear four-node quadrilateral element. The element is a square in element (ξ, η) coordinates and is related to the element in world coordinates by an affine mapping.



Bilinear Quadrilateral Element: local-global mapping

The element's natural (local) coordinates are

$$\xi = \left\{ \begin{array}{c} \xi \\ \eta \end{array} \right\}$$

which are mapped to world coordinates

$$\mathbf{x} = \left\{ \begin{array}{c} x \\ y \end{array} \right\}$$

This mapping is

$$x(\xi,\eta) = \sum_{a=1}^{4} N_a(\xi,\eta) x_a^e \qquad y(\xi,\eta) = \sum_{a=1}^{4} N_a(\xi,\eta) y_a^e$$

which is more concisely expressed as

$$\mathbf{x}(\xi) = \sum_{a=1}^{4} N_a(\xi) \mathbf{x}_a^e$$

Bilinear Quadrilateral Element: shape function generation

Assume the following bilinear expansions

$$
\begin{aligned}
x(\xi,\eta) &= \alpha_0 + \alpha_1\xi + \alpha_2\eta + \alpha_3\xi\eta \\
y(\xi,\eta) &= \beta_0 + \beta_1\xi + \beta_2\eta + \beta_3\xi\eta
\end{aligned}
$$

We clearly require

$$
\begin{aligned}
x_a(\xi_a,\eta_a) &= x_a^e \\
y_a(\xi_a,\eta_a) &= y_a^e
\end{aligned}
$$

where

$$
\begin{array}{c|cc}
a & \xi_a & \eta_a \\
\hline
1 & -1 & -1 \\
2 & 1 & -1 \\
3 & 1 & 1 \\
4 & -1 & 1
\end{array}
\implies N_a(\xi_b) = \delta_{ab}
$$

Bilinear Quadrilateral Element: shape functions

We can therefore express the nodal coordinates in terms of the coefficients α_i (and similarly for $\mathbf{y}^e$ in terms of the coefficients β_i) as

$$\left\{ \begin{array}{c} x_1^e \\ x_2^e \\ x_3^e \\ x_4^e \end{array} \right\} = \left[\begin{array}{cccc} 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{array} \right] \left\{ \begin{array}{c} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{array} \right\}$$

From this we can obtain expressions for α_i and with reference to the table of values of ξ_a and η_a above, we obtain

$$\mathbf{x}(\xi, \eta) = \sum_{a=1}^{4} N_a(\xi, \eta) \mathbf{x}_a^e$$

where

$$N_a(\xi, \eta) = \frac{1}{4}(1 + \xi_a \xi)(1 + \eta_a \eta) = \frac{1}{4}(1 + \xi_a \xi + \eta_a \eta + (\xi_a \eta_a) \xi \eta)$$

Isoparametric Elements

Assume that the element functions are given by similar expansions in terms of the shape functions, i.e.

$$\text{Linear Heat Conduction:} \quad u^h(\xi,\eta) = \sum_{a=1}^{4} N_a(\xi,\eta) d_a^e$$

It can be shown that the smoothness, continuity, and completeness requirements of the shape functions are met.

This case indicates an 'isoparametric' element: the same shape function is used for both the mappings and the element interpolation functions.

Isoparametric Elements

More formally we denote the element parent domain in ξ-space by $\square$. We then let the mapping $x : \square \rightarrow \Omega^e$ be of the form

$$\mathbf{x}(\xi) = \sum_{a=1}^{n_{en}} N_a(\xi)\mathbf{x}_a^e$$

If the element interpolation function $\mathbf{u}^h$ can be written as

$$\mathbf{u}^h(\xi) = \sum_{a=1}^{n_{en}} N_a(\xi)\mathbf{d}_a^e$$

then the element is isoparametric, as we have already seen for the bilinear quadrilateral.

Mapping Definitions

A mapping is defined as *one-to-one* if for each pair $\xi_1, \xi_2 \in \Box$ where $\xi_1 \neq \xi_2$, then $\mathbf{x}(\xi_1) \neq \mathbf{x}(\xi_2)$. We also require that the mapping is differentiable. This means that there exists a *Jacobean Determinant* defined as

$$j = det\left(\frac{\partial \mathbf{x}}{\partial \xi}\right)$$

In two dimensions, for example, the Jacobean is written more fully as

$$j = \begin{vmatrix} x_{,\xi} & x_{,\eta} \\ y_{,\xi} & y_{,\eta} \end{vmatrix}$$

If the mapping of $\mathbf{x}$ onto Ω^e is one-to-one then

$$j(\xi) > 0 \ \forall \ \xi \in \Box$$

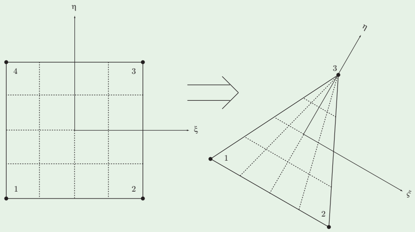
In this case the inverse mapping, $\xi = \mathbf{x}^{-1} : \Omega^e \rightarrow \Box$, exists and is at least C^1 continuous.

Element Degeneration

Many domains do not easily lend themselves to discretizations using quadrilateral elements alone. For this reason it is desirable to develop a triangular element which is compatible with the bilinear quadrilateral element developed above. We degenerate the quadrilateral element by coalescing nodes 3 and 4 to a single node.

Visualizer

Determine the shape functions for the triangular element shown



Higher-Order Elements

It can be desirable to add additional nodes to an element to improve resolution for a given mesh or, for example, to capture a more complex boundary geometry. Of course, this comes at the expense of the additional degrees of freedom leading to additional global equations and hence increased computational expense.

One way to add nodes to achieve a higher-order element is to use the Lagrange polynomials, which are defined as

$$l_a^{n_{en}-1}(\xi) = \frac{\prod_{b=1,b\neq a}^{n_{en}}(\xi - \xi_b)}{\prod_{b=1,b\neq a}^{n_{en}}(\xi_a - \xi_b)}$$

It can easily be seen that $l_a(\xi_a) = 1$ and if $b \neq a$ then $l_a(\xi_b) = 0$, i.e. $l_a(\xi_b) = \delta_{ab}$. We recall that this relationship is a requirement of the shape functions N_a and hence we define the shape functions of an n_{en}-noded element as

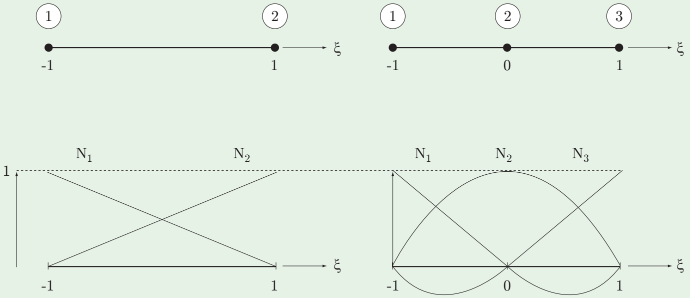
$$N_a = l_a^{n_{en}-1}$$

1-D Higher-Order Elements

Consider the use of Lagrange polynomials as shape functions for 1-D elements

Visualizer

Find first and second-order Lagrange polynomial shape functions for the 1-D element



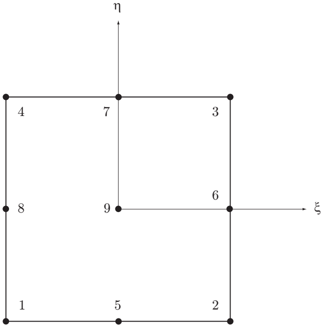
Note the previously-introduced linear shape functions are more formally represented as first-order Lagrange polynomials.

2-D Higher-Order Elements

We can obtain shape functions for two-dimensional elements simply by taking products of the one-dimensional polynomials, i.e.

$$N_a(\xi, \eta) = l_b^{n_{en}-1}(\xi) l_c^{n_{en}-1}(\eta)$$

This results in a family of Lagrange elements. The most complex second-order two-dimensional Lagrange element is the 9-node quadrilateral



Visualizer

Derive shape functions for all quadrilateral elements from the 4-node linear element to the 9 node Lagrange element

Summary

In this section we have considered the detailed implementation of shape functions for both 1-D and 2-D finite elements. In particular we have:

- formalized the concept of an isoparametric element
- introduced the bilinear quadrilateral finite element and shown how this may be degenerated to form compatible triangular elements
- shown how the use of Lagrange polynomials as shape functions has enabled the specification of higher-order finite elements