1-D and 2-D Elements

1-D elements

- coordinate transformation
- linear elements
  - linear basis functions
  - quadratic basis functions
  - cubic basis functions

2-D elements

- coordinate transformation
- triangular elements
  - linear basis functions
  - quadratic basis functions
- rectangular elements
  - linear basis functions
  - quadratic basis functions

1-D elements: coordinate tranformation

We wish to approximate a function u(x) defined in an interval [a,b] by some set of basis functions

$$u(x) = \sum_{i=1}^{n} c_i \varphi_i$$

where i is the number of grid points (the edges of our elements) defined at locations x_i. As the basis functions look the same in all elements (apart from some constant) we make life easier by moving to a local coordinate system

$$\xi = \frac{x - x_i}{x_{i+1} - x_i}$$

so that the element is defined for ξ=[0,1].

1-D linear elements

There is not much choice for the shape of a (straight) 1-D element! Notably the length can vary across the domain.

We require that our function u(ξ) be approximated locally by the linear function

$$u(\xi) = c_1 + c_2 \xi$$

Our node points are defined at $\xi_{1,2}=0,1$ and we require that

$$\begin{aligned} u_1 &= c_1 & &\Rightarrow & c_1 &= u_1 \\ u_2 &= c_1 + c_2 & &\Rightarrow & c_2 &= -u_1 + u_2 \end{aligned}$$

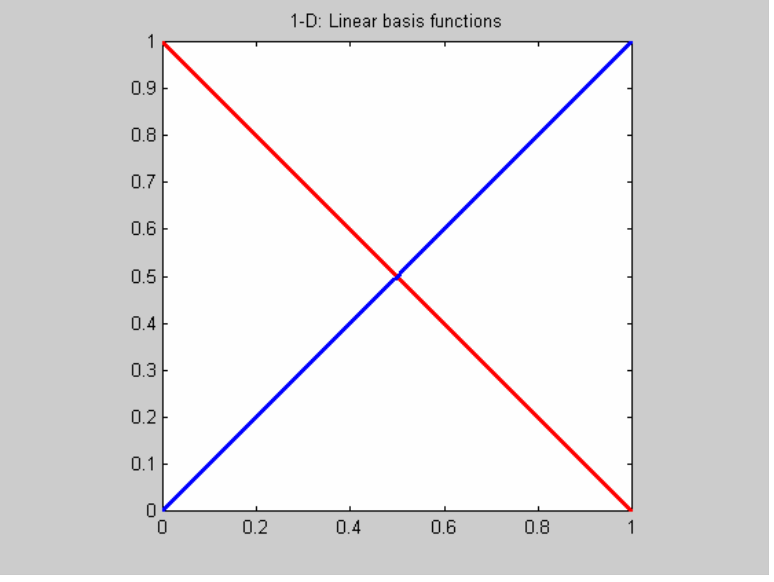
$$\mathbf{c} = \mathbf{A}\mathbf{u}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{-1} & \mathbf{1} \end{bmatrix}$$

1-D elements - linear basis functions

As we have expressed the coefficients c_i as a function of the function values at node points $\xi_{1,2}$ we can now express the approximate function using the node values

$$u(\xi) = u_1 + (-u_1 + u_2)\xi$$
$$= u_1(1-\xi) + u_2\xi$$
$$= u_1 N_1(\xi) + N_2(\xi)\xi$$



.. and $N_{1,2}(x)$ are the linear basis functions for 1-D elements.

1-D quadratic elements

Now we require that our function u(x) be approximated locally by the quadratic function

$$u(\xi) = c_1 + c_2\xi + c_3\xi^2$$

Our node points are defined at $\xi_{1,2,3}$=0,1/2,1 and we require that

$$u_1 = c_1$$
$$u_2 = c_1 + 0.5c_2 + 0.25c_3$$
$$u_3 = c_1 + c_2 + c_3$$

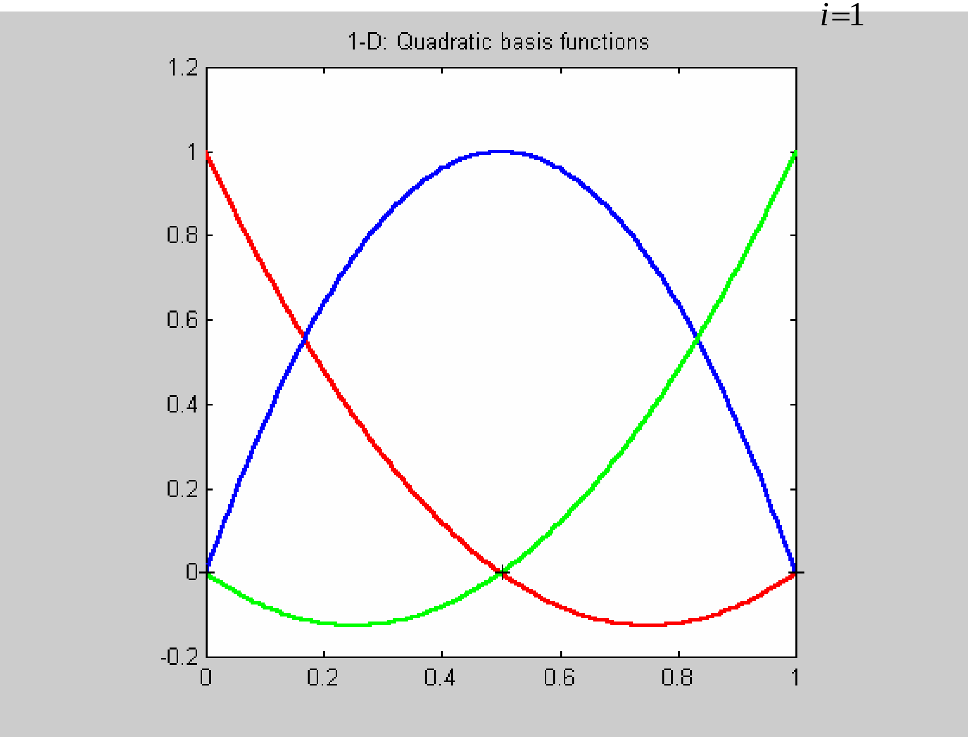
$\Rightarrow$ **c**= **A**u

$$\Rightarrow \mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ -3 & 4 & -1 \\ 2 & -4 & 2 \end{bmatrix}$$

... again we can now express our approximated function as a sum over our basis functions weighted by the values at three node points

$$u(\xi) = c_1 + c_2\xi + c_3\xi^2 = u_1(1-3\xi+2\xi^2) + u_2(4\xi-4\xi^2) + u_3(-\xi+2\xi^2)$$

$$= \sum_{i=1}^{3} u_i N_i(\xi)$$

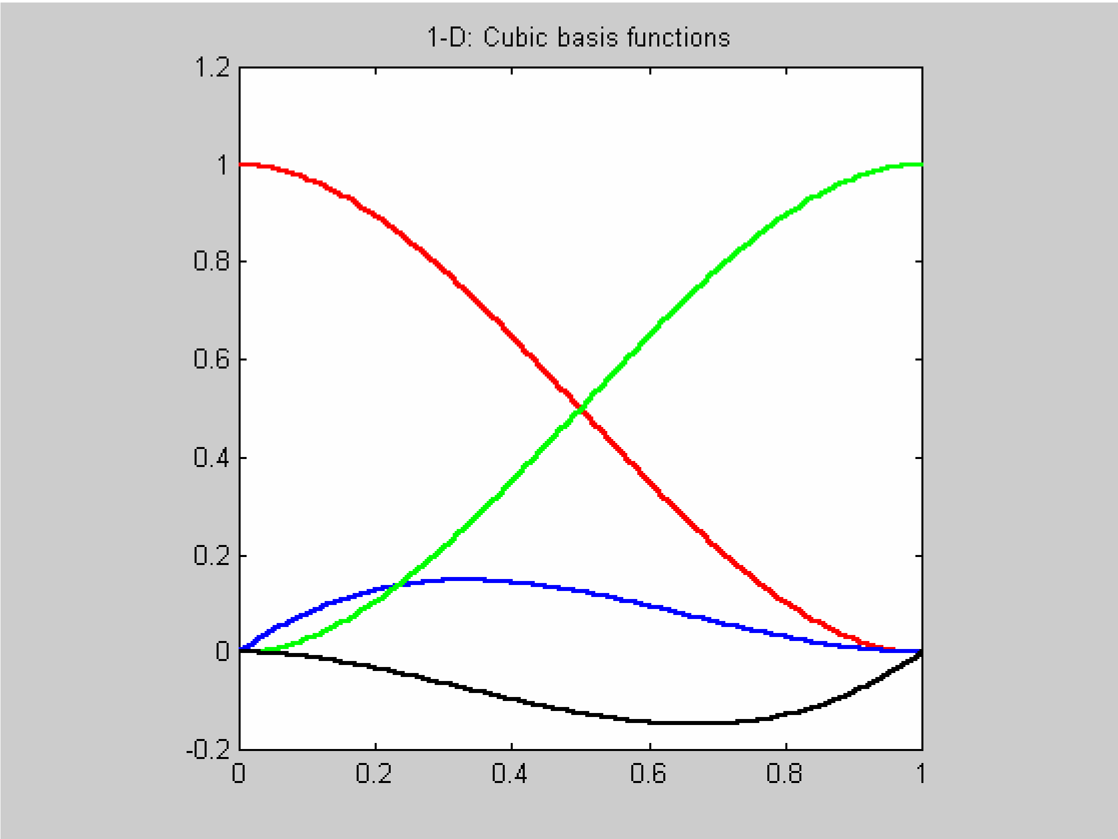


... note that now we re using three grid points per element ...

Can we approximate a constant function?

1-D cubic basis functions

... using similar arguments the cubic basis functions can be derived as



$$u(\xi) = c_1 + c_2\xi + c_3\xi^2 + c_4\xi^3$$

$$N_1(\xi) = 1 - 3\xi^2 + 2\xi^3$$

$$N_2(\xi) = \xi - 2\xi^2 + \xi^3$$

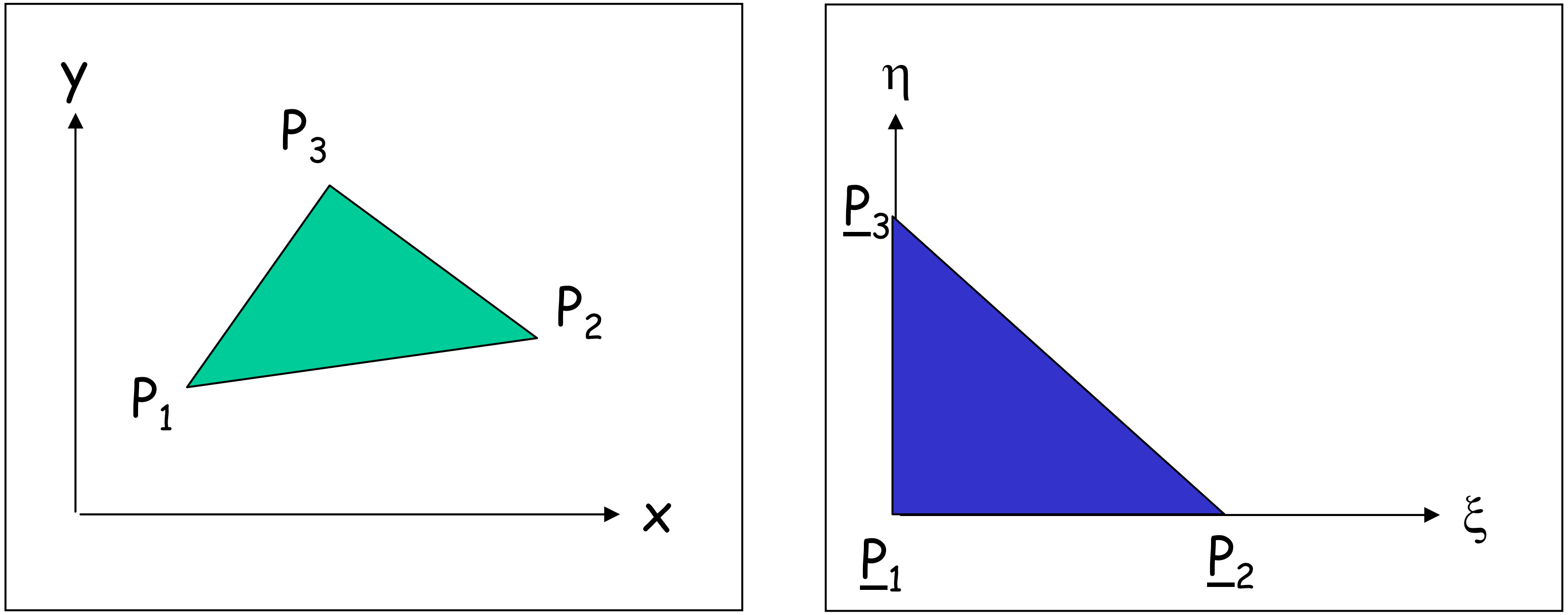
$$N_3(\xi) = 3\xi^2 - 2\xi^3$$

$$N_4(\xi) = -\zeta^2 + \xi^3$$

... note that here we need derivative information at the boundaries ...

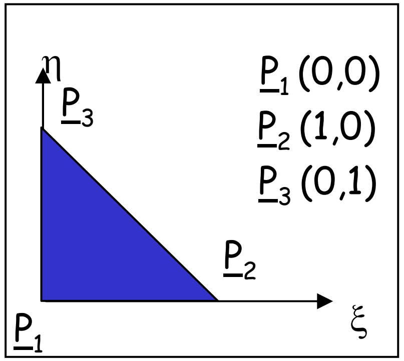
How can we approximate a constant function?

Let us now discuss the geometry and basis functions of 2-D elements, again we want to consider the problems in a local coordinate system, first we look at triangles



before after

2-D elements: coordinate transformation



Any triangle with corners $P_i(x_i,y_i)$, i=1,2,3 can be transformed into a rectangular, equilateral triangle with

$$x = x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta$$

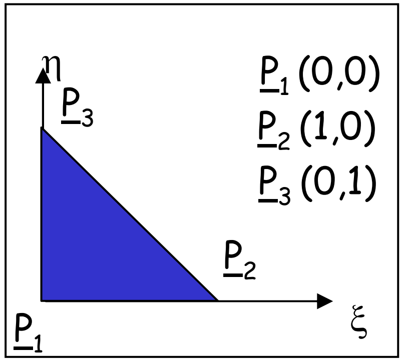
$$y = y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta$$

using counterclockwise numbering. Note that if η=0, then these equations are equivalent to the 1-D tranformations. We seek to approximate a function by the linear form

$$u(\xi,\eta) = c_1 + c_2\xi + c_3\eta$$

we proceed in the same way as in the 1-D case

2-D elements: coefficients



... and we obtain

$$u_1 = u(0,0) = c_1$$
$$u_2 = u(1,0) = c_1 + c_2$$
$$u_3 = u(0,1) = c_1 + c_3$$

... and we obtain the coefficients as a function of the values at the grid nodes by matrix inversion

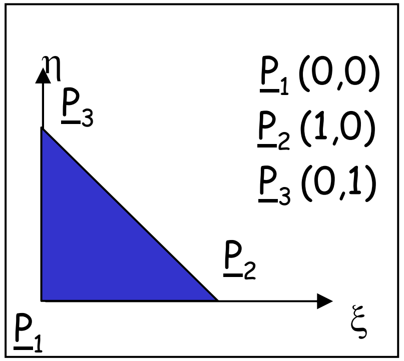
$$\Rightarrow \quad \mathbf{c} = \mathbf{A}\mathbf{u}$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$

containing the 1-D case

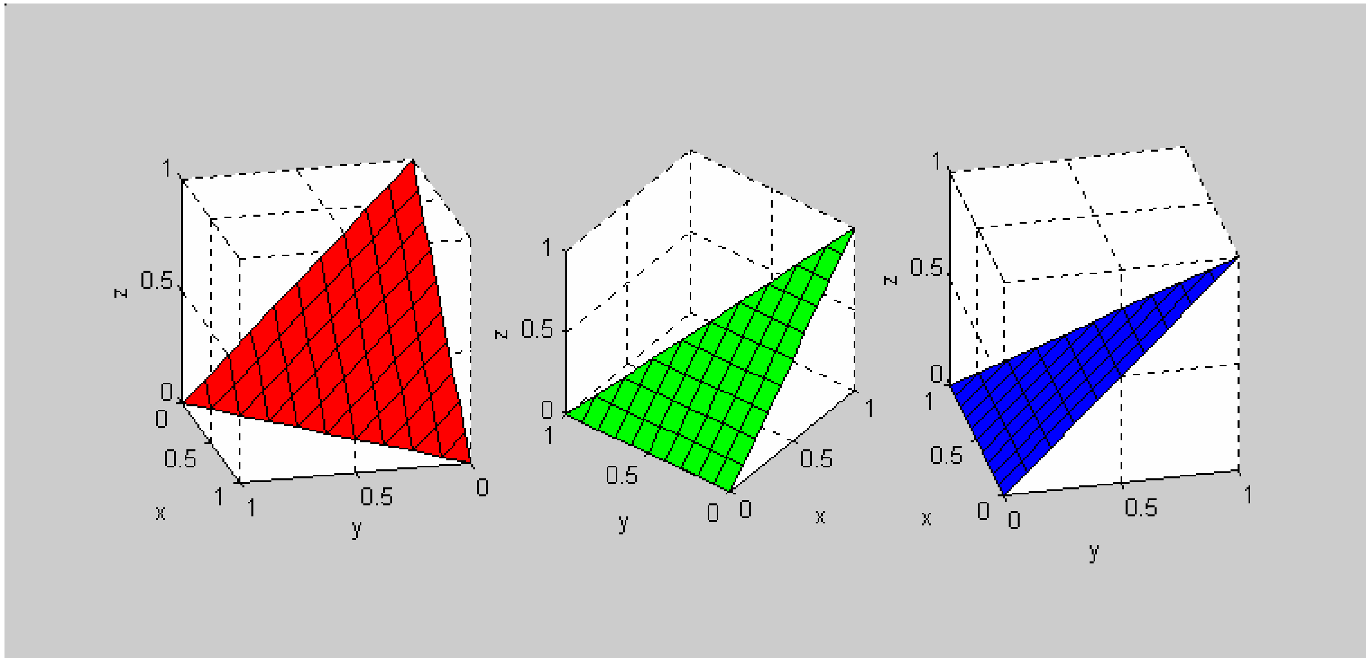
$$\mathbf{A} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{-1} & \mathbf{1} \end{bmatrix}$$

triangles: linear basis functions



from matrix A we can calculate the linear basis functions for triangles

$$
\begin{aligned}
N_1(\xi,\eta) &= 1-\xi-\eta \\
N_2(\xi,\eta) &= \xi \\
N_3(\xi,\eta) &= \eta
\end{aligned}
$$



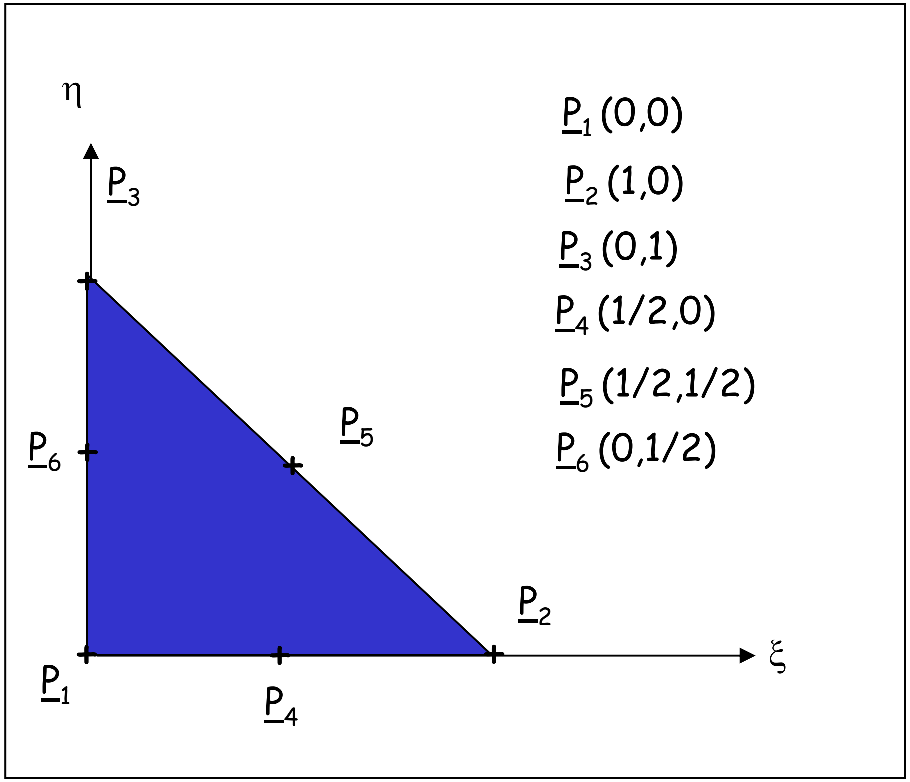
triangles: quadratic elements

Any function defined on a triangle can be approximated by the quadratic function

$$u(x,y) = \alpha_1 + \alpha_2 x + \alpha_3 y + \alpha_4 x^2 + \alpha_5 xy + \alpha_6 y^2$$

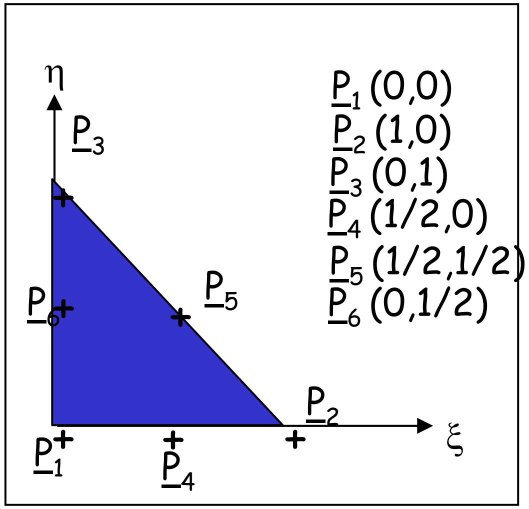
and in the transformed system we obtain

$$u(\xi,\eta) = c_1 + c_2\xi + c_3\eta + c_4\xi^2 + c_5\xi\eta + c_6\eta^2$$



as in the 1-D case we need additional points on the element.

triangles: quadratic elements



To determine the coefficients we calculate the function u at each grid point to obtain

$$u_1 = c_1$$

$$u_2 = c_1 + c_2 + c_4$$

$$u_3 = c_1 + c_3 + c_6$$

$$u_4 = c_1 + 1/2c_2 + 1/4c_4$$

$$u_5 = c_1 + 1/2c_2 + 1/2c_3 + 1/4c_4 + 1/4c_5 + 1/4c_6$$

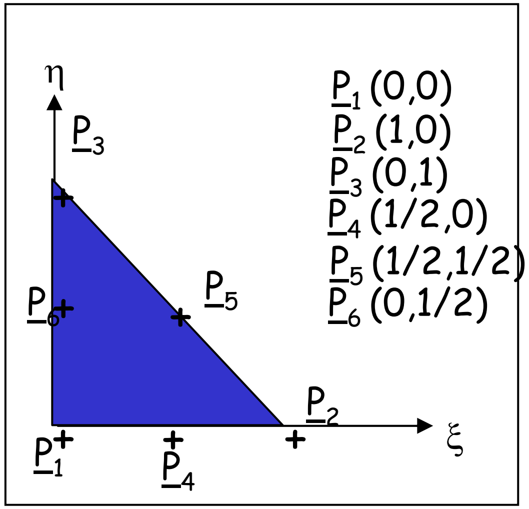
$$u_6 = c_1 + 1/2c_3 + 1/6c_6$$

... and by matrix inversion we can calculate the coefficients as a function of the values at P_i

$$\mathbf{c} = \mathbf{A}\mathbf{u}$$

c= Au

$$A=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ -3 & -1 & 0 & 4 & 0 & 0 \\ -3 & 0 & -1 & 0 & 0 & 4 \\ 2 & 2 & 0 & -4 & 0 & 0 \\ 4 & 0 & 0 & -4 & 4 & -4 \\ 2 & 0 & 2 & 0 & 0 & -4 \end{bmatrix}$$

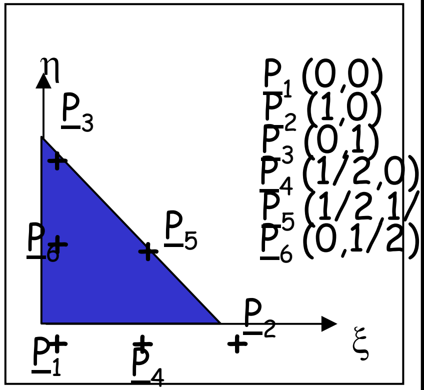


... to obtain the basis functions

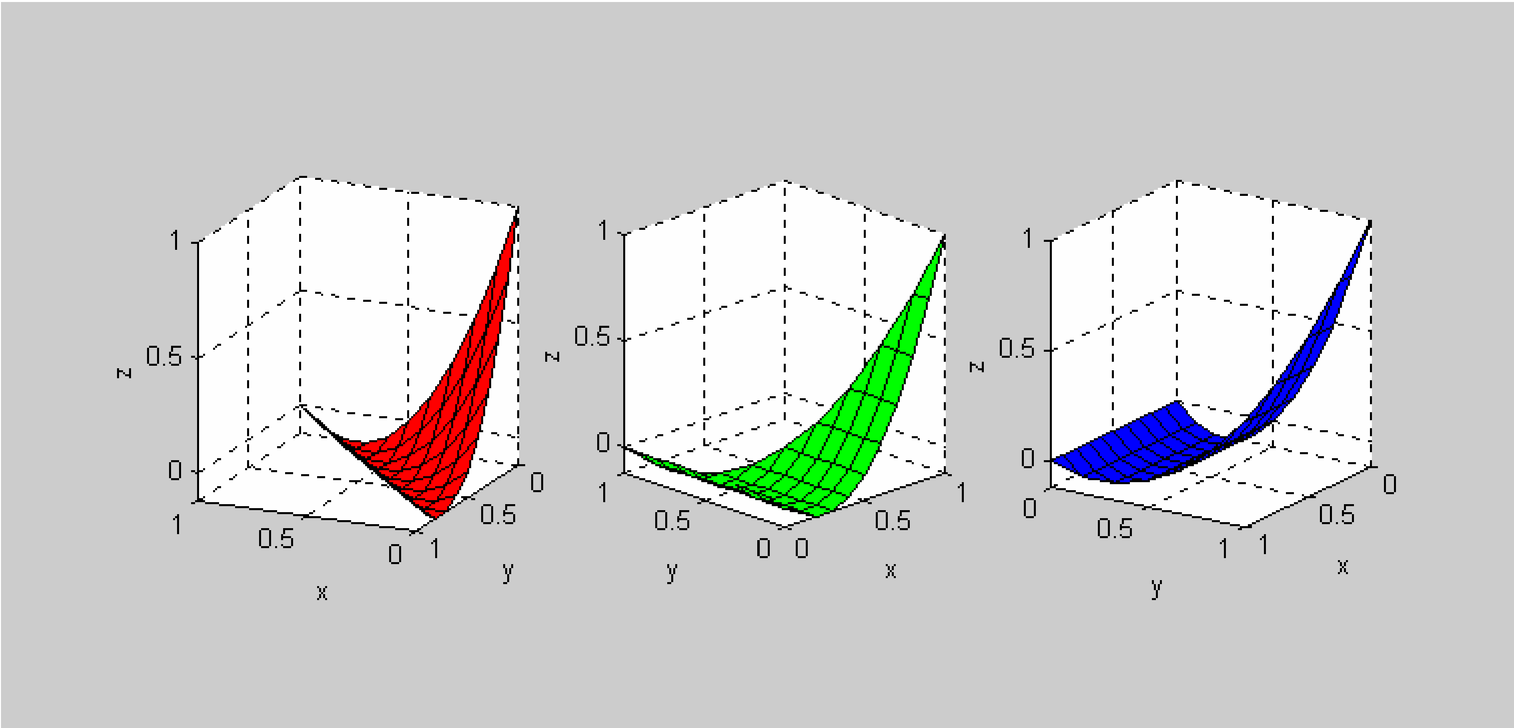
$$N_1(\xi,\eta)=(1-\xi-\eta)(1-2\xi-2\eta)$$
$$N_2(\xi,\eta)=\xi(2\xi-1)$$
$$N_3(\xi,\eta)=\eta(2\eta-1)$$
$$N_4(\xi,\eta)=4\xi(1-\xi-\eta)$$
$$N_5(\xi,\eta)=4\xi\eta$$
$$N_2(\xi,\eta)=4\eta(1-\xi-\eta)$$

... and they look like ...

triangles: quadratic basis functions

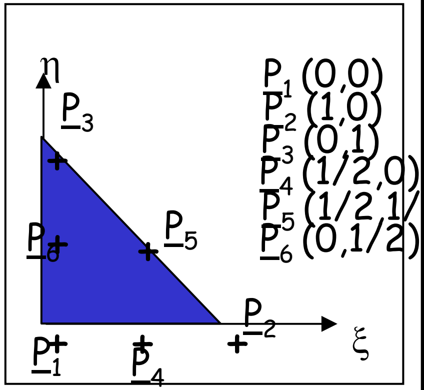


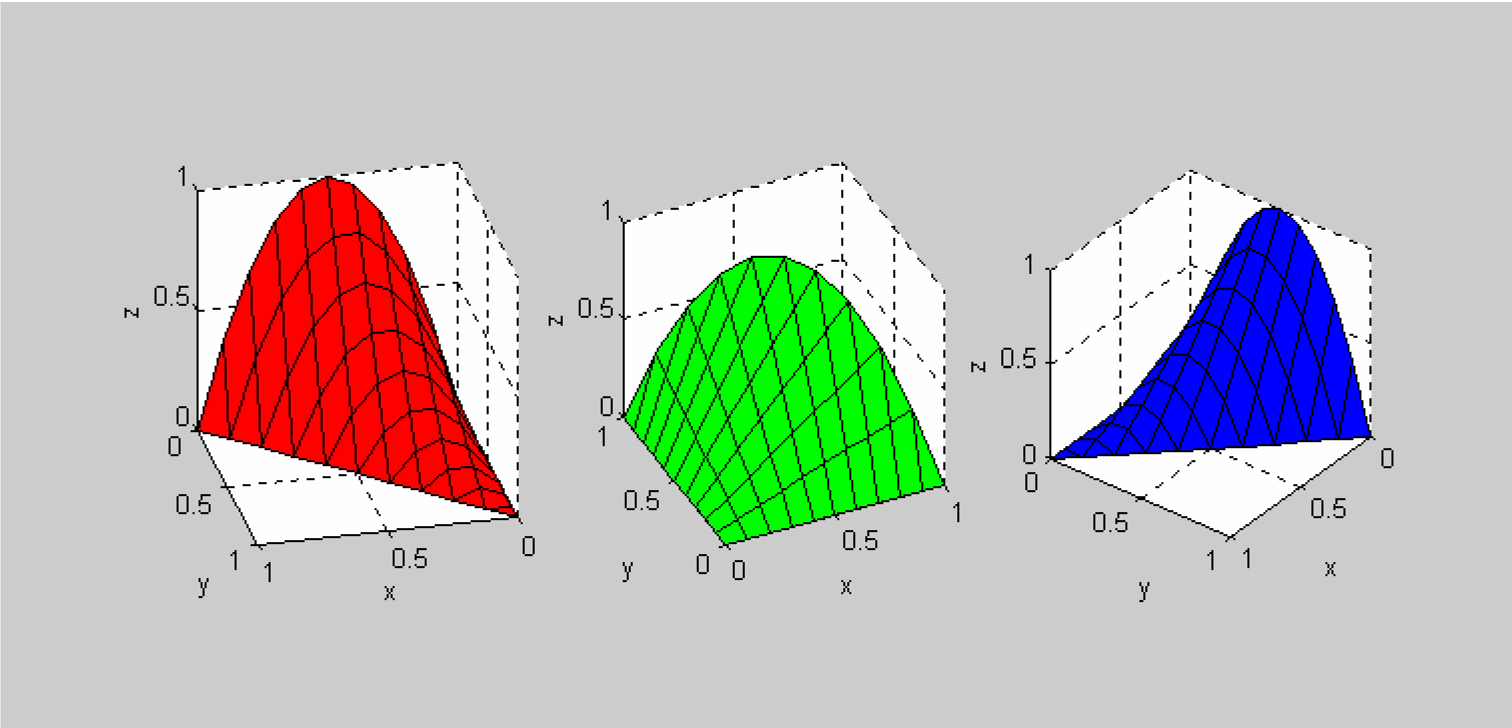
The first three quadratic basis functions ...



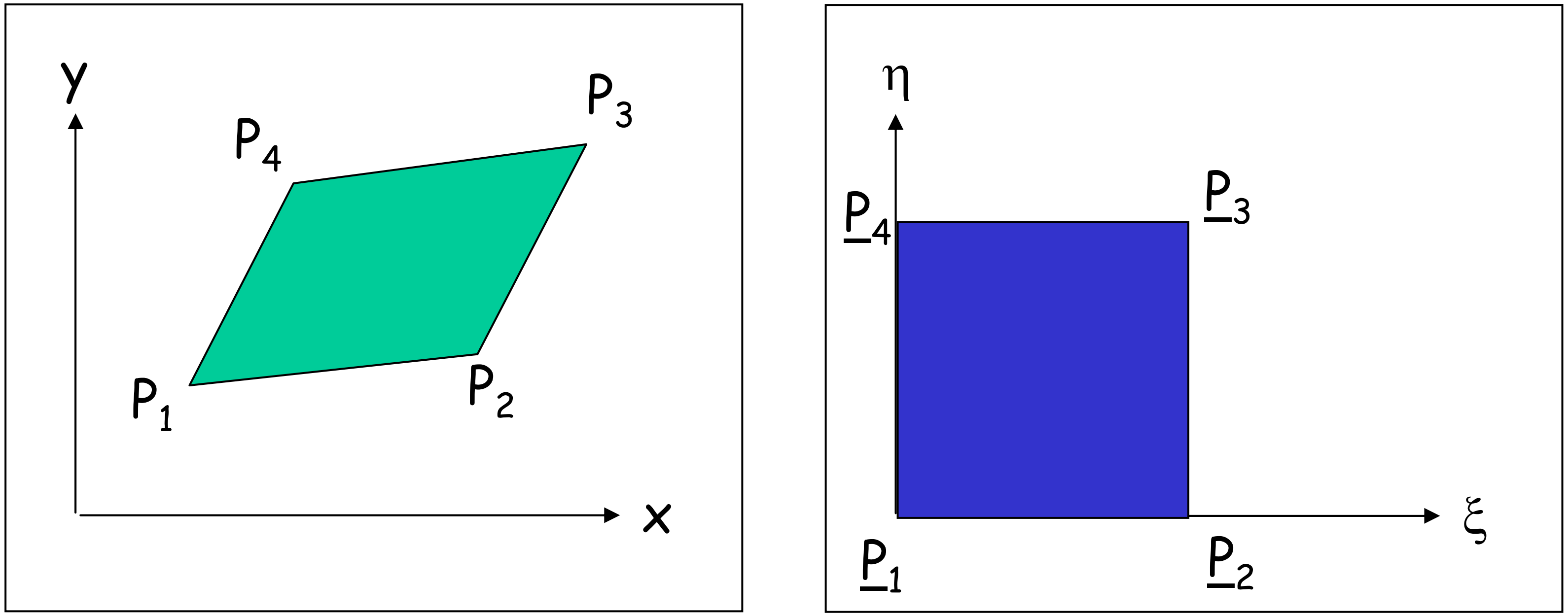
triangles: quadratic basis functions

.. and the rest ...





Let us consider rectangular elements, and transform them into a local coordinate system



before

after

rectangles: linear elements

With the linear *Ansatz*

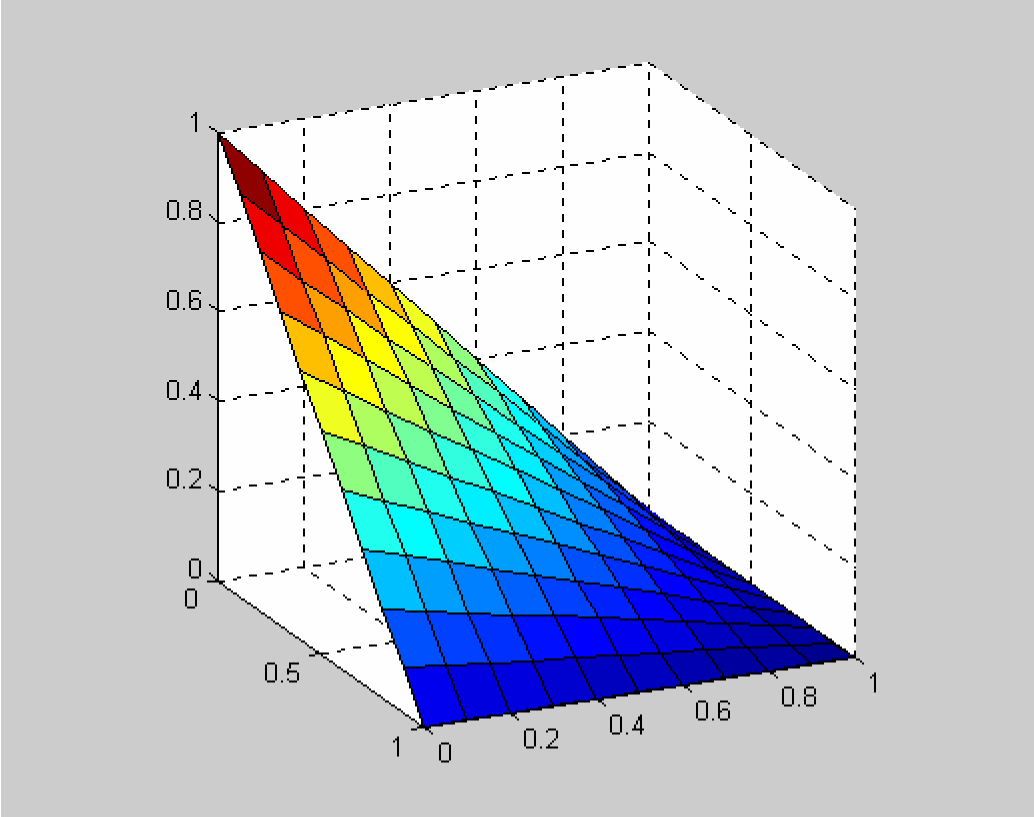
$$u(\xi,\eta) = c_1 + c_2\xi + c_3\eta + c_4\xi\eta$$

we obtain matrix *A* as

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 \\ 1 & -1 & 1 & -1 \end{bmatrix}$$

and the basis functions

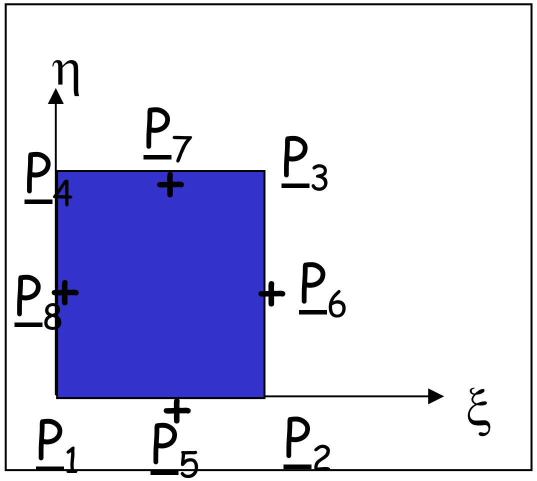
$$N_1(\xi,\eta) = (1-\xi)(1-\eta)$$
$$N_2(\xi,\eta) = \xi(1-\eta)$$
$$N_3(\xi,\eta) = \xi\eta$$
$$N_4(\xi,\eta) = (1-\xi)\eta$$



rectangles: quadratic elements

With the quadratic *Ansatz*

$$u(\xi,\eta)=c_1+c_2\xi+c_3\eta+c_4\xi^2+c_5\xi\eta+c_6\eta^2+c_7\xi^2\eta+c_8\xi\eta^2$$

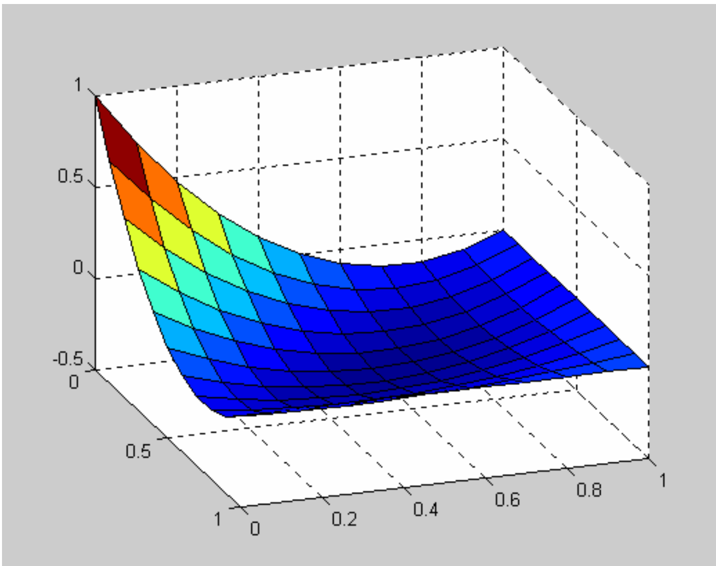


we obtain an 8x8 matrix A ... and a basis function look e.g. like

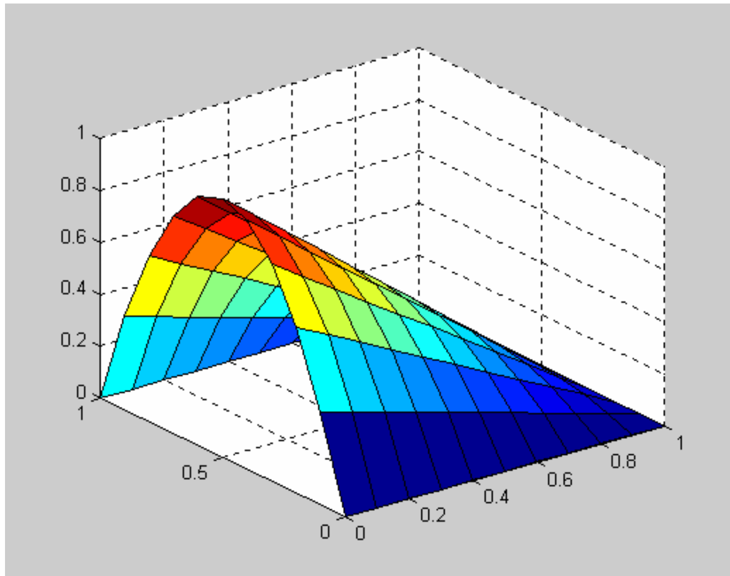
$$N_1(\xi,\eta)=(1-\xi)(1-\eta)(1-2\xi-2\eta)$$

$$N_5(\xi,\eta)=4\xi(1-\xi)(1-\eta)$$

N_1



N_2



1-D and 2-D elements: summary

The basis functions for finite element problems can be obtained by:

1. Transforming the system in to a local (to the element) system
2. Making a linear (quadratic, cubic) *Ansatz* for a function defined across the element.
3. Using the interpolation condition (which states that the particular basis functions should be one at the corresponding grid node) to obtain the coefficients as a function of the function values at the grid nodes.
4. Using these coefficients to derive the *n* basis functions for the *n* node points (or conditions).