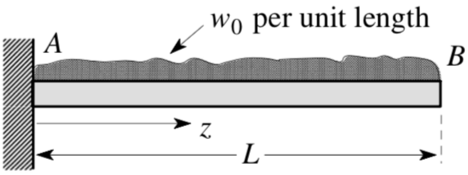
CE 620 Finite Element Method

1. Rayleigh-Ritz method

A cantilever beam is built-in at $z = 0$ and subjected to a uniformly distributed load w_0 per unit length, as shown in Figure 3.15. Use the Rayleigh-Ritz method to find an approximate expression for the deflection of the beam.



Consider the problem shown in the figure above. We want to use an approximate method to find out the deflection. We note that the equation of the beam is:

$M = -\frac{d^2}{dz^2} u(z)$, where u is +ve upwards.

The energy of the beam is:

$U = \frac{EI}{2} \int_0^L \left(\frac{d^2 u}{dz^2}\right)^2 dz.$

Make an approximation of the form:

$u = C z^2$,

This satisfies the kinematic boundary conditions.

```
u[z_] := c z^2;
U = EI/2 Integrate[(u''[z])^2, {z, 0, L}]
```

$2\,c^2\,EI\,L$

The potential energy term from the work done due to the applied load is:

$\Omega = \int w_0\, u\, dz$

```
Ω = Integrate[wo u[z], {z, 0, L}]
```

$\frac{1}{3}\,c\,L^3\,wo$

The total potential energy is: P = U + Ω.

```
P = U + Ω
```

$2\,c^2\,EI\,L + \frac{1}{3}\,c\,L^3\,wo$

Minimizing the potential energy with respect to C, we get:

$0 = \frac{dP}{dC}.$

```
D[P, c]
S = Solve[% == 0, c]
P /. S[[1]]
```

$$4\,c\,EI\,L + \frac{L^3\,wo}{3}$$

$$\left\{\left\{c \to -\frac{L^2\,wo}{12\,EI}\right\}\right\}$$

$$-\frac{L^5\,wo^2}{72\,EI}$$

The displacement function is:

```
u[z] /. S[[1]]
```

$$-\frac{L^2\,wo\,z^2}{12\,EI}$$

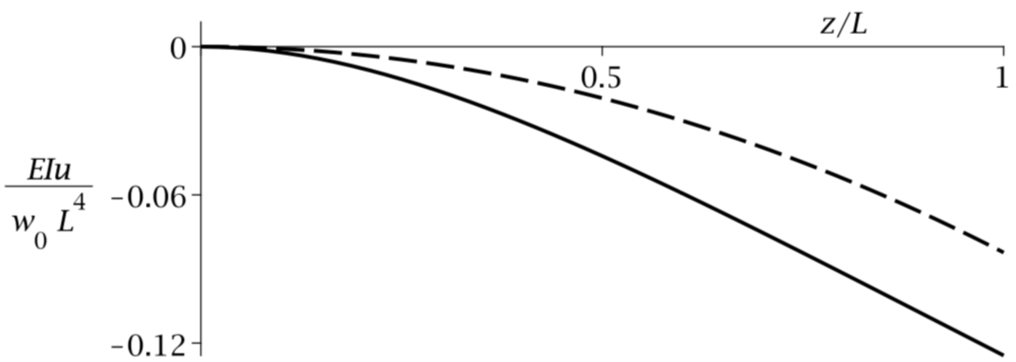
The actual solution is actually

$$\texttt{uact[z\_]} = -\frac{wo}{EI}\left(\frac{L^2\,z^2}{4} - \frac{L\,z^3}{6} + \frac{z^4}{24}\right);$$

$$\texttt{Pact} = \frac{EI}{2}\,\texttt{Integrate}\left[\texttt{uact''[z]}^2, \{z, 0, L\}\right] + \texttt{Integrate[wo uact[z], \{z, 0 , L\}]}$$

$$-\frac{L^5\,wo^2}{40\,EI}$$

We can see, clearly that the energy of the approximate solution is lesser than the energy of the actual solution as expected.



A better approximation:

$$u(z) = C_1\,z^2 + C_2\,z^3$$

with:

$$u''(z) = 2\,C_1 + 6\,C_2\,z$$

$u''(z) = 0$ at $z = L$. This implies that $C_2 = -C_1/3\,L$. Using this approximation we get the following.

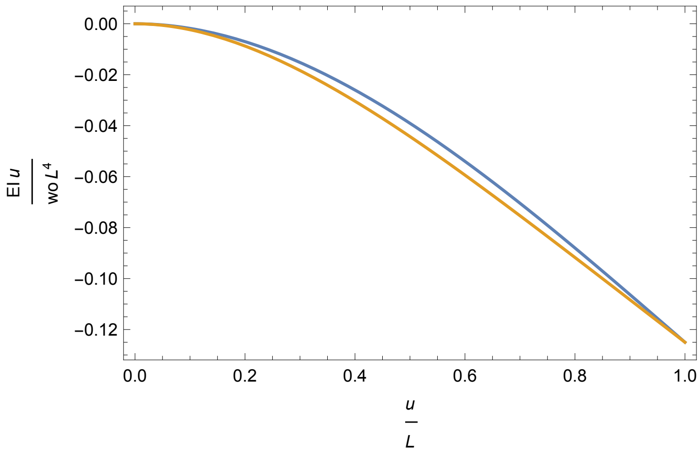
```
u[z_] = c (z^2 - z^3/(3 L));
P = EI/2 Integrate[u''[z]^2, {z, 0, L}] + Integrate[wo u[z], {z, 0, L}]
S = Solve[D[P, c] == 0, c]
P /. S[[1]]
```

$$\frac{2}{3} c^2 \text{EI L} + \frac{1}{4} c L^3 \text{wo}$$

$$\left\{\left\{c \to -\frac{3 L^2 \text{wo}}{16 \text{EI}}\right\}\right\}$$

$$-\frac{3 L^5 \text{wo}^2}{128 \text{EI}}$$

```
uact[z] /. {wo → 1, L → 1, EI → 1};
u[z] /. S[[1]] /. {wo → 1, L → 1, EI → 1};
Plot[{%, %%}, {z, 0, 1}, Frame → True, FrameLabel → {"u/L", "EI u/(wo L^4)"}]
```



Minimum potential energy is an amazing criteria for getting approximate solution, but an energy need to exist (not always true for many problems, especially those not in structures.)

Drawbacks of Rayleigh-Ritz:
1. Very problem specific and not generalizable.
2. Getting approximation function can be tricky.
3. Unless exact solution is known convergence rate is not clear.
4. More and more terms need to be involved and rounding off errors will become more dominant.

Finite element method is actually a variation of the RR method with an element-wise interpolation functions giving rise to the entire field of interest.

Weighted residue method

The differential equation could now be written in the following format:
$\int\left(\text{EI} \frac{d^2 u}{d z^2} + M(z)\right) \delta u \, dz = \delta W$, the virtual work done.

Using, integration by parts, we get:

$$-\mathrm{EI}\int\frac{du}{dz}\frac{d\,\delta u}{dz}\,dz + \int M(z)\,\delta u\,dz + \frac{du}{dz}\,\delta u\,\Big|_a^L = 0.$$

If we assume that $\delta u = \delta c\, z^2$, the same for $u = c\, z^2$, i.e., for the actual and the real function have the same form, the method is called a Galerkin method.

```
u[z_] = c z^2;
δu[z_] = δc z^2;
M[z_] = wo (L - z)^2 / 2;
δW = -EI Integrate[u'[z] × δu'[z], {z, 0, L}] +
   Integrate[M[z] × δu[z], {z, 0, L}] + EI u'[L] × δu[L]
Coefficient[%, δc]
Solve[% == 0, c]
```

$$\frac{2}{3}\,c\,\mathrm{EI}\,L^3\,\delta c + \frac{1}{60}\,L^5\,\mathrm{wo}\,\delta c$$

$$\frac{2}{3}\,c\,\mathrm{EI}\,L^3 + \frac{L^5\,\mathrm{wo}}{60}$$

$$\left\{\left\{c \to -\frac{L^2\,\mathrm{wo}}{40\,\mathrm{EI}}\right\}\right\}$$

2. Finite elements in 1D

For a single element, the following are the expressions. In this case, we will have the following equations. The terminology used is:

$\theta^e(x)$: field for *a* given variable θ^e

$\boldsymbol{p}(x)$: $(1,\ x)$

$\boldsymbol{d}_e(x)$: $(\theta_1^e,\ \theta_2^e)^T$

$\boldsymbol{M}(x)$: $\begin{pmatrix} 1 & x_1^e \\ 1 & x_2^e \end{pmatrix}$

$\boldsymbol{\alpha}$: $(\alpha_0^e,\ \alpha_1^e)^T$

Using, the formulation, we get the following:

$\boldsymbol{\theta}^e = \boldsymbol{p}\cdot\boldsymbol{\alpha}^e$

$\boldsymbol{\alpha}^e = (\boldsymbol{M})^{-1}\,\boldsymbol{d}_e$

$\boldsymbol{N}^e(x) = \boldsymbol{p}\,(\boldsymbol{M})^{-1}$

$\theta^e(x) = \boldsymbol{N}^e(x)\,\boldsymbol{d}^e$ and

$\frac{d\,\theta^e(x)}{dx} = \frac{d\,\boldsymbol{N}^e(x)}{dx}\,\boldsymbol{d}^e$, where $\boldsymbol{B}^e = \frac{d\,\boldsymbol{N}^e(x)}{dx}$.

The final equation is the ultimate expression, in which any variable is expression in terms of the shape functions for any finite element *e*. The overall field $\theta^h(x)$ in the discretized form is:

$\theta^h(x) = \sum_e \boldsymbol{N}^e(x)\,\boldsymbol{d}^e$.

This is called as the global interpolation function for the finite element analysis.

Below is the implementation for various types of elements.

```
(* for linear elements *)
p[x_] = {1, x};
αem = {αe[0], αe[1]};
de = {θe[1], θe[2]};
M = {{1, xe[1]}, {1, xe[2]}};
Minv = Inverse[M];
```

These are the shape functions.

```
Ne[x_] = p[x].Minv /. (xe[2] - xe[1]) → Le // FullSimplify
Be[x_] = D[Ne[x], x]
```

$$\left\{\frac{-x+xe[2]}{Le}, \frac{x-xe[1]}{Le}\right\}$$

$$\left\{-\frac{1}{Le}, \frac{1}{Le}\right\}$$

Final field within an element.

```
Minv /. (xe[2] - xe[1]) → Le // MatrixForm // FullSimplify
θel = Ne[x].de /. (xe[2] - xe[1]) → Le
```

$$\begin{pmatrix} \frac{xe[2]}{Le} & -\frac{xe[1]}{Le} \\ -\frac{1}{Le} & \frac{1}{Le} \end{pmatrix}$$

$$\frac{(-x+xe[2])\,\theta e[1]}{Le} + \frac{(x-xe[1])\,\theta e[2]}{Le}$$

Now, some of the properties of the shape functions. Shape functions at the node points.

```
Ne[xe[1]] /. (xe[2] - xe[1]) → Le
Ne[xe[2]] /. (xe[2] - xe[1]) → Le
```

{1, 0}

{0, 1}

Sum of shape functions is equal to one.

```
(Ne[x][[1]] + Ne[x][[2]]) // FullSimplify;
% /. (xe[2] - xe[1]) → Le
```

1

For quadratic.

```
(* for linear elements *)
p[x_] = {1, x, x^2};
αem = {αe[0], αe[1], αe[2]};
de = {θe[1], θe[2], θe[3]};
M = {p[xe[1]], p[xe[2]], p[xe[3]]};
Minv = Inverse[M];
```

```
Ne[x_] = p[x].Minv // FullSimplify
```

$$\left\{\frac{(x-xe[2])\,(x-xe[3])}{(xe[1]-xe[2])\,(xe[1]-xe[3])},\ -\frac{(x-xe[1])\,(x-xe[3])}{(xe[1]-xe[2])\,(xe[2]-xe[3])},\ -\frac{(x-xe[1])\,(x-xe[2])}{(xe[1]-xe[3])\,(-xe[2]+xe[3])}\right\}$$

```
Minv // FullSimplify // MatrixForm
```

$$\begin{pmatrix} \frac{xe[2]\times xe[3]}{(xe[1]-xe[2])\,(xe[1]-xe[3])} & \frac{xe[1]\times xe[3]}{(xe[1]-xe[2])\,(-xe[2]+xe[3])} & \frac{xe[1]\times xe[2]}{(xe[1]-xe[3])\,(xe[2]-xe[3])} \\ -\frac{xe[2]+xe[3]}{(xe[1]-xe[2])\,(xe[1]-xe[3])} & \frac{xe[1]+xe[3]}{(xe[1]-xe[2])\,(xe[2]-xe[3])} & \frac{xe[1]+xe[2]}{(xe[1]-xe[3])\,(-xe[2]+xe[3])} \\ \frac{1}{(xe[1]-xe[2])\,(xe[1]-xe[3])} & \frac{1}{(-xe[1]+xe[2])\,(xe[2]-xe[3])} & \frac{1}{(xe[1]-xe[3])\,(xe[2]-xe[3])} \end{pmatrix}$$

```
θel = Ne[x].de
```

$$\frac{(x-xe[2])\,(x-xe[3])\,\theta e[1]}{(xe[1]-xe[2])\,(xe[1]-xe[3])} - \frac{(x-xe[1])\,(x-xe[3])\,\theta e[2]}{(xe[1]-xe[2])\,(xe[2]-xe[3])} - \frac{(x-xe[1])\,(x-xe[2])\,\theta e[3]}{(xe[1]-xe[3])\,(-xe[2]+xe[3])}$$

```
Ne[xe[1]] // FullSimplify
Ne[xe[2]] // FullSimplify
Ne[xe[3]] // FullSimplify
Sum[Ne[x][[i]], {i, 1, 3}] // FullSimplify
```

{1, 0, 0}

{0, 1, 0}

{0, 0, 1}

1

The general shape function would be Lagrange polynomials, which are defined for any order of polynomial as:

$N_i^e(x) = \prod_{j \neq i} \frac{(x-x_j)}{(x_i-x_j)}$.

The global definition of the interpolation function is:

$\theta^h(x) = \sum_e \boldsymbol{N}^{\mathbf{e}}(x)\, \boldsymbol{d}^e$

However, using a more compact notation:

$\boldsymbol{d}^e = \boldsymbol{L}^e\, \boldsymbol{d}$, where **d** corresponds to the global vector for the all the nodal variables. If the total number of nodes is M and the total number of nodes per element is k, then $\boldsymbol{L}^{\mathrm{e}}$ is a matrix of size $k \times M$, where for every row i all the entries are zero except corresponding to the column J where the node number m of element e is mapped to.

With these definitions,

$\theta^h(x) = \boldsymbol{N}(x)\, \boldsymbol{d}$, where $\boldsymbol{N}(x) = \sum_e \boldsymbol{N}^e\, \boldsymbol{L}^e$, where $\boldsymbol{N}^e$is a row vector of size $1 \times N$. The general properties are still satisfied:

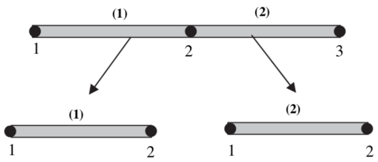
$\sum_i N_i(x) = 1$, and $N_i(x_j) = \delta_{\mathrm{ij}}$.

Since each shape function $N_i(x)$ is C^0continuous, the field $\theta^h(x)$ is also too. Do the elements have higher order continuity is something that we will check later.

Similarly, for the weight function:

$w^h(x) = \boldsymbol{N}(x)\,\boldsymbol{w}.$

Demonstrate this with a simple example:



$$\mathbf{d}^{(1)} = \begin{bmatrix} \theta_1^{(1)} \\ \theta_2^{(1)} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}}_{\mathbf{L}^{(1)}} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \mathbf{L}^{(1)}\mathbf{d},$$

$$\mathbf{d}^{(2)} = \begin{bmatrix} \theta_1^{(2)} \\ \theta_2^{(2)} \end{bmatrix} = \begin{bmatrix} \theta_2 \\ \theta_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\mathbf{L}^{(2)}} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \mathbf{L}^{(2)}\mathbf{d}.$$

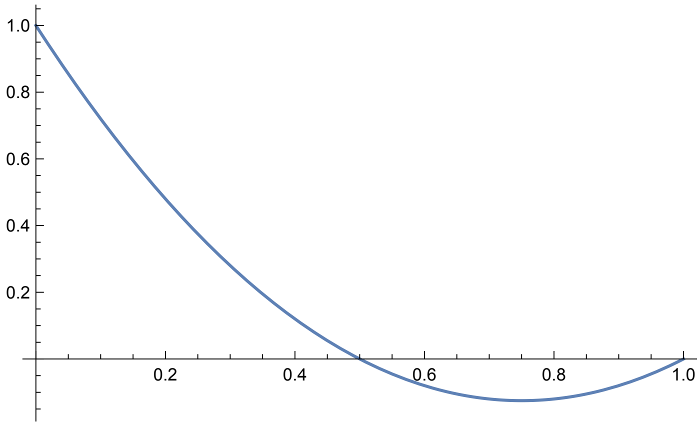
and

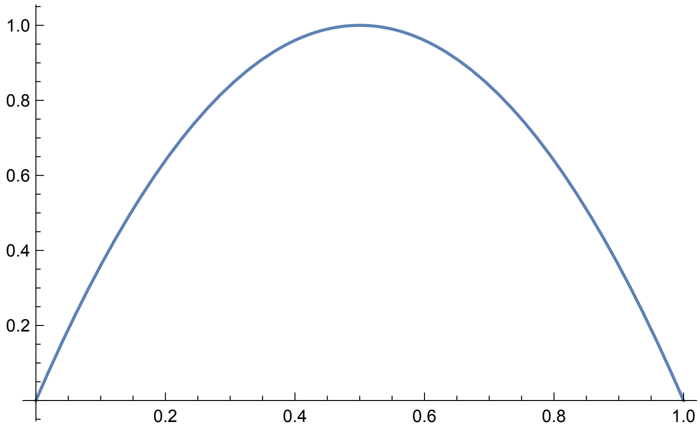
$$\mathbf{N} = \mathbf{N}^{(1)}\mathbf{L}^{(1)} + \mathbf{N}^{(2)}\mathbf{L}^{(2)} = \begin{bmatrix} \underbrace{N_1^{(1)}}_{N_1} & \underbrace{N_2^{(1)} + N_1^{(2)}}_{N_2} & \underbrace{N_2^{(2)}}_{N_3} \end{bmatrix}.$$

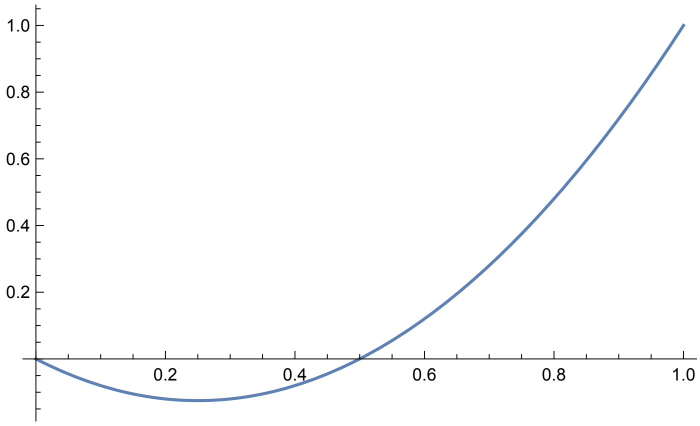
The number of shape functions is equal to the number of nodes.

The plot below, is for Lagrange quadratic shape functions.

```
Plot[(x - 0.5) (x - 1) / ((0 - 0.5) (0 - 1)), {x, 0, 1}]
Plot[(x - 0) (x - 1) / ((0.5 - 0) (0.5 - 1)), {x, 0, 1}]
Plot[(x - 0) (x - 0.5) / ((1 - 0) (1 - 0.5)), {x, 0, 1}]
```







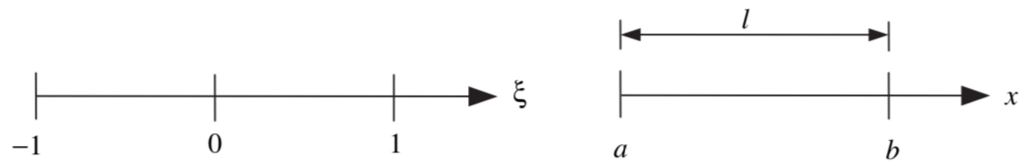
Gauss Quadrature

There are a number of integrals that one needs to evaluate with respect to the weak form. Consider the following integral

$$I = \int_a^b f(x)\, dx = ?$$

The Gauss quadrature formulas are always given over the parent domain [-1, 1].

$$x = \frac{1}{2}(a+b) + \frac{1}{2}\xi(b-a).$$



This can be expressed in the following form:

$$x = x_1 N_1(\xi) + x_2 N_2(\xi) = a\frac{1-\xi}{2} + b\frac{\xi+1}{2}.$$

From this we get,

$$\mathrm{d}x = \frac{1}{2}(b-a)\,\mathrm{d}\xi = \frac{l}{2}\mathrm{d}\xi = J\mathrm{d}\xi,$$

where, the Jacobian J is given as:

$$I = J\int_{-1}^{1} f(\xi)\mathrm{d}\xi = J\hat{I}, \qquad \text{where} \quad \hat{I} = \int_{-1}^{1} f(\xi)\mathrm{d}\xi.$$

which we want to be expressed in the following form.

$$\hat{I} = W_1 f(\xi_1) + W_2 f(\xi_2) + \cdots = \underbrace{\begin{bmatrix} W_1 & W_2 & \cdots & W_n \end{bmatrix}}_{\mathbf{W}^T} \underbrace{\begin{bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{bmatrix}}_{\mathbf{f}} = \mathbf{W}^{\mathrm{T}}\mathbf{f},$$

The goal of gauss quadrature is to choose the *n* integration points ξ_i and the corresponding weights W_i such that for the given *n* points, the highest order polynomial of some order *m* is exactly integrated.

$$f(\xi) = \alpha_1 + \alpha_2\xi + \alpha_3\xi^2 + \cdots = \underbrace{\begin{bmatrix} 1 & \xi & \xi^2 & \cdots \end{bmatrix}}_{\mathbf{p}} \underbrace{\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \end{bmatrix}}_{\boldsymbol{\alpha}} = \mathbf{p}(\boldsymbol{\xi})\boldsymbol{\alpha}. \quad ,$$

This can be expressed in the matrix form as

$$\underbrace{\begin{bmatrix} f(\xi_1) \\ f(\xi_2) \\ \vdots \\ f(\xi_n) \end{bmatrix}}_{\mathbf{f}} = \underbrace{\begin{bmatrix} 1 & \xi_1 & \xi_1^2 & \cdots \\ 1 & \xi_2 & \xi_2^2 & \cdots \\ \vdots & \vdots & \vdots & \vdots \\ 1 & \xi_n & \xi_n^2 & \cdots \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix}}_{\boldsymbol{\alpha}}.$$

The matrices have the following dimensions:

$\boldsymbol{f} \equiv n \times 1$

$\boldsymbol{M} \equiv n \times m$

$\boldsymbol{\alpha} \equiv n \times m$

and can be written in a compact form as:

$$\hat{I} = \mathbf{W}^{\mathrm{T}} \mathbf{M} \boldsymbol{\alpha}.$$

The integral can be written as:

$$\hat{I} = \int_{-1}^{1} f(\xi)\, d\xi = \int_{-1}^{1} \begin{bmatrix} 1 & \xi & \xi^2 & \xi^3 & \cdots \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_n \end{bmatrix} d\xi = \begin{bmatrix} \xi & \frac{\xi^2}{2} & \frac{\xi^3}{3} & \frac{\xi^4}{4} & \cdots \end{bmatrix}_{-1}^{1} \boldsymbol{\alpha}$$

$$= \underbrace{\begin{bmatrix} 2 & 0 & \frac{2}{3} & 0 & \cdots \end{bmatrix}}_{\hat{\mathbf{P}}} \boldsymbol{\alpha} = \hat{\mathbf{P}} \boldsymbol{\alpha}.$$

But could also be written as $\boldsymbol{W}^T \boldsymbol{M} \boldsymbol{\alpha} = \hat{\boldsymbol{P}} \boldsymbol{\alpha}$. Thus the integration points could be chosen such that:

$$\boxed{\mathbf{M}^{\mathrm{T}} \mathbf{W} = \hat{\mathbf{P}}^{\mathrm{T}}}.$$

It could be shown that Gauss-Legendre polynomials given below can be used to perform these integrations.

For a polynomial of order n, it can be shown that the minimum number of points for Gauss Integration $m_{\min} \geq \frac{n+1}{2}$.

```
P[n_, x_] := LegendreP[n, x]
S[n_] := Solve[P[n, x] == 0, x] // N
W[n_] := 2 / ((1 - x^2) D[P[n, x], x]^2) /. S[n]
```

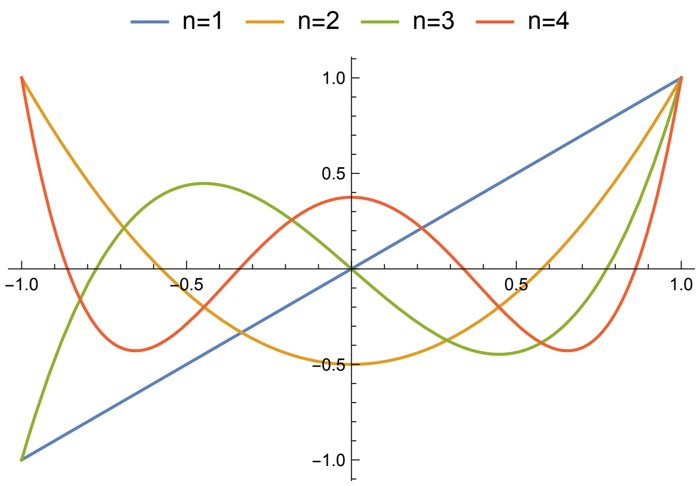
```
n = 3;
x /. S[n]
W[n]
```

```
{0., -0.774597, 0.774597}
```

```
{0.888889, 0.555556, 0.555556}
```

What is the Legendre Polynomial

```
Plot[{P[1, x], P[2, x], P[3, x], P[4, x]}, {x, -1, 1},
 PlotLegends → Placed[{"n=1", "n=2", "n=3", "n=4"}, Above]]
```



The integral will work between the interval ∈ [-1, 1]. In 2-D things are more complex, but tables are available, and we will not bother further with analytical equations. In 1-D the tables are as follows:

Table 4.1 Position of Gauss points and corresponding weights.

n_{gp}	Location, ξ_i	Weights, W_i
1	0.0	2.0
2	$\pm 1/\sqrt{3} = \pm 0.5773502692$	1.0
3	± 0.7745966692 0.0	0.555 555 5556 0.888 888 8889
4	± 0.8611363116 ± 0.3399810436	0.347 854 8451 0.652 145 1549
5	± 0.9061798459 ± 0.5384693101 0.0	0.236 926 8851 0.478 628 6705 0.568 888 8889
6	± 0.9324695142 ± 0.6612093865 ± 0.2386191861	0.171 324 4924 0.360 761 5730 0.467 913 9346

Finite Element Analysis in 1D

Finite element analysis has three steps:

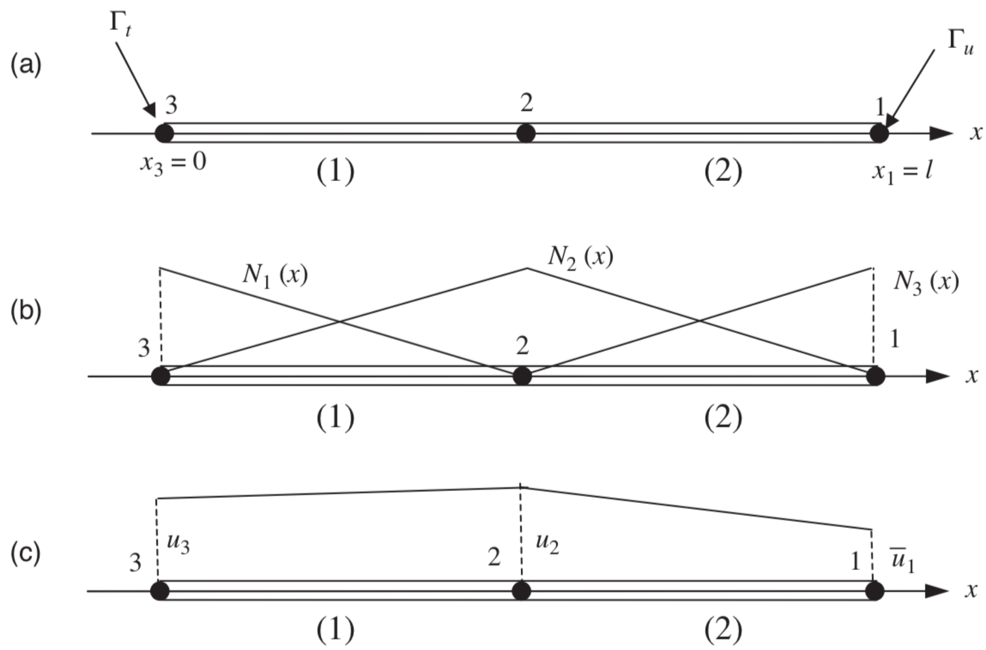
1. pre-processing in which the mesh is constructed
2. formulation of the discrete finite element equations
3. solving the discrete equations.
4. post-processing, where the solution is displayed and various variables that do not emanate

directly from the solution are calculated.

Quite straight forward for 1D problems. Very important in 2D and 3D problems. Consider the 1D problem of elasticity. The weak-form is:

$$\int_0^l \left(\frac{\mathrm{d}w}{\mathrm{d}x}\right)^{\mathrm{T}} AE\left(\frac{\mathrm{d}u}{\mathrm{d}x}\right)\mathrm{d}x - \int_0^l w^{\mathrm{T}}\, b\,\mathrm{d}x - (w^{\mathrm{T}}\bar{t}A)\Big|_{x=0} = 0 \qquad \forall w(x) \text{ with } w(l) = 0.$$

The transpose *T* is not important, but will be required when we discuss the discretisation in FEA.



(a) two-element mesh, (b) global shape functions and (c) and example of a trial solution that satisfies the essential boundary conditions.

The discretization is now provided in the following form:

$u(x) \approx u^h(x) = \boldsymbol{N}(x)\,\boldsymbol{u} = u_1 N_1(x) + u_2 N_2(x) + u_3 N_3(x)$ and

$w(x) \approx w^h(x) = \boldsymbol{N}(x)\,\boldsymbol{w} = w_1 N_1(x) + w_2 N_2(x) + w_3 N_3(x)$.

The trial functions should be chosen such that $u_1 = \bar{u}_1$ and $w_1 = 0$.

For every element:

$$\mathbf{w}^e = \mathbf{L}^e\mathbf{w}, \qquad \mathbf{d}^e = \mathbf{L}^e\mathbf{d}.$$

The weak form is now written in the form:

$$\sum_{e=1}^{n_{\mathrm{el}}}\left\{\int_{x_1^e}^{x_2^e}\left(\frac{\mathrm{d}w^e}{\mathrm{d}x}\right)^{\mathrm{T}} A^e E^e\left(\frac{\mathrm{d}u^e}{\mathrm{d}x}\right)\mathrm{d}x - \int_{x_1^e}^{x_2^e} w^{e\mathrm{T}}\, b\,\mathrm{d}x - (w^{e\mathrm{T}}A^e\bar{t})\Big|_{x=0}\right\} = 0$$

where *e* corresponds to element number. For each element we can write

$$u^e(x) = \mathbf{N}^e \mathbf{d}^e, \qquad \frac{\mathrm{d}u^e}{\mathrm{d}x} = \mathbf{B}^e \mathbf{d}^e,$$
$$w^{e\mathrm{T}} = \mathbf{w}^{e\mathrm{T}} \mathbf{N}^{e\mathrm{T}}, \qquad \left(\frac{\mathrm{d}w^e}{\mathrm{d}x}\right)^{\mathrm{T}} = \mathbf{w}^{e\mathrm{T}} \mathbf{B}^{e\mathrm{T}},$$

Note, that the global shape function **N** and the element-level shape function $\boldsymbol{N}^e$ are the same! Hence we now get:

$$\sum_{e=1}^{n_{el}} \mathbf{w}^{e\mathrm{T}} \left\{ \underbrace{\int_{x_1^e}^{x_2^e} \mathbf{B}^{e\mathrm{T}} A^e E^e\, \mathbf{B}^e \mathrm{d}x \mathbf{d}^e}_{\mathbf{K}^e} - \underbrace{\int_{x_1^e}^{x_2^e} \mathbf{N}^{e\mathrm{T}} b\, \mathrm{d}x}_{\mathbf{f}_{\Omega^e}} - (\underbrace{\mathbf{N}^{e\mathrm{T}} A^e \bar{t}}_{\mathbf{f}_{\Gamma^e}})_{x=0} \right\} = 0.$$

The following is what we now interpret:

(i) the *element stiffness matrix*

$$\boxed{\mathbf{K}^e = \int_{x_1^e}^{x_2^e} \mathbf{B}^{e\mathrm{T}} A^e E^e\, \mathbf{B}^e\, \mathrm{d}x = \int_{\Omega^e} \mathbf{B}^{e\mathrm{T}} A^e E^e\, \mathbf{B}^e\, \mathrm{d}x;}$$

(ii) the *element external force matrix*

$$\boxed{\mathbf{f}^e = \int_{x_1^e}^{x_2^e} \mathbf{N}^{e\mathrm{T}} b\, \mathrm{d}x + (\mathbf{N}^{e\mathrm{T}} A^e \bar{t})_{x=0} = \underbrace{\int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} b\, \mathrm{d}x}_{\mathbf{f}_{\Omega}^e} + \underbrace{(\mathbf{N}^{e\mathrm{T}} A^e \bar{t})\Big|_{\Gamma_t^e}}_{\mathbf{f}_{\Gamma}^e}}$$

In these equations, Γ_t^e is the portion of the element boundary on the natural boundary and f_Ω^e and f_Γ^e are the element external body and boundary forces matrices, respectively. This is very similar to what is done in the stiffness approach for structures.

From all the equations, we obtain:

$$\mathbf{w}^{\mathrm{T}} \left(\sum_{e=1}^{n_{el}} \mathbf{L}^{e\mathrm{T}} \mathbf{K}^e \mathbf{L}^e \mathbf{d} - \sum_{e=1}^{n_{el}} \mathbf{L}^{e\mathrm{T}} \mathbf{f}^e \right) = 0.$$

Here, the global stiffness matrix

$$\boxed{\mathbf{K} = \sum_{e=1}^{n_{el}} \mathbf{L}^{e\mathrm{T}} \mathbf{K}^e \mathbf{L}^e.}$$

The system matrix for the differential equation is *assembled by exactly the same operations as for the discrete systems: matrix scatter and add,* which is also equivalent to the directly assembly.

$$\boxed{\mathbf{f} = \sum_{e=1}^{n_{\rm el}} \mathbf{L}^{e\mathrm{T}} \mathbf{f}^e.}$$

This is the *column matrix assembly operation*. It consists of a *columns matrix scatter and add* and easier to learn that matrix assembly. Look at the Second Chapter of Fish and Belytschko.

Using these equations we get the following equation

$$\mathbf{w}^{\mathrm{T}}(\mathbf{K}\,\mathbf{d} - \mathbf{f}) = 0 \qquad \forall \mathbf{w} \text{ except } w_1 = w(l) = 0,$$

If now we say that:

$\boldsymbol{K}\,\boldsymbol{d} - \boldsymbol{f} = \boldsymbol{r}$, then

$\boldsymbol{w}^{\boldsymbol{T}}\,\boldsymbol{r} = 0$, which is true for $w_1 = 0$. This implies that:

$w_2\,r_2 + w_3\,r_3 = 0$, for any w_2 and w_3.

This can only happen when r_1 is something unknown but $r_2 = r_3 = 0$. This now simply implies that:

$$\mathbf{r} = \begin{bmatrix} r_1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} \bar{u}_1 \\ u_2 \\ u_3 \end{bmatrix} - \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix}.$$

This upon rearrangement gives

$$\begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} \bar{u}_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} f_1 + r_1 \\ f_2 \\ f_3 \end{bmatrix}.$$

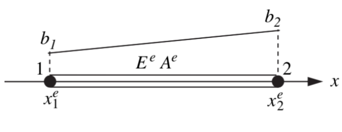
We can use matrix partition methods to provide us

$$\begin{bmatrix} K_{22} & K_{23} \\ K_{32} & K_{33} \end{bmatrix} \begin{bmatrix} u_2 \\ u_3 \end{bmatrix} = \begin{Bmatrix} f_2 - K_{21}\bar{u}_1 \\ f_3 - K_{31}\bar{u}_1 \end{Bmatrix},$$

and the reaction r_1 will be obtained as

$$r_1 = f_1 - [K_{11} \quad K_{12} \quad K_{13}] \begin{bmatrix} \bar{u}_1 \\ u_2 \\ u_3 \end{bmatrix}.$$

We now obtain the stiffness matrix for two noded element.



Two-node element with linear distribution of body force.

In this case,

$$\mathbf{N}^e = \left[\frac{x - x_2^e}{x_1^e - x_2^e} \quad \frac{x - x_1^e}{x_2^e - x_1^e} \right] = \frac{1}{l^e}\left[(x_2^e - x) \quad (x - x_1^e) \right],$$

$$\mathbf{B}^e = \frac{\mathrm{d}}{\mathrm{d}x}\mathbf{N}^e = \left[-\frac{1}{l^e} \quad \frac{1}{l^e} \right] = \frac{1}{l^e}\left[-1 \ 1 \right].$$

For this the stiffness matrix becomes,

$$\mathbf{K}^e = \int_{x_1^e}^{x_2^e} \mathbf{B}^{e\mathrm{T}} A^e E^e \mathbf{B}^e \,\mathrm{d}x = \int_{x_1^e}^{x_2^e} \underbrace{\frac{1}{l^e}\begin{bmatrix} -1 \\ 1 \end{bmatrix}}_{\mathbf{B}^{eT}} A^e E^e \underbrace{\frac{1}{l^e}[-1 \;\; 1]}_{\mathbf{B}^e} \,\mathrm{d}x = \frac{A^e E^e}{(l^e)^2}\begin{bmatrix} -1 \\ 1 \end{bmatrix}[-1 \;\; 1]\int_{x_1^e}^{x_2^e} \mathrm{d}x$$

$$= \frac{A^e E^e}{(l^e)^2}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}\left(\underbrace{x_2^e - x_1^e}_{l^e}\right),$$

$$\mathbf{K}^e = \frac{A^e E^e}{l^e}\begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

which is the same that we will get for a truss element! However, this simple form will not work in 2D and 3D. The body forces acting on the bar are given as:

$$\mathbf{f}_\Omega^e = \int_{x_1^e}^{x_2^e} \mathbf{N}^{e\mathrm{T}} b(x)\,\mathrm{d}x.$$

Since the body force distribution is linear, it can be expressed in terms of the same shape function as

$$b(x) = \mathbf{N}^e \mathbf{b}, \qquad \mathbf{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

$$\mathbf{f}_\Omega^e = \int_{x_1^e}^{x_2^e} \mathbf{N}^{e\mathrm{T}}\mathbf{N}^e \,\mathrm{d}x\,\mathbf{b} = \frac{1}{(l^e)^2}\int_{x_1^e}^{x_2^e} \begin{bmatrix} (x_2^e - x)^2 & (x_2^e - x)(x - x_1^e) \\ (x_2^e - x)(x - x_1^e) & (x - x_1^e)^2 \end{bmatrix} \mathrm{d}x\,\mathbf{b}$$

$$= \frac{l^e}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

In the special case, when $b_1 = b_2$, the half the body force is transferred to each node as expected. What does one do when the body forces are not linear? One could approximate with shape-functions of the order that could be used.

Matrices	Elasticity	Diffusion	Heat conduction
K	Stiffness	Diffusivity	Conductance
f	Force	Flux	Flux
d	Displacement	Concentration	Temperature

Table: Terminology for finite element matrices

Application to heat conduction and diffusion problems.

The differential equation is:

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}x}\left(Ak\frac{\mathrm{d}T}{\mathrm{d}x}\right)+s=0 \quad \text{on} \quad 0<x<l,\\
&-q=k\frac{\mathrm{d}T}{\mathrm{d}x}=\bar{q} \quad \text{on} \quad x=0,\\
&T=\overline{T} \quad \text{on} \quad x=l.
\end{aligned}
$$

Note that the source term s is positive is the heat flow is into the system and the boundary term q is positive if the heat flows **out** of the bar. In the current case, $q(x = 0) = -\bar{q}$, i.e., we are assuming that the heat flux is into the bar.

The equivalence between the heat conduction and the elasticity problem is:

$k \Longleftrightarrow E$

$q \Longleftrightarrow -p$

$s \Longleftrightarrow b$

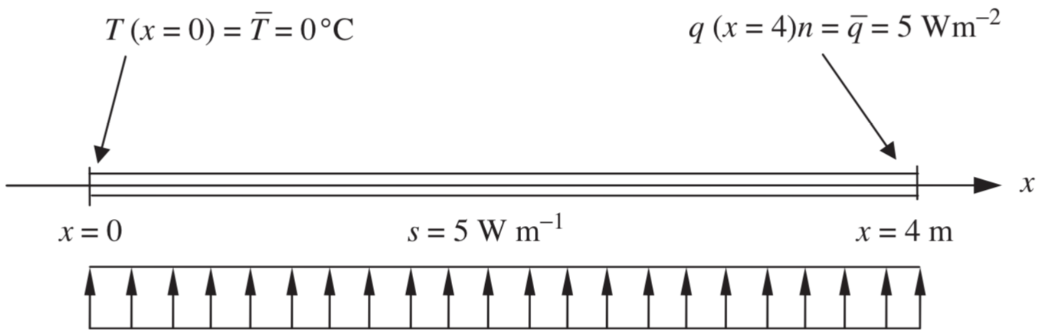
In the finite element formulation the equations would look like this:

$$
\begin{aligned}
\mathbf{K}^e &= \int_{\Omega^e} \mathbf{B}^{e\mathrm{T}} A^e \kappa^e \mathbf{B}^e \,\mathrm{d}x,\\
\mathbf{f}^e &= \underbrace{\int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} f \,\mathrm{d}x}_{\mathbf{f}^e_\Omega} + \underbrace{(\mathbf{N}^{e\mathrm{T}} A^e \bar{\Phi})\Big|_{\Gamma^e_\Phi}}_{\mathbf{f}^e_\Gamma}
\end{aligned}
$$

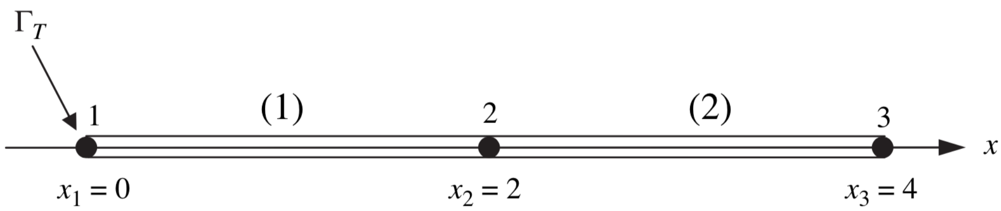
The quantity $\bar{\Phi} = -\bar{q}$ in the current case (the flux $\bar{q}$ is positive when flowing outwards.)

Problem

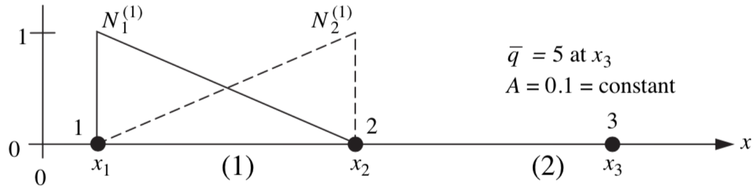
Consider a bar with a uniformly distributed heat source of $s = 5\ \mathrm{W\ m^{-1}}$. The bar has a uniform cross-sectional area of $A = 0.1\ \mathrm{m}^2$ and thermal conductivity $k = 2\ \mathrm{W\ {}^\circ C^{-1}\ m^{-1}}$. The length of the bar is 4 m. The boundary conditions are $T(0) = 0\,^\circ\mathrm{C}$ and $\bar{q}(x = 4) = 5\ \mathrm{W\ m^{-2}}$ as shown in Figure 5.3. Divide the problem domain into two linear temperature two-node elements and solve it by the FEM.



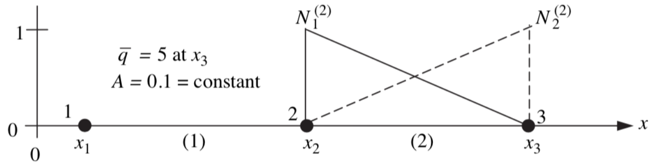
The finite element mesh is as shown below:



The shape functions are as below:



Shape functions for element-1



Shape functions for element-2

Element conductance Matrix

$$\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^{e\mathrm{T}} A^e k^e \mathbf{B}^e \,\mathrm{d}x = \frac{A^e k^e}{l^e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

The element stiffness is replaced with conductivity.

The conductance matrix for element-1 is:

$$x_1^{(1)} = 0, \qquad x_2^{(1)} = 2, \qquad l^{(1)} = 2, \qquad (Ak)^{(1)} = 0.2,$$

$$\mathbf{K}^{(1)} = \frac{0.2}{2} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix},$$

and for element-2

$$\mathbf{K}^{(2)} = \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix}.$$

Global conductance matrix

$$\mathbf{K} = \sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{K}^e \mathbf{L}^e = \mathbf{L}^{(1)\mathrm{T}} \mathbf{K}^{(1)} \mathbf{L}^{(1)} + \mathbf{L}^{(2)\mathrm{T}} \mathbf{K}^{(2)} \mathbf{L}^{(2)}.$$

The *gather* operators for the two elements are

$$\mathbf{d}^{(1)} = \begin{bmatrix} T_1^{(1)} \\ T_2^{(1)} \end{bmatrix} = \begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \mathbf{L}^{(1)} \mathbf{d},$$

$$\mathbf{d}^{(2)} = \begin{bmatrix} T_1^{(2)} \\ T_2^{(2)} \end{bmatrix} = \begin{bmatrix} T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \mathbf{L}^{(2)} \mathbf{d}.$$

The scatter of the conductance matrices gives

$$\tilde{\mathbf{K}}^{(1)} = \mathbf{L}^{(1)\mathrm{T}}\mathbf{K}^{(1)}\mathbf{L}^{(1)} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0.1 & -0.1 & 0 \\ -0.1 & 0.1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{\mathbf{K}}^{(2)} = \mathbf{L}^{(2)\mathrm{T}}\mathbf{K}^{(2)}\mathbf{L}^{(2)} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.1 & -0.1 \\ 0 & -0.1 & 0.1 \end{bmatrix}$$

The total *stiffness* is obtained by adding the scattered element stiffness mattices.

$$\mathbf{K} = \tilde{\mathbf{K}}^{(1)} + \tilde{\mathbf{K}}^{(2)} = \begin{bmatrix} 0.1 & -0.1 & 0 \\ -0.1 & 0.2 & -0.1 \\ 0 & -0.1 & 0.1 \end{bmatrix}.$$

However, though this is conceptually easier to discuss, in practice the stiffness matrices are not add like these, but assembled.

$$\mathbf{K}^{(1)} = \begin{array}{cc} \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix} & \begin{array}{c} [1] \\ [2] \end{array} \\ \begin{array}{cc} [1] & \quad [2] \end{array} & \end{array} \qquad \mathbf{K}^{(2)} = \begin{array}{cc} \begin{bmatrix} 0.1 & -0.1 \\ -0.1 & 0.1 \end{bmatrix} & \begin{array}{c} [2] \\ [3] \end{array} \\ \begin{array}{cc} [2] & \quad [3] \end{array} & \end{array}$$

The resulting global conductance matrix is the same as above.

Boundary flux matrix

The element boundary flux are calculated as per the following expression

$$\mathbf{f}_\Gamma^e = -(\mathbf{N}^{e\mathrm{T}} A^e \cdot \bar{q})\Big|_{\Gamma_q^e} = -\mathbf{N}^{e\mathrm{T}}(x_3) \times 0.1 \times 5 = -0.5\,\mathbf{N}^{e\mathrm{T}}(x_3).$$

Shape functions for element 1 vanish on boundary 1, i.e. Γ_q. Only the shape functions that are non-zero at the natural boundary survive need to be considered. More explicitly,

$$\mathbf{f}_\Gamma^{(1)} = -0.5 \begin{bmatrix} N_1^{(1)}(x_3) \\ N_2^{(1)}(x_3) \end{bmatrix} = -0.5 \begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{array}{c} [1] \\ [2] \end{array}$$

$$\mathbf{f}_\Gamma^{(2)} = -0.5 \begin{bmatrix} N_1^{(2)}(x_3) \\ N_2^{(2)}(x_3) \end{bmatrix} = -0.5 \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ -0.5 \end{bmatrix} \begin{array}{c} [2] \\ [3] \end{array}$$

The scatter process then gives the global boundary flux matrix

$$\mathbf{f}_\Gamma = \sum_{e=1}^{2} \mathbf{L}^{e\mathrm{T}} \mathbf{f}_\Gamma^e,$$

$$\mathbf{f}_\Gamma = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -0.5 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -0.5 \end{bmatrix} \begin{array}{c} [1] \\ [2] \\ [3] \end{array}.$$

This result is exactly the same as assigning ($-A\,\overline{q}$) directly on the boundary node. In practise this is how it is done.

Source flux matrix

The element source flux matrix is now simply obtained as:

$$\mathbf{f}_\Omega^e = \int_{x_1^e}^{x_{n_{\text{en}}}^e} \mathbf{N}^{e\mathrm{T}} s \,\mathrm{d}x = \frac{l^e}{6}\begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}\begin{bmatrix} s_1 \\ s_2 \end{bmatrix}.$$

Since $s_1 = s_2$, the above reduces to

$$\mathbf{f}_\Omega^e = \frac{l^e s}{2}\begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

Since the length of both elements is the same $l^{(1)} = l^{(2)} = 2$ and $s = 5$, which gives

$$\mathbf{f}_\Omega^{(1)} = \mathbf{f}_\Omega^{(2)} = \begin{bmatrix} 5 \\ 5 \end{bmatrix}.$$

The assemble source flux matrix now becomes:

$$\mathbf{f}_\Omega = \sum_{e=1}^{2} \mathbf{L}^{e\mathrm{T}} \mathbf{f}_\Omega^e = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} 5 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 5 \\ 5 \end{bmatrix} = \begin{bmatrix} 5 \\ 10 \\ 5 \end{bmatrix}.$$

In practice, a direct assembly is used:

$$\begin{matrix} \mathbf{f}_\Omega^{(1)} = \begin{bmatrix} 5 \\ 5 \end{bmatrix} \begin{matrix} [1] \\ [2] \end{matrix} \\ \mathbf{f}_\Omega^{(2)} = \begin{bmatrix} 5 \\ 5 \end{bmatrix} \begin{matrix} [2] \\ [3] \end{matrix} \end{matrix} \quad \Rightarrow \quad \mathbf{f}_\Omega = \begin{bmatrix} 5 \\ 5+5 \\ 5 \end{bmatrix} \begin{matrix} [1] \\ [2] \\ [3] \end{matrix}.$$

Partition and solution

The global system of equations is given by

$$\begin{bmatrix} 0.1 & -0.1 & 0 \\ -0.1 & 0.2 & -0.1 \\ 0 & -0.1 & 0.1 \end{bmatrix}\begin{bmatrix} 0 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 5 \\ 10 \\ 5 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -0.5 \end{bmatrix} + \begin{bmatrix} r_1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} r_1 + 5 \\ 10 \\ 4.5 \end{bmatrix}.$$

Since node 1 is on the essential boundary, we partition after the first row, which gives

$$\begin{bmatrix} 0.2 & -0.1 \\ -0.1 & 0.1 \end{bmatrix}\begin{bmatrix} T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 10 \\ 4.5 \end{bmatrix} \quad \Rightarrow \quad \begin{bmatrix} T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 145 \\ 190 \end{bmatrix}.$$

Postprocessing

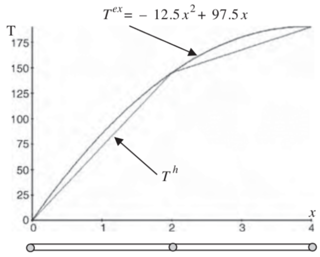
The temperature gradient is given as:

$$\frac{\mathrm{d}T^{(1)}}{\mathrm{d}x} = \mathbf{B}^{(1)}\mathbf{L}^{(1)}\mathbf{d} = \frac{1}{2}[-1 \ \ 1]\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 \\ 145 \\ 190 \end{bmatrix} = 72.5,$$

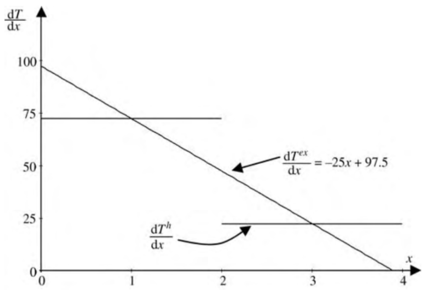
$$\frac{\mathrm{d}T^{(2)}}{\mathrm{d}x} = \mathbf{B}^{(2)}\mathbf{L}^{(2)}\mathbf{d} = \frac{1}{2}[-1 \ \ 1]\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} 0 \\ 145 \\ 190 \end{bmatrix} = 22.5.$$

The temperature gradient is piecewise constant as expected.

Comparision with the exact solution



Comparision of the the exact and finite element solutions of the temperature. Note that for 1D FEA the nodal values are exactly the same as the actual solution. **Note:** this does not mean, however, that the error of the solution is zero.



Comparison of the exact and finite element solutions of temperature gradient.

The strong form of the differential equation will not be satisfied, especially at the transition between the two elements, where due to the C^o continuity the derivative is infinite (or does not exist.)

Convergence to the actual solution

Definition of error.

To define error, we need to first define now concept of “distance” or “norm”. For any vector $\vec{a}$, sometimes called the norm of the vector and denoted by $||\vec{a}||$,

$||\vec{a}|| = (\sum_{i=1}^{n} a_i^2)^{\frac{1}{2}}$,

where $n$is the number of components of the vector. For any function, the norm of the function is defined by:

$$\| f(x) \|_{L_2} = \left(\int_{x_1}^{x_2} f^2(x)\, \mathrm{d}x \right)^{\frac{1}{2}},$$

where $[x_1, \ x_2]$ is the interval over the function is defined. The error norm in the finite element solution is:

$$\| \mathrm{e} \|_{L_2} = \| u^{\mathrm{ex}}(x) - u^h(x) \| = \left(\int_{x_1}^{x_2} (u^{\mathrm{ex}}(x) - u^h(x))^2\, \mathrm{d}x \right)^{\frac{1}{2}},$$

The normalized error is given by

$$\bar{\mathrm{e}}_{L_2} = \frac{\| u^{\mathrm{ex}}(x) - u^h(x) \|_{L_2}}{\| u^{\mathrm{ex}}(x) \|_{L_2}} = \frac{\left(\int_{x_1}^{x_2} (u^{\mathrm{ex}}(x) - u^h(x))^2 \, \mathrm{d}x\right)^{\frac{1}{2}}}{\left(\int_{x_1}^{x_2} (u^{\mathrm{ex}}(x))^2 \, \mathrm{d}x\right)^{\frac{1}{2}}}.$$

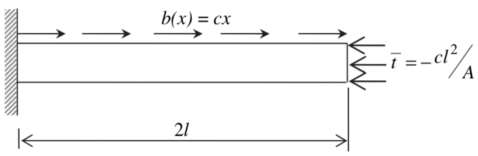
A more important quantity is the error in the derivative:

$$\| \mathrm{e} \|_{\mathrm{en}} = \| u^{\mathrm{ex}}(x) - u^h(x) \|_{\mathrm{en}} = \left(\frac{1}{2}\int_{x_1}^{x_2} E(\varepsilon^{\mathrm{ex}}(x) - \varepsilon^h(x))^2 \, \mathrm{d}x\right)^{\frac{1}{2}}.$$

This is also called as the *energy* norm. The error fraction percentage is:

$$\bar{\mathrm{e}}_{\mathrm{en}} = \frac{\| u^{\mathrm{ex}}(x) - u^h(x) \|_{\mathrm{en}}}{\| u^{\mathrm{ex}}(x) \|_{\mathrm{en}}} = \frac{\left(\frac{1}{2}\int_{x_1}^{x_2} E(\varepsilon^{\mathrm{ex}}(x) - \varepsilon^h(x))^2 \, \mathrm{d}x\right)^{\frac{1}{2}}}{\left(\frac{1}{2}\int_{x_1}^{x_2} E(\varepsilon^{\mathrm{ex}}(x))^2 \, \mathrm{d}x\right)^{\frac{1}{2}}}.$$

Consider the problem below:



Consider a bar of length $2\,l$, cross-section area A and Young's modulus E. The bar is fized at $x = 0$ subjected to linear force $c\,x$ and applied traction $\bar{t} = -c\,l^2/A$ at $x = 2\,l$as shown in the figure above. The strong form is:

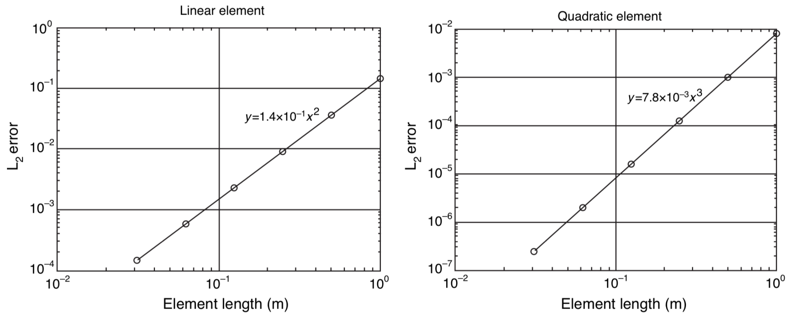
$$\frac{\mathrm{d}}{\mathrm{d}x}\left(AE\frac{\mathrm{d}u}{\mathrm{d}x}\right) + cx = 0,$$
$$u(0) = 0,$$
$$\bar{t} = E\frac{\mathrm{d}u}{\mathrm{d}x}n\bigg|_{x=2l} = -\frac{cl^2}{A}.$$

In the equation above $n = 1$.

The solution for the above problem can be obtained in the closed form.

$$u^{ex}(x) = \frac{c}{AE}\left(-\frac{x^3}{6} + l^2 x\right),$$
$$\varepsilon^{ex}(x) = \frac{du}{dx} = \frac{c}{AE}\left(-\frac{x^2}{2} + l^2\right).$$

Consider the following parameters:
$E = 10^4\, N\, m^{-2},\ A = 1\, m^2,\ c = 1\, N\, m^{-2}$ and $l = 1\, m$.



L_2 norm of error for linear (left) and quadratic (right) finite element meshes.

This is a log-log plot and the fit to the error can be expressed as:

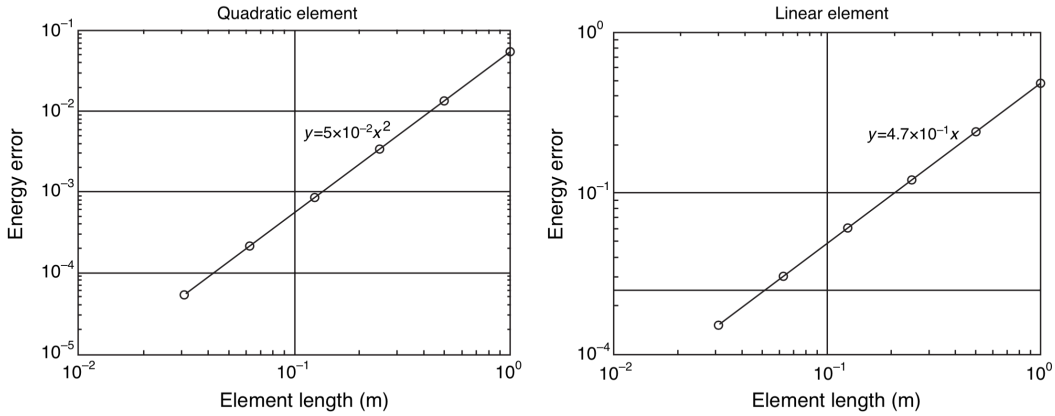
$$\log(\|e\|_{L_2}) = C + \alpha \log h,$$

where C is an arbitrary constant, the y –intercept of the curve. The the power of both sides gives:

$$\|e\|_{L_2} = Ch^{\alpha}.$$

For a finite element that containts the complete polynomial of order p, the convergence can be showed to be:

$$\|e\|_{L_2} = Ch^{p+1}.$$



Energy norm of error for linear (left) and quadratic (right) finite element meshes.

We will proceed further with FEniCS.

Weak form for multi-dimensional scalar field

First consider 2D problems. The conversion from 2D to 3D is extremely straight forward. Though the FEA formulation for 2D is very much the same as in 1D, the ideas required to obtain the weak form is somewhat different and involves a more generalised version of integration by parts and called as

Green’s theorem.

In 2D, the analogy between 1D is as follows

$\frac{d}{dx} \Longleftrightarrow \nabla = \left(\boldsymbol{i}\,\frac{\partial}{\partial x} + \boldsymbol{j}\,\frac{\partial}{\partial y}\right)$

Hence the full derivative has to be replaced with the appropriate partial derivative. Taking the example of heat equation, the heat flux now is a vector

$\boldsymbol{q} = q_x \boldsymbol{i} + q_y \boldsymbol{j}$.

And could be written interchangeably in the matrix notation:

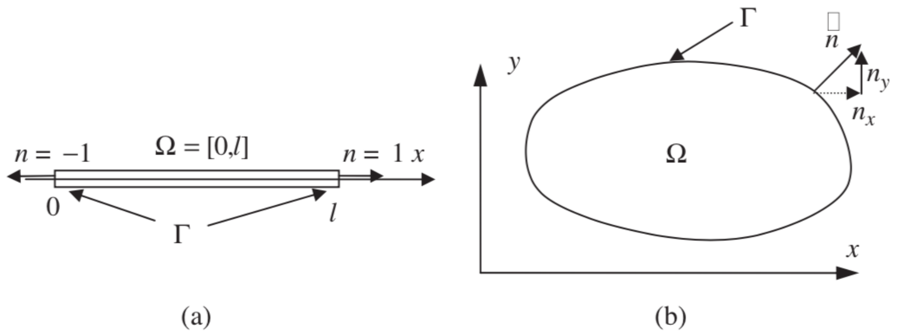
$\boldsymbol{q} = \begin{pmatrix} q_x \\ q_y \end{pmatrix}$ and $\boldsymbol{q}^T = \begin{pmatrix} q_x & q_y \end{pmatrix}$. Note, how the row and column vector are written.

The gradient vector can also be written in terms of column vector

$\nabla = \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}$ and similarly for ∇^T.

Note that the row and column vectors obtained earlier are not **real vectors**. However, for all practical purposes, we can replace vectors in terms of their components, and hence column or row matrices.

All the other notations are similar to what we have done in the class.



(a) One-dimensional domain and (b) Two-dimensional domain.

The 2D domain could be quite complex with multiple holes (multiply connected) or with corners.

The unit normal vector to the domain is given as:

$\vec{n} = n_x\,\vec{i} + n_y\,\vec{j}$.

Here, $\vec{n}$ is of unit magnitude, i.e., $n_x^2 + n_y^2 = 1$. What is the equivalent of integration by parts in 2D.

For any C^0 integrable function in one-dimensional domain Ω, we have:

$$\int_\Omega \frac{d\theta(x)}{dx}\,dx = (\theta n)|_\Gamma.$$

The boundaries correspond to $x = 0,\ l$. In 2D (or multi-dimension) the equation becomes:

$$\boxed{\begin{array}{l}\text{If } \theta(x,y) \in C^0 \text{ and integrable, then} \\ \displaystyle\int_\Omega \vec{\nabla}\theta\,d\Omega = \oint_\Gamma \theta\vec{n}\,d\Gamma \quad \text{or} \quad \int_\Omega \boldsymbol{\nabla}\theta\,d\Omega = \oint_\Gamma \theta\mathbf{n}\,d\Gamma.\end{array}}$$

The 1D formula is a special case of this. This equation can be used to derive the Gauss divergence theorem.

$$\boxed{\int_\Omega \vec{\nabla}\cdot\vec{q}\,\mathrm{d}\Omega = \oint_\Gamma \vec{q}\cdot\vec{n}\,\mathrm{d}\Gamma \quad \text{or} \quad \int_\Omega \boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q}\,\mathrm{d}\Omega = \oint_\Gamma \mathbf{q}^{\mathrm{T}}\mathbf{n}\,\mathrm{d}\Gamma.}$$

The Green’s theorem can be written in terms of two vector equations:

$$\text{(a)} \quad \int_\Omega \frac{\partial\theta}{\partial x}\mathrm{d}\Omega = \int_\Gamma \theta n_x\,\mathrm{d}\Gamma, \qquad \text{(b)} \quad \int_\Omega \frac{\partial\theta}{\partial y}\,\mathrm{d}\Omega = \int_\Gamma \theta n_y\,\mathrm{d}\Gamma.$$

Now, writing $\theta = q_x$and $\theta = q_y$in (a) and (b) above and adding them we get

$$\int_\Omega \left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y}\right)\mathrm{d}\Omega = \oint_\Gamma (q_x n_x + q_y n_y)\,\mathrm{d}\Gamma \quad \text{or} \quad \int_\Omega \vec{\nabla}\cdot\vec{q}\,\mathrm{d}\Omega = \oint_\Gamma \vec{q}\cdot\vec{n}\,\mathrm{d}\Gamma,$$

which is the Gauss divergence theorem stated above. Below, we have the ***Green’s formula***:

$$\int_\Omega w\vec{\nabla}\cdot\vec{q}\,\mathrm{d}\Omega = \oint_\Gamma w\vec{q}\cdot\vec{n}\,\mathrm{d}\Gamma - \int_\Omega \vec{\nabla}w\cdot\vec{q}\,\mathrm{d}\Omega \quad \text{or} \quad \int_\Omega w\boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q}\,\mathrm{d}\Omega = \oint_\Gamma w\,\mathbf{q}^{\mathrm{T}}\mathbf{n}\,\mathrm{d}\Gamma - \int_\Omega (\boldsymbol{\nabla}w)^{\mathrm{T}}\mathbf{q}\,\mathrm{d}\Omega.$$

This formula is obtained by noting that

$$\begin{aligned}\vec{\nabla}\cdot(w\vec{q}) &= \frac{\partial}{\partial x}(wq_x) + \frac{\partial}{\partial y}(wq_y) = \frac{\partial w}{\partial x}q_x + w\frac{\partial q_x}{\partial x} + \frac{\partial w}{\partial y}q_y + w\frac{\partial q_y}{\partial y} \\ &= w\underbrace{\left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y}\right)}_{\vec{\nabla}\cdot\vec{q}} + \underbrace{\left(\frac{\partial w}{\partial x}q_x + \frac{\partial w}{\partial y}q_y\right)}_{\vec{\nabla}w\cdot\vec{q}} = w\vec{\nabla}\cdot\vec{q} + \vec{\nabla}w\cdot\vec{q}.\end{aligned}$$

Integrating this expression over the domain Ω we get

$$\int_\Omega \vec{\nabla}\cdot(w\vec{q})\,\mathrm{d}\Omega = \int_\Omega w\vec{\nabla}\cdot\vec{q}\,\mathrm{d}\Omega + \int_\Omega \vec{\nabla}w\cdot\vec{q}\,\mathrm{d}\Omega.$$

We then apply the divergence theorem to get:

$$\boxed{\int_\Omega w\vec{\nabla}\cdot\vec{q}\,\mathrm{d}\Omega = \oint_\Gamma w\vec{q}\cdot\vec{n}\,\mathrm{d}\Gamma - \int_\Omega \vec{\nabla}w\cdot\vec{q}\,\mathrm{d}\Omega.}$$

This formula is extremely important to obtain the weak form from the strong form for the steady-state heat equation. For a rectangular domain of dimension $l \times 1$ with one-dimensional heat flow, where

$\vec{q} = q_x\,\vec{i}$ and $\hat{n} = n\,\hat{i}$, where $n(0) = -n(l) = 1$.

$$\int_\Omega w\frac{\partial q_x}{\partial x}\mathrm{d}\Omega = \oint_\Gamma q_x w n\,\mathrm{d}\Gamma - \int_\Omega \frac{\partial w}{\partial x}q_x\,\mathrm{d}\Omega.$$

If both w and q are only functions of x then

$$\int_0^l w \frac{\partial q_x}{\partial x}\,\mathrm{d}x = (q_x w)_{x=l} - (q_x w)_{x=0} - \int_0^l q_x \frac{\partial w}{\partial x}\,\mathrm{d}x.$$

This is pretty much the same as the expression for the 1D case.

Problem:

Given a rectangular domain as shown in Figure 6.2. Consider a scalar function $\theta = x^2 + 2y^2$. Let $\vec{q}$ be the gradient of θ defined as $\vec{q} = \vec{\nabla}\theta$. Contour lines are lines along which a function is constant.

(a) Find the normal to the contour line of θ passing through the point $x = y = 0.5$.
(b) Verify the divergence theorem for $\vec{q}$.

The function and the gradient are obtained as follows:

```
θ[x_, y_] := x^2 + 2 y^2
gradθ[x_, y_] = {∂_x θ[x, y], ∂_y θ[x, y]}
```

```
{2 x, 4 y}
```

To solve part (a), i.e., the contour of $\theta(x, y)$ which points through $x = y = 0.5$. The value of θ is:

```
θ[0.5, 0.5]
```

```
0.75
```

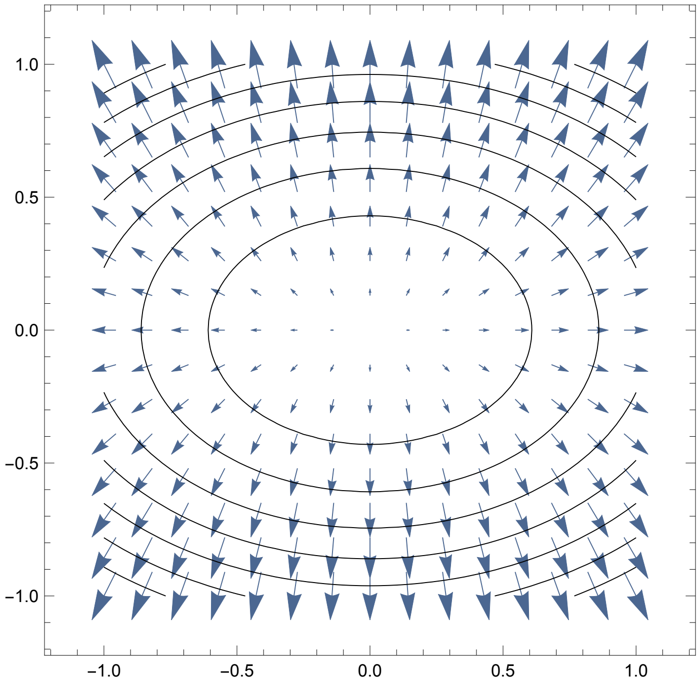
The contour corresponds to $x^2 + 2y^2 = 0.75$. The normal is the unit vector passing through 0.5, 0.5.

```
gradθ[1/2, 1/2] / Norm[gradθ[1/2, 1/2]]
```

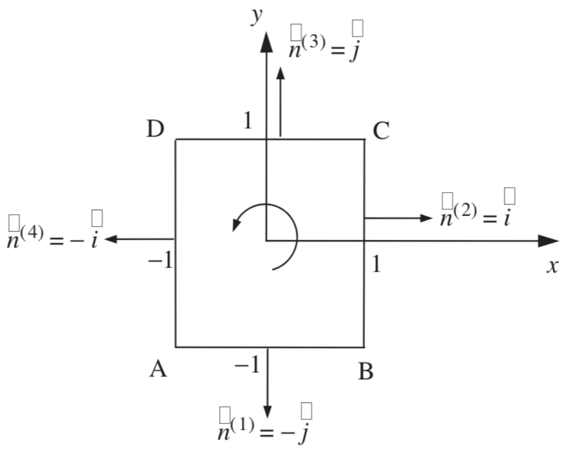
$\left\{\frac{1}{\sqrt{5}}, \frac{2}{\sqrt{5}}\right\}$

To make the full contour plot along with vector field:

```
A = VectorPlot[gradθ[x, y], {x, -1, 1}, {y, -1, 1}];
B = ContourPlot[θ[x, y], {x, -1, 1}, {y, -1, 1},
   ContourShading → None, ContourLabels → False, Contours → 7];
Show[
 A,
 B]
```



To address part (b), we use the following domain



Note that the empty box above many symbols should have had →.

$$\vec{\nabla} \cdot \vec{q} = \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = 2 + 4 = 6.$$

Integration of this over the domain we get

$$\int_{\Omega} \vec{\nabla} \cdot \vec{q}\, \mathrm{d}\Omega = \int_{-1}^{1} \left(\int_{-1}^{1} 6\, \mathrm{d}y \right) \mathrm{d}x = 24.$$

Evaluating the boundary integral counter-clockwise:

Evaluating the boundary integral counterclockwise gives

$$\oint_{\Gamma} \vec{q} \cdot \vec{n}\, \mathrm{d}\Gamma = \int_{AB} (-4y) \underbrace{\mathrm{d}\Gamma}_{\mathrm{d}x} + \int_{BC} 2x \underbrace{\mathrm{d}\Gamma}_{\mathrm{d}y} + \int_{CD} 4y \underbrace{\mathrm{d}\Gamma}_{-\mathrm{d}x} + \int_{DA} (-2x) \underbrace{\mathrm{d}\Gamma}_{-\mathrm{d}y}$$
$$= \int_{-1}^{1} 4\, \mathrm{d}x + \int_{-1}^{1} 2\, \mathrm{d}y + \int_{-1}^{1} 4\, \mathrm{d}x + \int_{-1}^{1} 2\, \mathrm{d}x = 24.$$

Thus the divergence theorem for the domain is verified.

Application to heat equation

Fourier’s law in 1D is:

$$q = -k \frac{\mathrm{dT}}{\mathrm{dx}}$$

In 2D the flux has two components q_x and q_y. In this case, Fourier law becomes.

$$\vec{q} = -k\vec{\nabla} T \text{ or } \mathbf{q} = -k\boldsymbol{\nabla} T,$$

where k is a positive constant and the -ve sign indicates that heat flows from hot to cold temperature. The flux balance equation can be derived link in 1D as:

$\nabla \cdot \boldsymbol{q} + s = 0$, where s is the source time. Hence, the full equation now becomes:

$$k \nabla^2 T + s = 0,$$

where s is positive if the heat flows into the system and sign convention for the heat flux $\boldsymbol{q}$ is positive along the outward normal $\boldsymbol{n}$ to the boundary. Here ∇^2 is:

$$\nabla^2 = \vec{\nabla} \cdot \vec{\nabla} = \boldsymbol{\nabla}^{\mathrm{T}} \boldsymbol{\nabla} = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}.$$

This equation is called as Poisson’s equation. In the case when $s = 0$, this becomes the Laplace equation. If the conductivity is not constant or anisotropic then the Fourier’s law gets modified to

$$\begin{bmatrix} q_x \\ q_y \end{bmatrix} = -\underbrace{\begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix}}_{\mathbf{D}} \begin{bmatrix} \dfrac{\partial T}{\partial x} \\ \dfrac{\partial T}{\partial y} \end{bmatrix},$$

which in the matrix form becomes:

$$\boldsymbol{q} = -\boldsymbol{D} \nabla T,$$

where **D** is conductivity matrix. The final balance equation becomes:

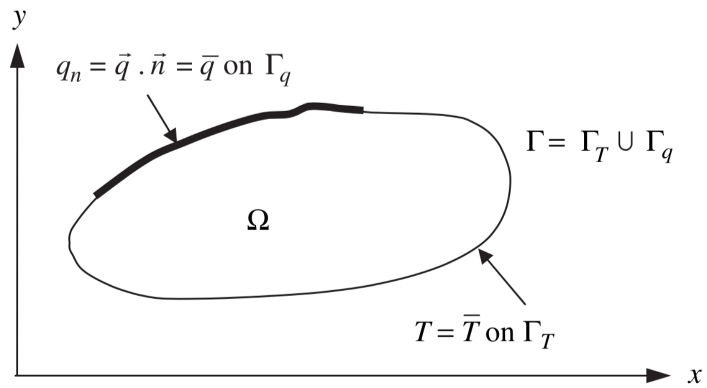
$$\boldsymbol{\nabla}^{\mathrm{T}} (\mathbf{D} \boldsymbol{\nabla} T) + s = 0.$$

Here,

$$\mathbf{D} = \begin{bmatrix} k & 0 \\ 0 & k \end{bmatrix} = k\mathbf{I}.$$

corresponds to the isotropic case. The boundary conditions are specified as follows:

$$\Gamma_q \cup \Gamma_T = \Gamma, \qquad \Gamma_q \cap \Gamma_T = 0.$$



Problem domain and boundary conditions: temperature Γ_T and flux Γ_q. The boundary conditions are specified as

$$T(x,y) = \overline{T}(x,y) \quad \text{on} \quad \Gamma_T,$$

which are essential boundary conditions also called as Dirichlet conditions. The other boundary conditions are:

$$q_n = \vec{q} \cdot \vec{n} = \bar{q} \quad \text{on} \quad \Gamma_q.$$

This could be written more explicitly as:

$q_n = -k\,\boldsymbol{n}^T \nabla T$, on Γ_q.

The flux depends on the derivatives of the temperature and correspond to natural boundary conditions. The box below describes how to Strong form is written in vector and matrix form as:

Box 6.1. Strong form (vector notation) for heat conduction

$$\begin{array}{llll}
\text{(a) energy balance :} & \vec{\nabla} \cdot \vec{q} - s = 0 & \text{on} & \Omega, \\
\text{(b) Fourier's law :} & \vec{q} = -k\vec{\nabla}T & \text{on} & \Omega, \\
\text{(c) natural BC :} & q_n = \vec{q} \cdot \vec{n} = \bar{q} & \text{on} & \Gamma_q, \\
\text{(d) essential BC :} & T = \overline{T} & \text{on} & \Gamma_T.
\end{array}$$

Box 6.2. Strong form (matrix notation) for heat conduction

$$
\begin{array}{lll}
\text{(a) energy balance :} & \boldsymbol{\nabla}^{\mathrm{T}}\mathbf{q} - s = 0 & \text{on } \Omega, \\
\text{(b) Fourier's law :} & \mathbf{q} = -\mathbf{D}\boldsymbol{\nabla}T & \text{on } \Omega, \\
\text{(c) natural BC :} & q_n = \mathbf{q}^{\mathrm{T}}\mathbf{n} = \bar{q} & \text{on } \Gamma_q, \\
\text{(d) essential BC :} & T = \overline{T} & \text{on } \Gamma_T,
\end{array}
$$

The variables s, $\boldsymbol{D}$, $\bar{T}$ and $\bar{q}$ are the inputs to the problem along with the geometry of the domain.

Weak form:

The weak form for the strong form and the natural boundary condition becomes:

$$\text{(a)} \quad \int_{\Omega} w(\vec{\nabla}\cdot\vec{q} - s)\, \mathrm{d}\Omega = 0 \;\forall w, \qquad \text{(b)} \quad \int_{\Gamma_q} w(\bar{q} - \vec{q}\cdot\vec{n})\, \mathrm{d}\Gamma = 0 \qquad \forall w.$$

where w correspond to a sufficiently smooth but otherwise arbitrary weight function. Now, applying the Green's formula, we get:

$$\int_{\Omega} w\vec{\nabla}\cdot\vec{q}\, \mathrm{d}\Omega = \oint_{\Gamma} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma - \int_{\Omega} \vec{\nabla}w\cdot\vec{q}\, \mathrm{d}\Omega \qquad \forall w.$$

Using the above two equations, we get:

$$\int_{\Omega} \vec{\nabla}w\cdot\vec{q}\, \mathrm{d}\Omega = \oint_{\Gamma} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega = \int_{\Gamma_q} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma + \int_{\Gamma_T} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega.$$

In this equation, the boundary is split into two parts Γ_q and Γ_T.

$$\int_{\Omega} \vec{\nabla}w\cdot\vec{q}\, \mathrm{d}\Omega = \int_{\Gamma_q} w\bar{q}\, \mathrm{d}\Gamma + \int_{\Gamma_T} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega.$$

The substituting the weak form in (b) above we get

$$\int_{\Omega} \vec{\nabla}w\cdot\vec{q}\, \mathrm{d}\Omega = \int_{\Gamma_q} w\bar{q}\, \mathrm{d}\Gamma + \int_{\Gamma_T} w\vec{q}\cdot\vec{n}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega.$$

If now we note that $w = 0$ as earlier on the essential boundary Γ_T, this simplies to

$$\int_{\Omega} \vec{\nabla}w\cdot\vec{q}\, \mathrm{d}\Omega = \int_{\Gamma_q} w\bar{q}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega \qquad \forall w \in U_0,$$

where U_0 is the set of sufficiently smooth functions that vanish on the essential boundary. The definition of smoothness is the same as in 1D – C^0continuous and L_2 integrable over the domain. Expressing the final weak form in matrix notation we obtain

$$\int_{\Omega} (\boldsymbol{\nabla}w)^{\mathrm{T}}\mathbf{q}\, \mathrm{d}\Omega = \int_{\Gamma_q} w\bar{q}\, \mathrm{d}\Gamma - \int_{\Omega} ws\, \mathrm{d}\Omega \qquad \forall w \in U_0.$$

Note that this weak form is true for any material, linear or nonlinear. In the current case for the linear case, we use the expression for Fourier's law in the term involving $\boldsymbol{q}$ above, we obtain

Box 6.3. Weak form (matrix notation) for heat conduction

find $T \in U$ such that:

$$\int_{\Omega} (\nabla w)^{\mathrm{T}} \mathbf{D} \nabla T \, \mathrm{d}\Omega = - \int_{\Gamma_q} w \bar{q} \, \mathrm{d}\Gamma + \int_{\Omega} w s \, \mathrm{d}\Omega \qquad \forall w \in U_0.$$

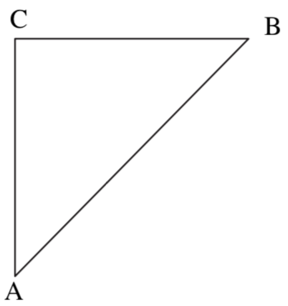
Exercise: Show the equivalence of weak form and strong form.
The 2D formulation could easily be extended to 3D.

Shape functions for 2D scalar field.

The trial functions should satisfy C^0 continuity and be complete. The approach is very similar to that in 1D but there are some complexities in 2D that we need to address.

Triangle elements

The entire domain is divided into triangular elements



Triangular domain

as shown above. The trial solution for each element can take different forms:

$$\begin{aligned}
&\text{(a)} \quad \theta^e(x,y) = \alpha_1^e + \alpha_2^e x + \alpha_3^e y, \\
&\text{(b)} \quad \theta^e(x,y) = \alpha_1^e + \alpha_2^e x + \alpha_3^e y^2, \\
&\text{(c)} \quad \theta^e(x,y) = \alpha_1^e + \alpha_2^e x + \alpha_3^e y + \alpha_4^e xy + \alpha_5^e x^2 + \alpha_6^e x^3 y, \\
&\text{(d)} \quad \theta^e(x,y) = \alpha_1^e + \alpha_2^e x + \alpha_3^e y + \alpha_4^e x^2 y^2 + \alpha_5^e xy + \alpha_6^e y^3.
\end{aligned}$$

(a) is a complete, linear element. (b) will not converge since it is missing the linear term in *y*. (c) is complete and quadratic in *x* and *y*. (d) The quadratic terms do not come systematically (complete), so the convergence will be the same as that of (a). Complete polynomials appear from the Pascal triangle.

$$\begin{array}{ccccccccl}
 & & & 1 & & & & & \text{constant} \\
 & & x & & y & & & & \text{linear} \\
 & x^2 & & xy & & y^2 & & & \text{quadratic} \\
x^3 & & x^2 y & & xy^2 & & y^3 & & \text{cubic}
\end{array}$$

Pascal triangle in 2D. Using the Pascal triangle the interpolation functions can be composed as follows.

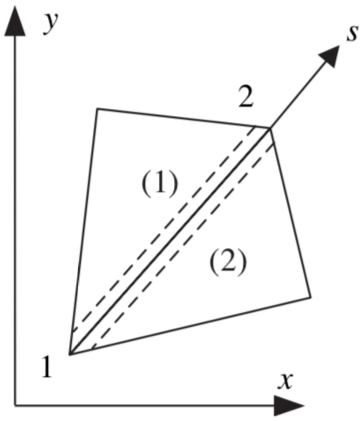
$$\begin{aligned}
\text{linear}: \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y,\\
\text{quadratic}: \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y + \alpha_3^e x^2 + \alpha_4^e xy + \alpha_5^e y^2,\\
\text{cubic}: \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y + \alpha_3^e x^2 + \alpha_4^e xy + \alpha_5^e y^2 + \alpha_6^e x^3 + \alpha_7^e x^2 y + \alpha_8^e y^2 x + \alpha_9^e y^3.
\end{aligned}$$

Construction of linear elements with C^0continuity

Consider two elements with the interpolation functions:

$$\theta^{(1)} = \alpha_0^{(1)} + \alpha_1^{(1)} x + \alpha_2^{(1)} y, \qquad \theta^{(2)} = \alpha_0^{(2)} + \alpha_1^{(2)} x + \alpha_2^{(2)} y,$$

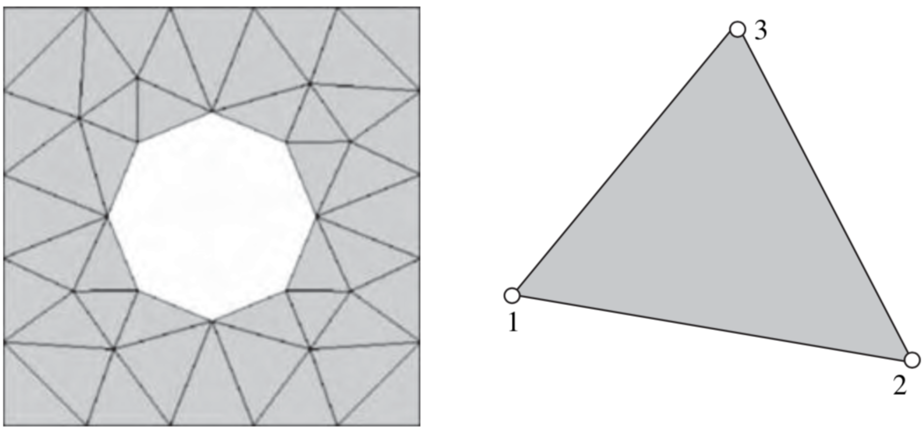
The C^0continuity has to be seen as below



i.e.

$$\theta^{(1)}(s) = \theta^{(2)}(s).$$

The procedure is the same as in 1D FEA, i.e., writing the coefficients in terms of θ**.**



(a) (b)

Triangle elements are extremely versatile, but have lower convergence than quadrilateral elements. The edges are straight, and so any curved domain is approximated with straight lines. If the element size is sufficiently small then the error of approximation is diminished. So for the scalar field θ^e:

$$\theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y = \underbrace{\begin{bmatrix} 1 & x & y \end{bmatrix}}_{\mathbf{p}(x,y)} \underbrace{\begin{bmatrix} \alpha_0^e \\ \alpha_1^e \\ \alpha_2^e \end{bmatrix}}_{\boldsymbol{\alpha}^e} = \mathbf{p}(x,y)\boldsymbol{\alpha}^e.$$

The number of α unknowns are the same as the number of θ unknowns on the nodes of the element:

$$\theta^e(x_1^e, y_1^e) = \theta_1^e,\ \theta^e(x_2^e, y_2^e) = \theta_2^e \text{ and } \theta^e(x_3^e, y_3^e) = \theta_3^e$$

The number of nodes is done in **anti-clockwise** manner. This could be written in the matrix form as:

$$\begin{aligned} \theta_1^e &= \alpha_0^e + \alpha_1^e x_1^e + \alpha_2^e y_1^e \\ \theta_2^e &= \alpha_0^e + \alpha_1^e x_2^e + \alpha_2^e y_2^e \\ \theta_3^e &= \alpha_0^e + \alpha_1^e x_3^e + \alpha_2^e y_3^e \end{aligned} \quad \Rightarrow \quad \underbrace{\begin{bmatrix} \theta_1^e \\ \theta_2^e \\ \theta_3^e \end{bmatrix}}_{\mathbf{d}^e} = \underbrace{\begin{bmatrix} 1 & x_1^e & y_1^e \\ 1 & x_2^e & y_2^e \\ 1 & x_3^e & y_3^e \end{bmatrix}}_{\mathbf{M}^e} \underbrace{\begin{bmatrix} \alpha_0^e \\ \alpha_1^e \\ \alpha_2^e \end{bmatrix}}_{\boldsymbol{\alpha}^e}.$$

The intermediate steps are the same as in 1D and can be written as:

$$\mathbf{d}^e = \mathbf{M}^e \boldsymbol{\alpha}^e.$$

Hence,

$$\boldsymbol{\alpha}^e = (\mathbf{M}^e)^{-1}\mathbf{d}^e.$$

Substitution gives

$$\theta^e(x,y) = \mathbf{p}(x,y)(\mathbf{M}^e)^{-1}\mathbf{d}^e.$$

as in the 1D case,

$$\theta^e(x,y) = \mathbf{N}^e(x,y)\mathbf{d}^e.$$

Thus the shape functions are then given as

$$\mathbf{N}^e(x,y) = \mathbf{p}(x,y)(\mathbf{M}^e)^{-\mathbf{1}} \equiv [N_1^e(x,y) \quad N_2^e(x,y) \quad N_3^e(x,y)].$$

The shape functions can be evaluated from the following expression:

$$(\mathbf{M}^e)^{-1} = \frac{1}{2A^e}\begin{bmatrix} y_2^e - y_3^e & y_3^e - y_1^e & y_1^e - y_2^e \\ x_3^e - x_2^e & x_1^e - x_3^e & x_2^e - x_1^e \\ x_2^e y_3^e - x_3^e y_2^e & x_3^e y_1^e - x_1^e y_3^e & x_1^e y_2^e - x_2^e y_1^e \end{bmatrix},$$

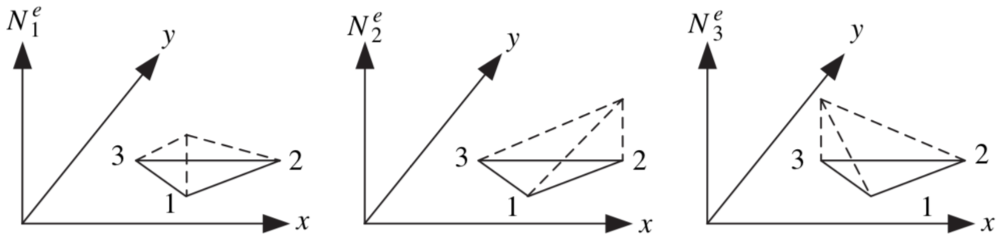
Since the area of the triangle is:

$$2A^e = \det(\mathbf{M}^e) = (x_2^e y_3^e - x_3^e y_2^e) - (x_1^e y_3^e - x_3^e y_1^e) + (x_1^e y_2^e - x_2^e y_1^e).$$

The final expression for the shape functions are:

$$\begin{aligned} N_1^e &= \frac{1}{2A^e}(x_2^e y_3^e - x_3^e y_2^e + (y_2^e - y_3^e)x + (x_3^e - x_2^e)y), \\ N_2^e &= \frac{1}{2A^e}(x_3^e y_1^e - x_1^e y_3^e + (y_3^e - y_1^e)x + (x_1^e - x_3^e)y), \\ N_3^e &= \frac{1}{2A^e}(x_1^e y_2^e - x_2^e y_1^e + (y_1^e - y_2^e)x + (x_2^e - x_1^e)y). \end{aligned}$$

As expected, the shape functions are linear in x and y.



Three noded triangular element shape functions.

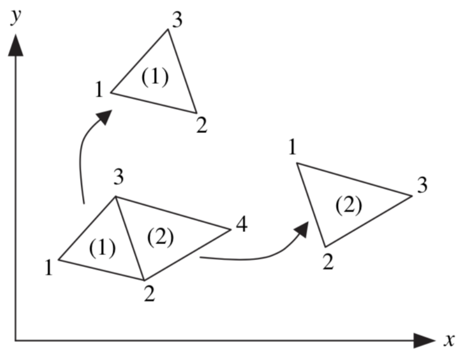
It could be easily verified that:

$$N_I^e(x_J^e, y_J^e) = \delta_{IJ}.$$

and the sum of shape functions is equal to 1.

Also note that the dashed line corresponds to the value of function along the edges of the triangles, is clearly linear. Hence, the value of the field along the edge only depend on the values of the function on the two nodes of the edge.

Global approximation and continuity



Local and global numbering

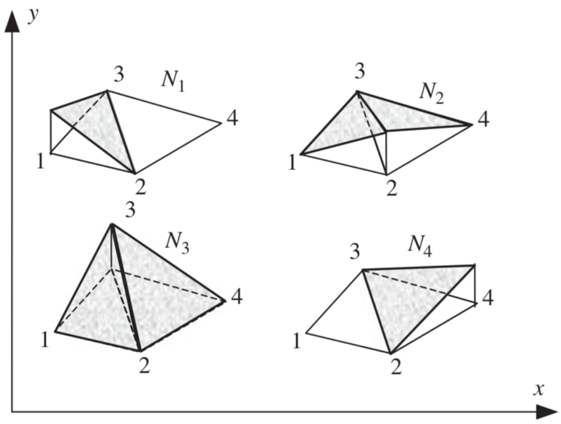
Like in the 1D case, the global shape functions could be given in terms of the element shape functions as:

$$\mathbf{N}^{\mathrm{T}} = \sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{N}^{e\mathrm{T}},$$

The trial solutions are approximated by a linear combination of C^0 global shape functions

$$\theta^h = \mathbf{N}\mathbf{d} = \sum_{I=1}^{n_{\mathrm{np}}} N_I d_I;$$

so that continuity C^0 continuity of $\theta^h$is guaranteed. Since the value of shape function along the shared edge depends only on values at the nodes that are shared between the two elements, the C^0continuity of the global shape functions is automatically guaranteed.



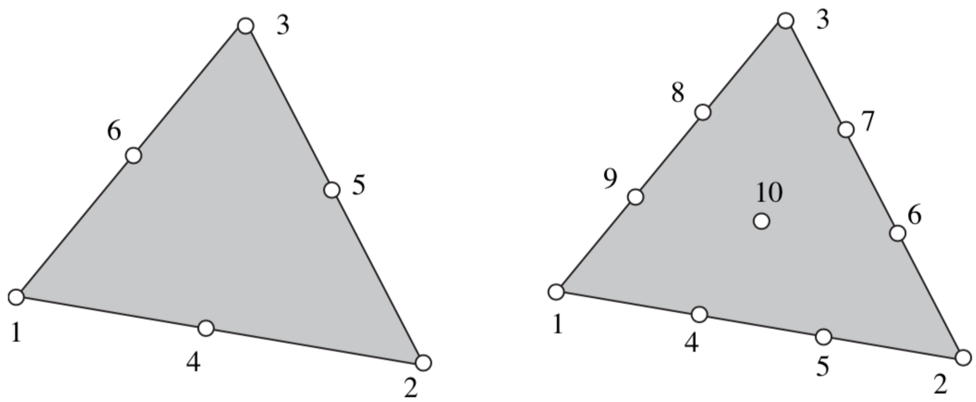
C^0shape functions for a two-element mesh. Only global node numbering is shown.

Using higher order triangular elements.

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

$$\theta^e(s) = \beta_0^e + \beta_1^e s + \beta_2^e s^2.$$

$$\begin{aligned} \text{linear:} \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y, \\ \text{quadratic:} \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y + \alpha_3^e x^2 + \alpha_4^e xy + \alpha_5^e y^2, \\ \text{cubic:} \quad & \theta^e(x,y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y + \alpha_3^e x^2 + \alpha_4^e xy + \alpha_5^e y^2 + \alpha_6^e x^3 + \alpha_7^e x^2 y + \alpha_8^e y^2 x + \alpha_9^e y^3. \end{aligned}$$



Very interesting idea to generate general shape functions

$$\nabla\theta^e = \begin{bmatrix} \dfrac{\partial\theta^e}{\partial x} \\ \dfrac{\partial\theta^e}{\partial y} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial N_1^e}{\partial x}\theta_1^e + \dfrac{\partial N_2^e}{\partial x}\theta_2^e + \dfrac{\partial N_3^e}{\partial x}\theta_3^e \\ \dfrac{\partial N_1^e}{\partial y}\theta_1^e + \dfrac{\partial N_2^e}{\partial y}\theta_2^e + \dfrac{\partial N_3^e}{\partial y}\theta_3^e \end{bmatrix}$$

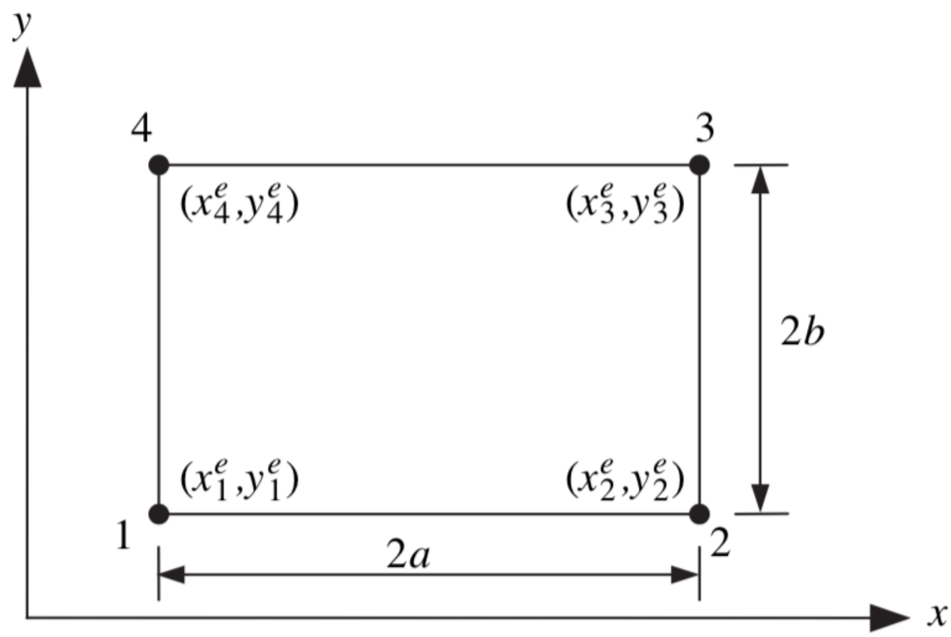
$$= \underbrace{\begin{bmatrix} \dfrac{\partial N_1^e}{\partial x} & \dfrac{\partial N_2^e}{\partial x} & \dfrac{\partial N_3^e}{\partial x} \\ \dfrac{\partial N_1^e}{\partial y} & \dfrac{\partial N_2^e}{\partial y} & \dfrac{\partial N_3^e}{\partial y} \end{bmatrix}}_{\mathbf{B}^e} \underbrace{\begin{bmatrix} \theta_1^e \\ \theta_2^e \\ \theta_3^e \end{bmatrix}}_{\mathbf{d}^e} = \mathbf{B}^e\mathbf{d}^e.$$

For linear element, the $\boldsymbol{B}^e$ matrix is given as:

$$\mathbf{B}^e = \frac{1}{2A^e}\begin{bmatrix} (y_2^e - y_3^e) & (y_3^e - y_1^e) & (y_1^e - y_2^e) \\ (x_3^e - x_2^e) & (x_1^e - x_3^e) & (x_2^e - x_1^e) \end{bmatrix}.$$

Similar to the linear 1D element.

Quadrilateral elements.

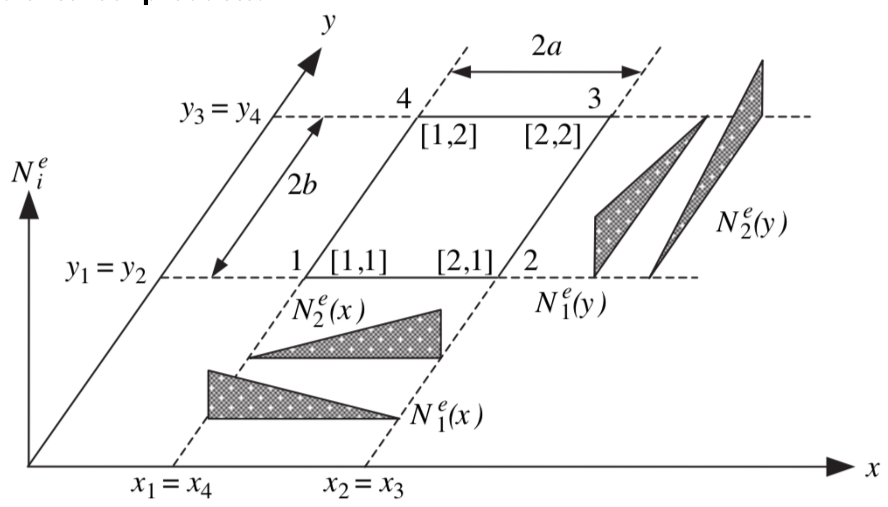


Four-node rectangle element:

$$\theta^e(x, y) = \alpha_0^e + \alpha_1^e x + \alpha_2^e y + \alpha_3^e xy.$$

Why the term $x\,y$? We need four equations (nodal variables) to find the four unknowns. This term has all the linearity properties, since the shape functions need to be linear at the edges (else mode nodes to be added)!

Even though we can invert the shape functions, a more clever method is generally implemented: **method of tensor products.**



Hence, for node I, J, the shape function becomes:

$$N_{[I,J]}^e(x, y) = N_I^e(x) N_J^e(y) \quad \text{for} \quad I = 1, 2 \quad \text{and} \quad J = 1, 2.$$

These shape functions automatically satisfy:

$$N_{[I,J]}^e(x_M^e, y_L^e) = N_I^e(x_M) N_J^e(y_L) = \delta_{IM}\delta_{JL}.$$

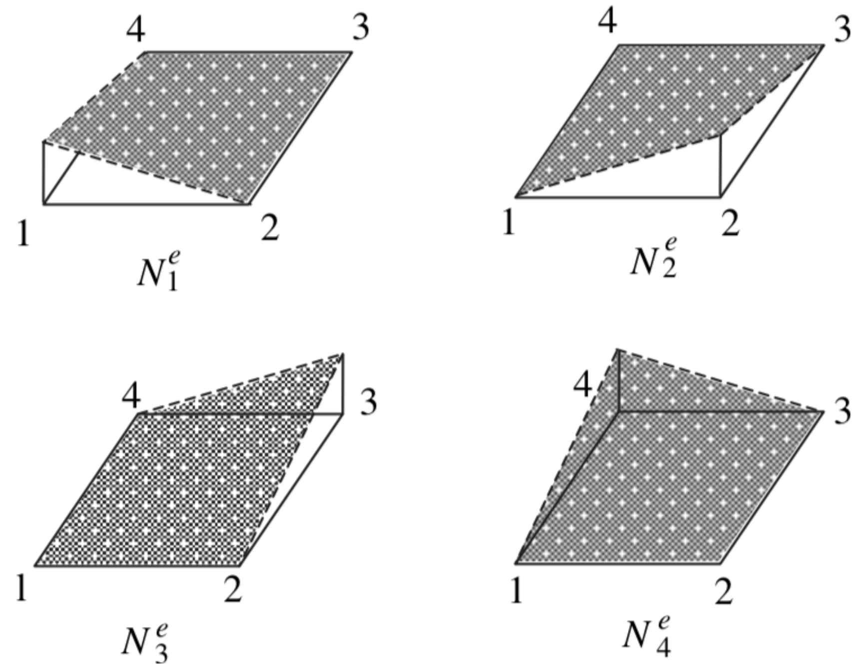
The delta property of the shape functions is also automatically satisfied.

$$N_{[I,J]}^e(x_M^e, y_L^e) = N_I^e(x_M) N_J^e(y_L) = \delta_{IM}\delta_{JL}.$$

The nodal values of the shape functions are given in the table below:

K	I	J	$N_1^e(x_I^e)$	$N_2^e(x_I^e)$	$N_1^e(y_J^e)$	$N_2^e(y_J^e)$	2D: $N_K^e(x,y) = N_{[I,J]}^e(x,y)$
1	1	1	1	0	1	0	$N_1^e(x)N_1^e(y)$
2	2	1	0	1	1	0	$N_2^e(x)N_1^e(y)$
3	2	2	0	1	0	1	$N_2^e(x)N_2^e(y)$
4	1	2	1	0	0	1	$N_1^e(x)N_2^e(y)$

Consequently, the shape functions look as follows:



In a more explicit form, the shape functions are written as:

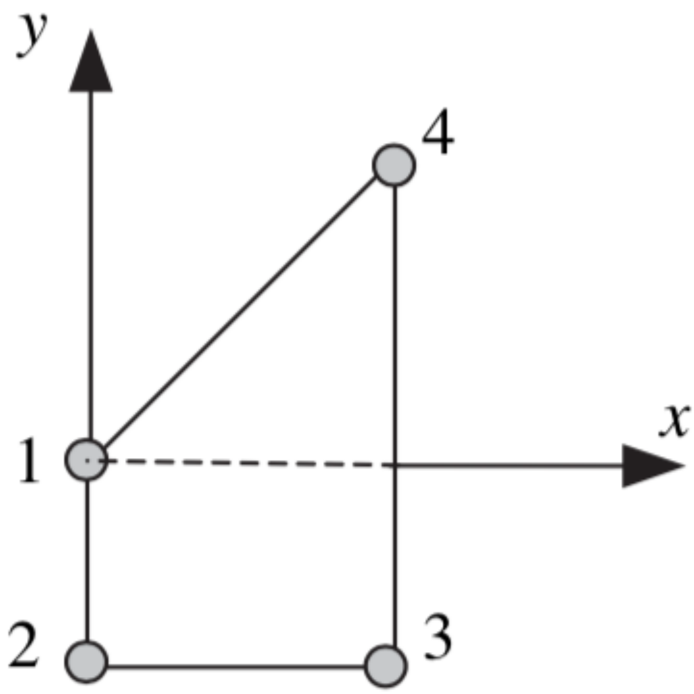
$$N_1^e(x,y) = \frac{x - x_2^e}{x_1^e - x_2^e}\frac{y - y_4^e}{y_1^e - y_4^e} = \frac{1}{A^e}(x - x_2^e)(y - y_4^e),$$

$$N_2^e(x,y) = \frac{x - x_1^e}{x_2^e - x_1^e}\frac{y - y_4^e}{y_1^e - y_4^e} = -\frac{1}{A^e}(x - x_1^e)(y - y_4^e),$$

$$N_3^e(x,y) = \frac{x - x_1^e}{x_2^e - x_1^e}\frac{y - y_1^e}{y_4^e - y_1^e} = \frac{1}{A^e}(x - x_1^e)(y - y_1^e),$$

$$N_4^e(x,y) = \frac{x - x_2^e}{x_1^e - x_2^e}\frac{y - y_1^e}{y_4^e - y_1^e} = -\frac{1}{A^e}(x - x_2^e)(y - y_1^e),$$

The shape functions obtained thus are not applicable for any general quadrilateral other than a rectangle. For example consider the shape below:



Along the line 1 – 4, the equation of the line is $x = y$. In this case, the shape function along the edge 1 – 4 has to be quadratic and not linear. Hence, to maintain C^0continuity, we need to have additional node along that edge, which essentially contradicts what we want. There is an extremely powerful technique called isoparametric mapping that can be used to generate shape functions for any arbitrary quadrilateral with multiple nodes.

Isoparametric Mapping

For a two noded linear element:

$$x = x_1^e N_1^e(\xi) + x_2^e N_2^e(\xi) = x_1^e \frac{1-\xi}{2} + x_2^e \frac{1+\xi}{2}, \qquad \xi \in [-1,\ 1].$$

Similarly, for the field θ, we also get:

$$\begin{aligned}\theta &= \theta_1^e \frac{x - x_2^e}{x_1^e - x_2^e} + \theta_2^e \frac{x - x_1^e}{x_2^e - x_1^e} \\ &= \theta_1^e \frac{x_1^e(1-\xi) + x_2^e(1+\xi) - 2x_2^e}{2(x_1^e - x_2^e)} + \theta_2^e \frac{x_1^e(1-\xi) + x_2^e(1+\xi) - 2x_1^e}{2(x_2^e - x_1^e)} \\ &= \theta_1^e \frac{1-\xi}{2} + \theta_2^e \frac{1+\xi}{2}.\end{aligned}$$

Thus, remarkably, we see that the same interpolation is true for both the spatial coordinate x and the field θ.

For a quadrilateral, the shape functions should look something as follows:

$$x(\xi,\eta) = \mathbf{N}^{4Q}(\xi,\eta)\mathbf{x}^e, \qquad y(\xi,\eta) = \mathbf{N}^{4Q}(\xi,\eta)\mathbf{y}^e,$$

$$\mathbf{x}^e = [x_1^e \quad x_2^e \quad x_3^e \quad x_4^e]^{\mathrm{T}}, \qquad \mathbf{y}^e = [y_1^e \quad y_2^e \quad y_3^e \quad y_4^e]^{\mathrm{T}}.$$

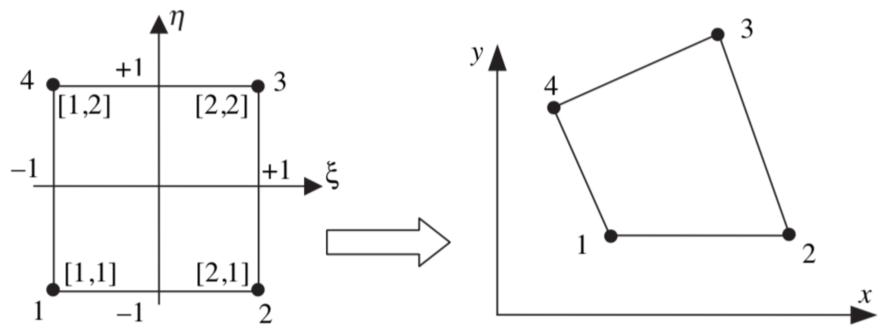
and $\boldsymbol{N}^{4Q}(\xi,\ \eta)$ are the four-node element shape functions in the parent coordinates.

Node I	ξ_I	η_I
1	-1	-1
2	1	-1
3	1	1
4	-1	1

Replacing the values of coordinates with ξ, η, we get the following shape function:

$$N_I^{4Q}(\xi,\eta) = \frac{1}{4}(1+\xi_I\xi)(1+\eta_I\eta),$$

Note that this shape functions are independent of element. Hence, for a quadrilateral, if we think of the **parent shape** as:



mapped to the **physical** quadrilateral, and use the idea for the shape outer product for the rectangular shape, we get the following. The shape function will also look as:

$$\theta^e(\xi,\eta) = \alpha_0^e + \alpha_1^e\xi + \alpha_2^e\eta + \alpha_3^e\xi\eta.$$

Continuity of isoparametric elements:

Along any of the edge elements, the mapping is linear. Thus we need only two values to exactly describe the value of the function along the edge – the two nodal values can then exactly describe the function.

For example, along the right edge, $\xi = 1$,

$$N_2^{4Q}(\xi = 1, \eta) = \frac{1}{2}(1 - \eta).$$

Thus, very clearly the function is linear along this edge. This is also C^0 continous with the

Some calculations to show that the all the edges map to straight edges, but not straight lines map to straight lines.

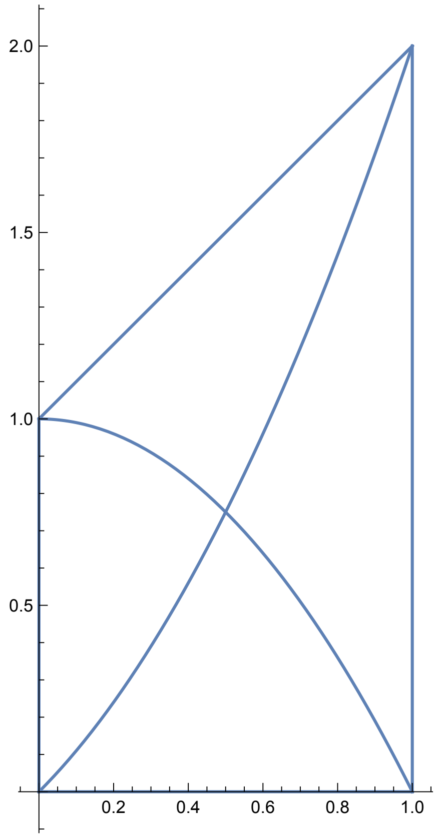
```
N1[ξ_, η_] := 1/4 (1 - ξ) (1 - η);
N2[ξ_, η_] := 1/4 (1 + ξ) (1 - η);
N3[ξ_, η_] := 1/4 (1 + ξ) (1 + η);
N4[ξ_, η_] := 1/4 (1 - ξ) (1 + η);

{x1, x2, x3, x4} = {0, 1, 1, 0};
{y1, y2, y3, y4} = {0, 0, 2, 1};
x = N1[ξ, η] x1 + N2[ξ, η] x2 + N3[ξ, η] x3 + N4[ξ, η] x4
y = N1[ξ, η] y1 + N2[ξ, η] y2 + N3[ξ, η] y3 + N4[ξ, η] y4
```

$$\frac{1}{4}(1-\eta)(1+\xi) + \frac{1}{4}(1+\eta)(1+\xi)$$

$$\frac{1}{4}(1+\eta)(1-\xi) + \frac{1}{2}(1+\eta)(1+\xi)$$

```
X = x /. {ξ → η};
Y = y /. {ξ → η};
```

```
A = ParametricPlot[{X, Y}, {η, -1, 1}];
B = ParametricPlot[{x, y} /. ξ → -1, {η, -1, 1}];
c = ParametricPlot[{x, y} /. ξ → 1, {η, -1, 1}];
d = ParametricPlot[{x, y} /. η → 1, {ξ, -1, 1}];
e = ParametricPlot[{x, y} /. η → -1, {ξ, -1, 1}];
f = ParametricPlot[{x, y} /. ξ → -η, {η, -1, 1}];
Show[A, B, c, d, e, f]
```



Derivatives of isoparametric shape functions

This process is a bit more involved due to changes of parameters in 2D.

$$\boldsymbol{\nabla}\theta^e = \mathbf{B}^e \mathbf{d}^e,$$

$$\mathbf{B}^e = \begin{bmatrix} \dfrac{\partial N_1^{4Q}}{\partial x} & \dfrac{\partial N_2^{4Q}}{\partial x} & \dfrac{\partial N_3^{4Q}}{\partial x} & \dfrac{\partial N_4^{4Q}}{\partial x} \\ \dfrac{\partial N_1^{4Q}}{\partial y} & \dfrac{\partial N_2^{4Q}}{\partial y} & \dfrac{\partial N_3^{4Q}}{\partial y} & \dfrac{\partial N_4^{4Q}}{\partial y} \end{bmatrix}.$$

We will need to use the chain rule as follows:

$$\frac{\partial N_I^{4Q}}{\partial \xi} = \frac{\partial N_I^{4Q}}{\partial x}\frac{\partial x}{\partial \xi} + \frac{\partial N_I^{4Q}}{\partial y}\frac{\partial y}{\partial \xi} \qquad \text{or} \qquad \begin{bmatrix} \frac{\partial N_I^{4Q}}{\partial \xi} \\ \frac{\partial N_I^{4Q}}{\partial \eta} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}}_{\mathbf{J}^e} \begin{bmatrix} \frac{\partial N_I^{4Q}}{\partial x} \\ \frac{\partial N_I^{4Q}}{\partial y} \end{bmatrix}.$$
$$\frac{\partial N_I^{4Q}}{\partial \eta} = \frac{\partial N_I^{4Q}}{\partial x}\frac{\partial x}{\partial \eta} + \frac{\partial N_I^{4Q}}{\partial y}\frac{\partial y}{\partial \eta}$$

The derivatives in terms of x and y will be expressed as

$$\begin{bmatrix} \frac{\partial N_I^{4Q}}{\partial x} \\ \frac{\partial N_I^{4Q}}{\partial y} \end{bmatrix} = (\mathbf{J}^e)^{-1} \begin{bmatrix} \frac{\partial N_I^{4Q}}{\partial \xi} \\ \frac{\partial N_I^{4Q}}{\partial \eta} \end{bmatrix}, \qquad \mathbf{J}^e = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}.$$

In terms of concise matrix form we will get:

$$\nabla \mathbf{N}_I^{4Q} = (\mathbf{J}^e)^{-1} \mathbf{G} \mathbf{N}_I^{4Q},$$

Where, **G** is the gradient operator given as:

$$\mathbf{G} = \begin{bmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{bmatrix}.$$

and the Jacobian is expressible in terms of element co-ordinates and hence has the subscript e on it.

$$\mathbf{J}^e = \begin{bmatrix} \sum_{I=1}^{4} \frac{\partial N_I^{4Q}}{\partial \xi} x_I^e & \sum_{I=1}^{4} \frac{\partial N_I^{4Q}}{\partial \xi} y_I^e \\ \sum_{I=1}^{4} \frac{\partial N_I^{4Q}}{\partial \eta} x_I^e & \sum_{I=1}^{4} \frac{\partial N_I^{4Q}}{\partial \eta} y_I^e \end{bmatrix} = \begin{bmatrix} \frac{\partial N_1^{4Q}}{\partial \xi} & \frac{\partial N_2^{4Q}}{\partial \xi} & \frac{\partial N_3^{4Q}}{\partial \xi} & \frac{\partial N_4^{4Q}}{\partial \xi} \\ \frac{\partial N_1^{4Q}}{\partial \eta} & \frac{\partial N_2^{4Q}}{\partial \eta} & \frac{\partial N_3^{4Q}}{\partial \eta} & \frac{\partial N_4^{4Q}}{\partial \eta} \end{bmatrix} \begin{bmatrix} x_1^e & y_1^e \\ x_2^e & y_2^e \\ x_3^e & y_3^e \\ x_4^e & y_4^e \end{bmatrix}.$$

The quantity $\boldsymbol{B}^e$ is given as:

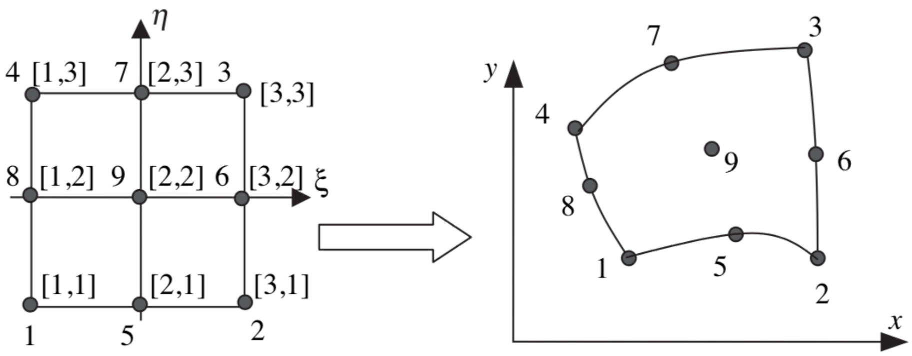
$$\mathbf{B}^e = (\mathbf{J}^e)^{-1} \mathbf{G} \mathbf{N}^{4Q}.$$

The value $(\boldsymbol{J}^e)^{-1}$ to exist,

$$|\mathbf{J}^e| \equiv \det(\mathbf{J}^e) > 0 \qquad \forall e \text{ and } (x, y).$$

The map between (ξ, η) and (x, y) is unique.

Higher order quadrilateral elements



Note the numbering convention.

The shape function is simply:

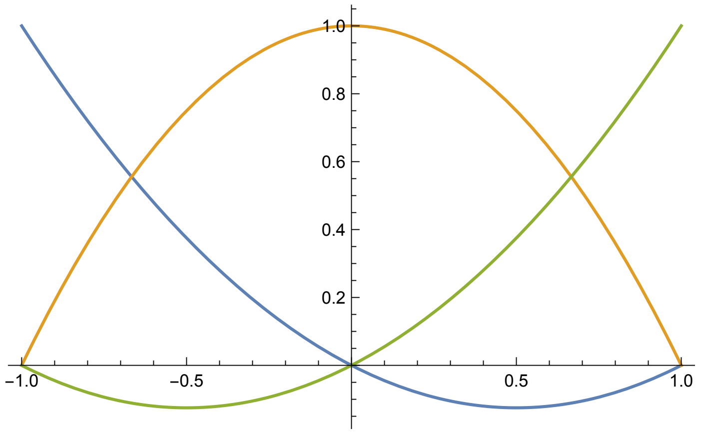
$$N_K^{9Q}(\xi, \eta) = N_{[I,J]}^{9Q}(\xi, \eta) = N_I^{3L}(\xi) N_J^{3L}(\eta),$$

Relating the node number in terms of *I* and *J* as shown in the above figure.

K	*I*	*J*
1	1	1
2	3	1
3	3	3
4	1	3
5	2	1
6	3	2
7	2	3
8	1	2
9	2	2

The most important observation is that, we can map the parent square to physically curved edges.
Plotting the actual shape functions

```
N1[x_] := x (x - 1)/2
N2[x_] := (x + 1) (1 - x)
N3[x_] := (x + 1) x/2
Plot[{N1[x], N2[x], N3[x]}, {x, -1, 1}]
```




```
N1[ξ, η] = N1[ξ] × N1[η]
N2[ξ, η] = N3[ξ] × N1[η]
N3[ξ, η] = N3[ξ] × N3[η]
N4[ξ, η] = N1[ξ] × N3[η]
N5[ξ, η] = N2[ξ] × N1[η]
N6[ξ, η] = N3[ξ] × N2[η]
N7[ξ, η] = N2[ξ] × N3[η]
N8[ξ, η] = N1[ξ] × N2[η]
N9[ξ, η] = N2[ξ] × N2[η]
```

$$\frac{1}{4} (-1+\eta)\, \eta\, (-1+\xi)\, \xi$$

$$\frac{1}{4} (-1+\eta)\, \eta\, \xi\, (1+\xi)$$

$$\frac{1}{4} \eta\, (1+\eta)\, \xi\, (1+\xi)$$

$$\frac{1}{4} \eta\, (1+\eta)\, (-1+\xi)\, \xi$$

$$\frac{1}{2} (-1+\eta)\, \eta\, (1-\xi)\, (1+\xi)$$

$$\frac{1}{2} (1-\eta)\, (1+\eta)\, \xi\, (1+\xi)$$

$$\frac{1}{2} \eta\, (1+\eta)\, (1-\xi)\, (1+\xi)$$

$$\frac{1}{2} (1-\eta)\, (1+\eta)\, (-1+\xi)\, \xi$$

$$(1-\eta)\, (1+\eta)\, (1-\xi)\, (1+\xi)$$

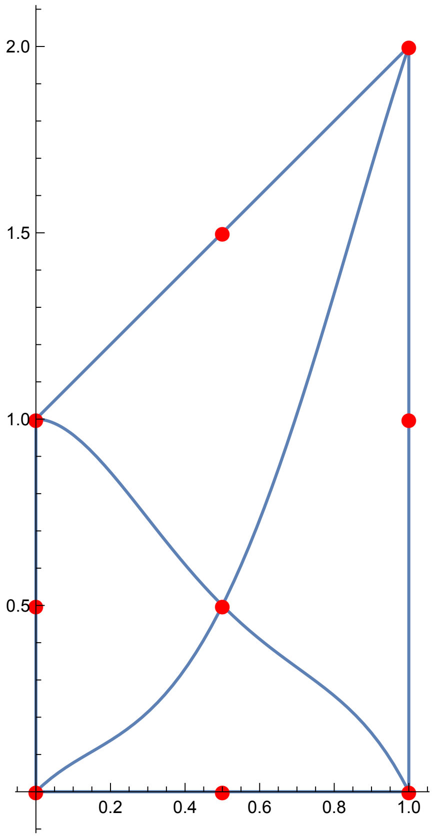
```
{x1, x2, x3, x4, x5, x6, x7, x8, x9} = {0, 1, 1, 0, 0.5, 1, 0.5, 0., 0.5};
{y1, y2, y3, y4, y5, y6, y7, y8, y9} = {0, 0, 2, 1, 0, 1, 1.5, 0.5, 0.5};
xx = {x1, x2, x3, x4, x5, x6, x7, x8, x9};
yy = {y1, y2, y3, y4, y5, y6, y7, y8, y9};
NN = {N1[ξ, η], N2[ξ, η], N3[ξ, η], N4[ξ, η],
   N5[ξ, η], N6[ξ, η], N7[ξ, η], N8[ξ, η], N9[ξ, η]};
x = NN.xx
y = NN.yy
```

$$0. + 0.25\,(-1+\eta)\,\eta\,(1-\xi)\,(1+\xi) + 0.5\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi) + 0.25\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi) + \frac{1}{4}\,(-1+\eta)\,\eta\,\xi\,(1+\xi) + \frac{1}{2}\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi) + \frac{1}{4}\,\eta\,(1+\eta)\,\xi\,(1+\xi)$$

$$0.25\,(1-\eta)\,(1+\eta)\,(-1+\xi)\,\xi + \frac{1}{4}\,\eta\,(1+\eta)\,(-1+\xi)\,\xi + 0.5\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi) + 0.75\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi) + \frac{1}{2}\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi) + \frac{1}{2}\,\eta\,(1+\eta)\,\xi\,(1+\xi)$$

```
A = ParametricPlot[{x, y} /. ξ → η, {η, -1, 1}];
B = ParametricPlot[{x, y} /. ξ → -1, {η, -1, 1}];
c = ParametricPlot[{x, y} /. ξ → 1, {η, -1, 1}];
d = ParametricPlot[{x, y} /. η → 1, {ξ, -1, 1}];
e = ParametricPlot[{x, y} /. η → -1, {ξ, -1, 1}];
f = ParametricPlot[{x, y} /. ξ → -η, {η, -1, 1}];
Transpose[{xx, yy}]
g = ListPlot[%, PlotMarkers → {Automatic, Medium}, PlotStyle → Red];
Show[A, B, c, d, e, f, g]
```

{{0, 0}, {1, 0}, {1, 2}, {0, 1}, {0.5, 0}, {1, 1}, {0.5, 1.5}, {0., 0.5}, {0.5, 0.5}}




```
g
```

$$\left\{0. + 1.5\,(-1+\eta)\,\eta\,(1-\xi)\,(1+\xi) + 3.\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi) + 1.5\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi) + \frac{3}{2}\,(-1+\eta)\,\eta\,\xi\,(1+\xi) + 3\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi) + \frac{3}{2}\,\eta\,(1+\eta)\,\xi\,(1+\xi),\right.$$
$$\left. 1.75\,(1-\eta)\,(1+\eta)\,(-1+\xi)\,\xi + \frac{7}{4}\,\eta\,(1+\eta)\,(-1+\xi)\,\xi + 3.5\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi) + 5.25\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi) + \frac{7}{2}\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi) + \frac{7}{2}\,\eta\,(1+\eta)\,\xi\,(1+\xi)\right\}$$

Along edge 1-7-4

```
(N3[ξ, η] θ3 + N4[ξ, η] θ4 + N7[ξ, η] θ7) /. η → 1
```

$$\frac{1}{2}\,\theta 4\,(-1+\xi)\,\xi + \theta 7\,(1-\xi)\,(1+\xi) + \frac{1}{2}\,\theta 3\,\xi\,(1+\xi)$$

Mapping curved edges

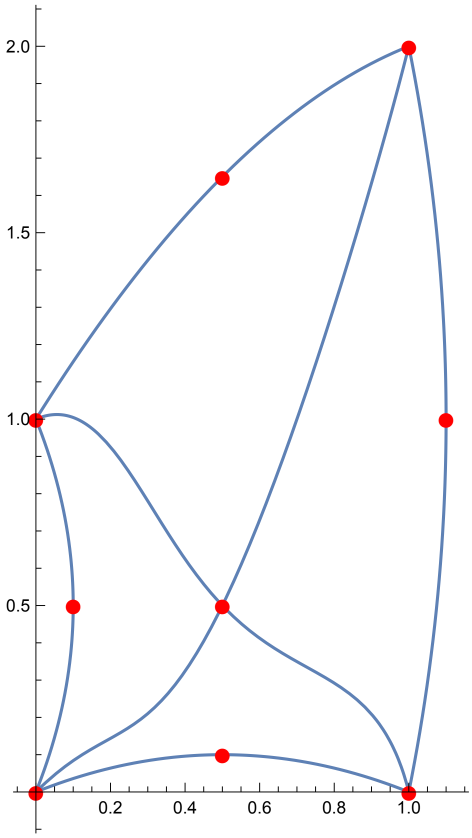
```
{x1, x2, x3, x4, x5, x6, x7, x8, x9} = {0, 1, 1, 0, 0.5, 1.1, 0.5, 0.1, 0.5};
{y1, y2, y3, y4, y5, y6, y7, y8, y9} = {0, 0, 2, 1, 0.1, 1, 1.65, 0.5, 0.5};
xx = {x1, x2, x3, x4, x5, x6, x7, x8, x9};
yy = {y1, y2, y3, y4, y5, y6, y7, y8, y9};
NN = {N1[ξ, η], N2[ξ, η], N3[ξ, η], N4[ξ, η],
   N5[ξ, η], N6[ξ, η], N7[ξ, η], N8[ξ, η], N9[ξ, η]};
x = NN.xx
y = NN.yy
```

$$0.05\,(1-\eta)\,(1+\eta)\,(-1+\xi)\,\xi+0.25\,(-1+\eta)\,\eta\,(1-\xi)\,(1+\xi)+ 0.5\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi)+0.25\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi)+ \frac{1}{4}\,(-1+\eta)\,\eta\,\xi\,(1+\xi)+0.55\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi)+\frac{1}{4}\,\eta\,(1+\eta)\,\xi\,(1+\xi)$$

$$0.25\,(1-\eta)\,(1+\eta)\,(-1+\xi)\,\xi+\frac{1}{4}\,\eta\,(1+\eta)\,(-1+\xi)\,\xi+ 0.05\,(-1+\eta)\,\eta\,(1-\xi)\,(1+\xi)+0.5\,(1-\eta)\,(1+\eta)\,(1-\xi)\,(1+\xi)+ 0.825\,\eta\,(1+\eta)\,(1-\xi)\,(1+\xi)+\frac{1}{2}\,(1-\eta)\,(1+\eta)\,\xi\,(1+\xi)+\frac{1}{2}\,\eta\,(1+\eta)\,\xi\,(1+\xi)$$

```
A = ParametricPlot[{x, y} /. ξ → η, {η, -1, 1}];
B = ParametricPlot[{x, y} /. ξ → -1, {η, -1, 1}];
c = ParametricPlot[{x, y} /. ξ → 1, {η, -1, 1}];
d = ParametricPlot[{x, y} /. η → 1, {ξ, -1, 1}];
e = ParametricPlot[{x, y} /. η → -1, {ξ, -1, 1}];
f = ParametricPlot[{x, y} /. ξ → -η, {η, -1, 1}];
Transpose[{xx, yy}]
g = ListPlot[%, PlotMarkers → {Automatic, Medium}, PlotStyle → Red];
Show[A, B, c, d, e, f, g, PlotRange → All]
```

```
{{0, 0}, {1, 0}, {1, 2}, {0, 1}, {0.5, 0.1},
 {1.1, 1}, {0.5, 1.65}, {0.1, 0.5}, {0.5, 0.5}}
```



Along the edge 1-7-4

```
(N3[ξ, η] θ3 + N4[ξ, η] θ4 + N7[ξ, η] θ7) /. η → 1
```

$$\frac{1}{2}\theta 4\ (-1+\xi)\ \xi+\theta 7\ (1-\xi)\ (1+\xi)+\frac{1}{2}\theta 3\ \xi\ (1+\xi)$$

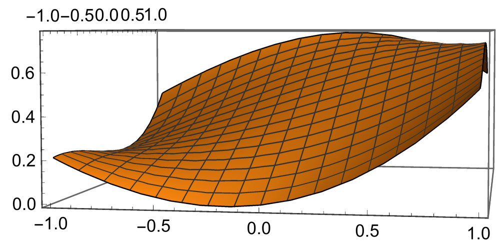
What is the Jacobian for the nine-node element.

```
rξ = {∂ξN1[ξ, η], ∂ξN2[ξ, η], ∂ξN3[ξ, η], ∂ξN4[ξ, η],
    ∂ξN5[ξ, η], ∂ξN6[ξ, η], ∂ξN7[ξ, η], ∂ξN8[ξ, η], ∂ξN9[ξ, η]};
rη = {∂ηN1[ξ, η], ∂ηN2[ξ, η], ∂ηN3[ξ, η], ∂ηN4[ξ, η], ∂ηN5[ξ, η],
    ∂ηN6[ξ, η], ∂ηN7[ξ, η], ∂ηN8[ξ, η], ∂ηN9[ξ, η]};
```

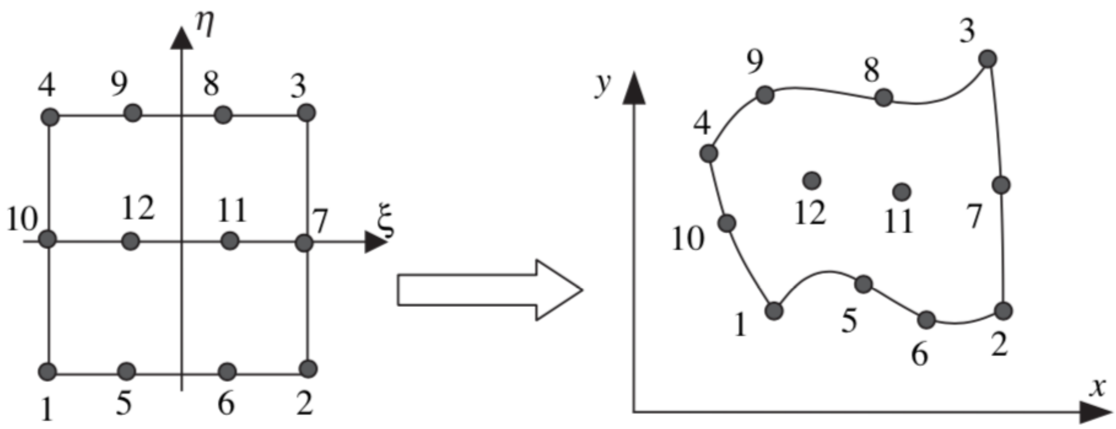
```
{rξ, rη}.Transpose[{xx, yy}] // FullSimplify // Expand;
MatrixForm[%]
Det[%%] // Expand
Plot3D[%, {ξ, -1, 1}, {η, -1, 1}]
```

$$\begin{pmatrix} 0.5 + 0.2\,\xi - 0.2\,\eta^2\,\xi & 0.25 + 0.25\,\eta + 0.5\,\xi - 0.05\,\eta\,\xi - 0.75\,\eta^2\,\xi \\ 0. - 0.2\,\eta\,\xi^2 & 0.775 + 0.75\,\eta + 0.25\,\xi - 0.025\,\xi^2 - 0.75\,\eta\,\xi^2 \end{pmatrix}$$

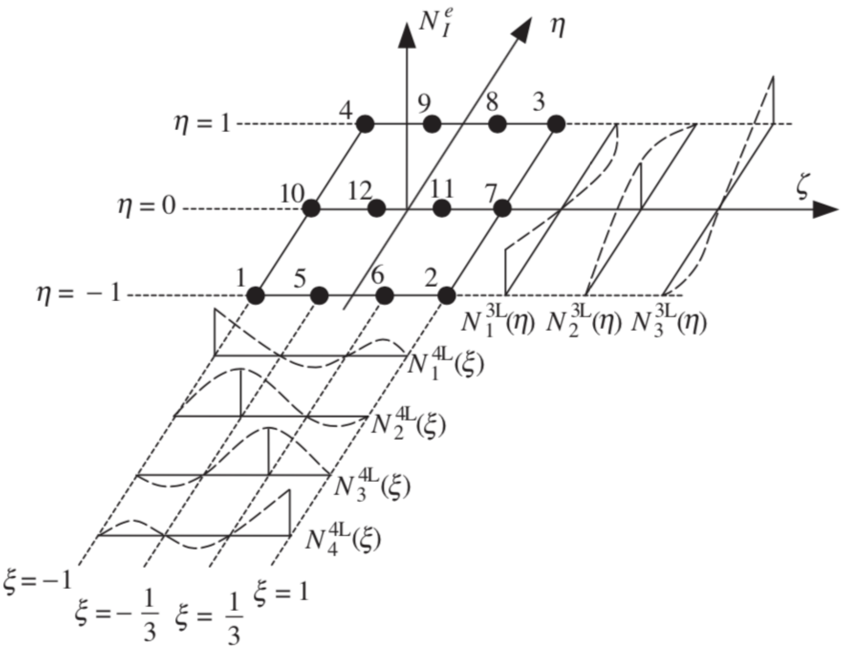
$$0.3875 + 0.375\,\eta + 0.28\,\xi + 0.15\,\eta\,\xi - 0.155\,\eta^2\,\xi - 0.15\,\eta^3\,\xi + 0.0375\,\xi^2 - 0.325\,\eta\,\xi^2 - 0.005\,\xi^3 - 0.05\,\eta\,\xi^3 - 0.005\,\eta^2\,\xi^3$$



Higher order elements



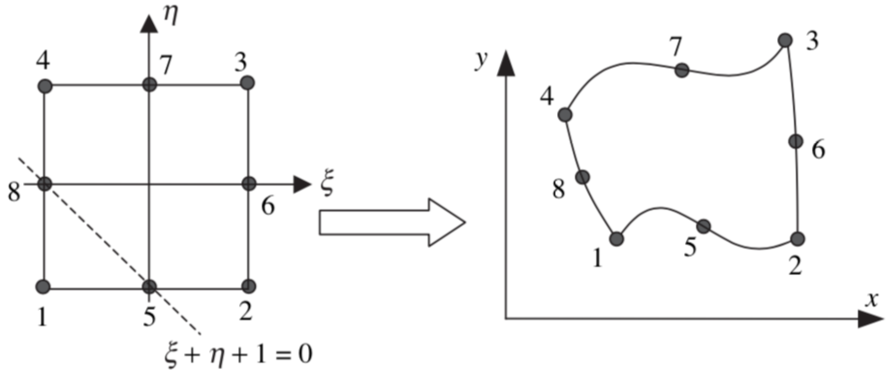
K	I	J	$N_1^{4L}(\xi_I)$	$N_2^{4L}(\xi_I)$	$N_3^{4L}(\xi_I)$	$N_4^{4L}(\xi_I)$	$N_1^{3L}(\eta_I)$	$N_2^{3L}(\eta_I)$	$N_3^{3L}(\eta_I)$	$N_I^{12Q}(\xi,\eta)$
1	1	1	1	0	0	0	1	0	0	$N_1^{4L}(\xi)N_1^{3L}(\eta)$
2	4	1	0	0	0	1	1	0	0	$N_4^{4L}(\xi)N_1^{3L}(\eta)$
3	4	3	0	0	0	1	0	0	1	$N_4^{4L}(\xi)N_3^{3L}(\eta)$
4	1	3	1	0	0	0	0	0	1	$N_1^{4L}(\xi)N_3^{3L}(\eta)$
5	2	1	0	1	0	0	1	0	0	$N_2^{4L}(\xi)N_1^{3L}(\eta)$
6	3	1	0	0	1	0	1	0	0	$N_3^{4L}(\xi)N_1^{3L}(\eta)$
7	4	2	0	0	0	1	0	1	0	$N_4^{4L}(\xi)N_2^{3L}(\eta)$
8	3	3	0	0	1	0	0	0	1	$N_3^{4L}(\xi)N_3^{3L}(\eta)$
9	2	3	0	1	0	0	0	0	1	$N_2^{4L}(\xi)N_3^{3L}(\eta)$
10	1	2	1	0	0	0	0	1	0	$N_1^{4L}(\xi)N_2^{3L}(\eta)$
11	3	2	0	0	1	0	0	1	0	$N_3^{4L}(\xi)N_2^{3L}(\eta)$
12	2	2	0	1	0	0	0	1	0	$N_2^{4L}(\xi)N_2^{3L}(\eta)$



Construction of shape functions for the 12 noded element. The inner nodes are not required for continuity.

Serendipity elements:

Some such elements do not have internal nodes. For example,



This is 8 nodes. How to construct the 8 shape functions? The most important property is the Kroenecker Delta property, i.e., the shape function is 1 at the given node and zero at other nodes.

Consider Node 8, The shape function is $N_1^{8\,Q}(\xi,\ \eta)$. This function $(1-\xi)(1-\eta)(1+\xi+\eta)$ vanishes at all the points except at 1! The value of this function at (-1, -1) is -4, hence the shape function should be

$$N_1^{8\,Q}(\xi,\ \eta) = -\frac{1}{4}(1-\xi)(1-\eta)(1+\xi+\eta)$$

Similarly, for node 4

$$N_4^{8\,Q}(\xi,\ \eta) = \frac{1}{4}(1-\xi)(1+\eta)(-1-\xi+\eta)$$

For node 8

$$N_8^{8\,Q}(\xi,\ \eta) = \frac{1}{2}(1\text{-}\eta)\,(1+\eta)\,(1\text{-}\xi)$$

Such element is called as the Serendipity element.

```
(* checking the Kroenecker Delta property *)
n1 = -1/4 (1 - ξ) (1 - η) (1 + ξ + η) ;
% /. {ξ → -1, η → -1}
n4 = 1/4 (1 - ξ) (1 + η) (-1 - ξ + η) ;
% /. {ξ → -1, η → 1}
n8 = 1/2 (1 - η) (1 + η) (1 - ξ);
% /. {ξ → -1, η → 0}
```

1

1

1

How does the function vary along the edge 1-8-4? Here $\xi = -1$

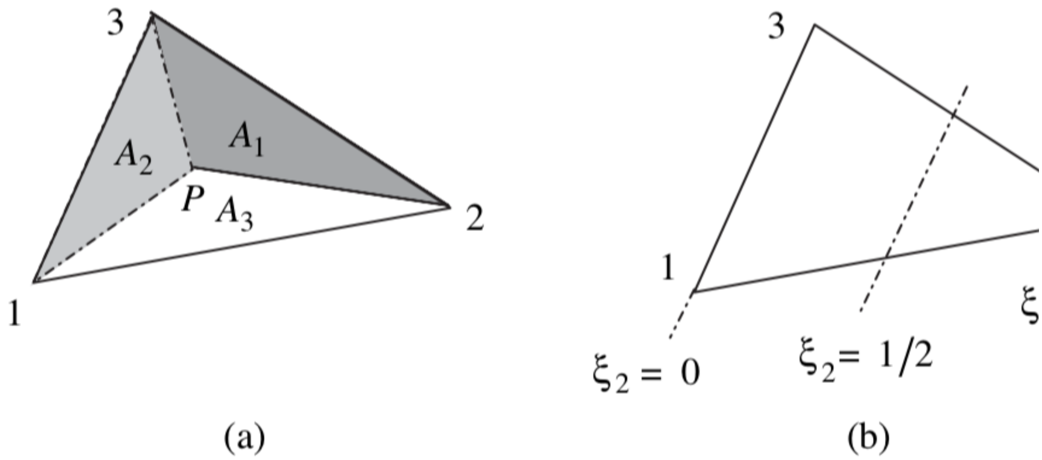
```
(n1 θ1 + n4 θ4 + n8 θ8) /. ξ → -1 // Expand
```

$$-\frac{\eta\,\theta 1}{2}+\frac{\eta^2\,\theta 1}{2}+\frac{\eta\,\theta 4}{2}+\frac{\eta^2\,\theta 4}{2}+\theta 8-\eta^2\,\theta 8$$

This is a complete quadratic polynomial!

Isoparametric elements for triangles:

For a three-node triangular element, the most natural way to define isoparametric coordinates are through centroid coordinates



For every point:
$\xi_I = \frac{A_I}{A}$, as a result of which, the Kroenecker delta property is automatically satisfied.

$$\xi_I(x_J^e, y_J^e) = \delta_{IJ},$$

Consequently, very likely these are actually the interpolants, i.e., the linear functions that were obtained in the previous classes. It is obvious that

$\sum \xi_I = 1$,

Thus, we can express a relation between ξ_I and x, y coordinates.

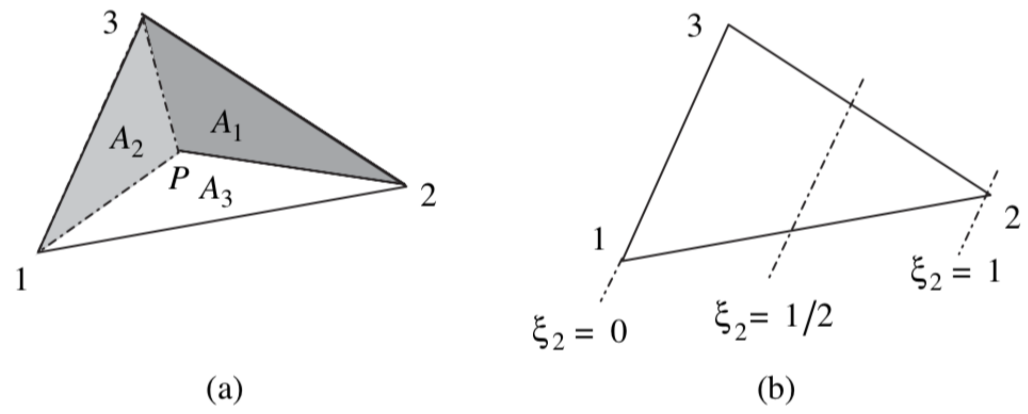
$$\begin{bmatrix} 1 \\ x \\ y \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ x_1^e & x_2^e & x_3^e \\ y_1^e & y_2^e & y_3^e \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix}.$$

This could be inverted to provide:

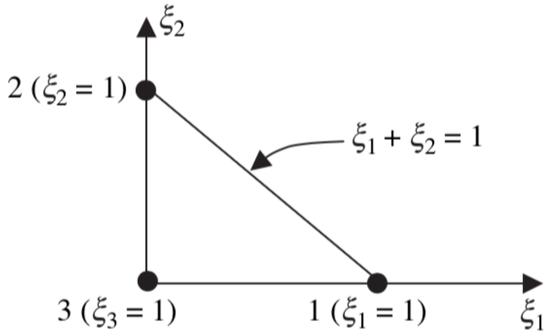
$$\begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} = \frac{1}{2A^e} \begin{bmatrix} x_1^e y_3^e - x_3^e y_2^e & y_{23}^e & x_{32}^e \\ x_3^e y_1^e - x_1^e y_3^e & y_{31}^e & x_{13}^e \\ x_1^e y_2^e - x_2^e y_1^e & y_{12}^e & x_{21}^e \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \end{bmatrix},$$

Here, the notation $x_{IJ}^e = x_I^e - x_J^e$ and so on. Hence, ξ_I are also linear functions of x and y. Since there will be unique shape functions that will satisfy the Kroenecker delta property and linearity, ξ_I indeed has to be shape functions N_I that were obtained earlier. Hence, we can write:

$$\theta^e = \sum_{I=1}^{3} \theta_I^e \xi_I = \theta_1^e \xi_1 + \theta_2^e \xi_2 + \theta_3^e \xi_3.$$



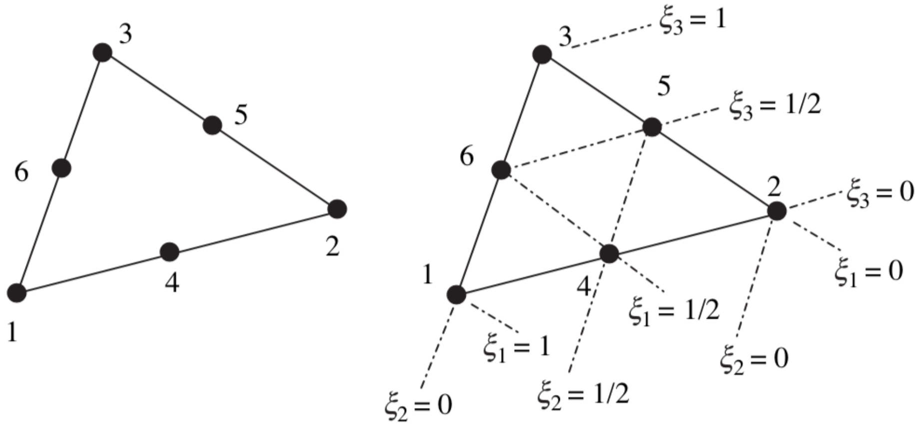
The triangular coordinates can be interpreted is shown in the Figure above. Based, on all these thoughts, similar to what we do for isoparametric quadrilaterals, we can obtain a **parent** element for the physical three noded triangle as:



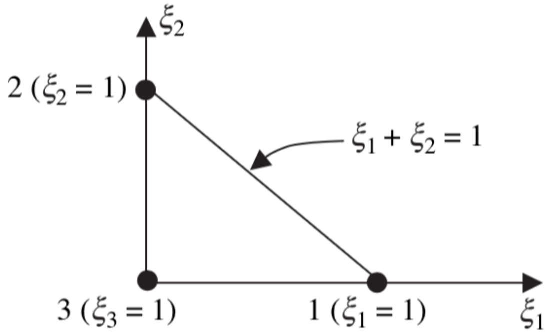
Since the triangular coordinates are linear in x and y, we get the following equation for the relation between the material derivatives and physical derivatives.

$$\frac{\partial \xi_1}{\partial x} = \frac{y_{23}^e}{2A^e} \quad \frac{\partial \xi_2}{\partial x} = \frac{y_{31}^e}{2A^e} \quad \frac{\partial \xi_3}{\partial x} = \frac{y_{12}^e}{2A^e},$$
$$\frac{\partial \xi_1}{\partial y} = \frac{x_{32}^e}{2A^e} \quad \frac{\partial \xi_2}{\partial y} = \frac{x_{13}^e}{2A^e} \quad \frac{\partial \xi_3}{\partial y} = \frac{x_{21}^2}{2A^e}.$$

Higher order elements can be obtained using the same trick as that used for the quadrilaterals.



Also consider this parent triangle for reference



We need 6 nodes for a six node element. The shape functions are easily obtained for example:

Node 1

$N_1 \sim \xi_1\left(\xi_1 - \frac{1}{2}\right)$

Node 4

$N_4 \sim \xi_2\, \xi_1$

The normalisation is to be done by noting that at that particular node I the corresponding value of the shape function in 1.

The final list of shape functions is:

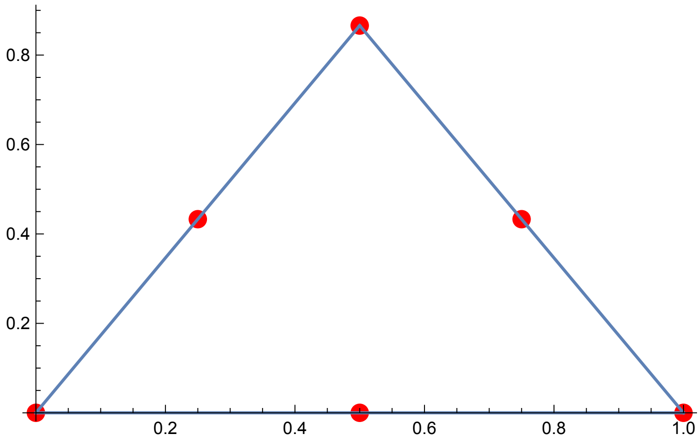
I	$\xi_1(x_I^e, y_I^e)$	$\xi_2(x_I^e, y_I^e)$	$\xi_3(x_I^e, y_I^e)$	$N_I^{6T}(\xi_1, \xi_2, \xi_3)$
1	1	0	0	$\xi_1(2\xi_1 - 1)$
2	0	1	0	$\xi_2(2\xi_2 - 1)$
3	0	0	1	$\xi_3(2\xi_3 - 1)$
4	1/2	1/2	0	$4\xi_1\xi_2$
5	0	1/2	1/2	$4\xi_2\xi_3$
6	1/2	0	1/2	$4\xi_1\xi_3$

How does the map happen?

```
{x1, x2, x3, x4, x5, x6} = {0, 1, 1/2, 0.5, 0.5 (1 + 0.5), 1 / 4};
xx = %;
{y1, y2, y3, y4, y5, y6} = {0, 0, √3/2, 0, √3/4, √3/4};
yy = %;
a = ListPlot[Transpose[{xx, yy}],
   PlotMarkers → {Automatic, Medium}, PlotStyle → Red ];
N1[ξ1_, ξ2_, ξ3_] := ξ1 (2 ξ1 - 1)
N2[ξ1_, ξ2_, ξ3_] := ξ2 (2 ξ2 - 1)
N3[ξ1_, ξ2_, ξ3_] := ξ3 (2 ξ3 - 1)
N4[ξ1_, ξ2_, ξ3_] := 4 ξ1 ξ2
N5[ξ1_, ξ2_, ξ3_] := 4 ξ3 ξ2
N6[ξ1_, ξ2_, ξ3_] := 4 ξ1 ξ3
(* The shape functions are *)
NN = {N1[ξ1, ξ2, ξ3], N2[ξ1, ξ2, ξ3],
   N3[ξ1, ξ2, ξ3], N4[ξ1, ξ2, ξ3], N5[ξ1, ξ2, ξ3], N6[ξ1, ξ2, ξ3]};
x = NN.xx // Expand
y = NN.yy // Expand
(* Plotting the edges *)
b = ParametricPlot[{x, y} /. ξ3 → (1 - ξ1 - ξ2) /. ξ2 → 0, {ξ1, 0, 1}];
c = ParametricPlot[{x, y} /. ξ2 → (1 - ξ1 - ξ3) /. ξ3 → 0, {ξ1, 0, 1}];
d = ParametricPlot[{x, y} /. ξ2 → (1 - ξ1 - ξ3) /. ξ1 → 0, {ξ3, 0, 1}];
Show[a, b, c, d]
```

$$-\xi2 + 2.\ \xi1\ \xi2 + 2\ \xi2^2 - \frac{\xi3}{2} + \xi1\ \xi3 + 3.\ \xi2\ \xi3 + \xi3^2$$

$$-\frac{\sqrt{3}\ \xi3}{2} + \sqrt{3}\ \xi1\ \xi3 + \sqrt{3}\ \xi2\ \xi3 + \sqrt{3}\ \xi3^2$$



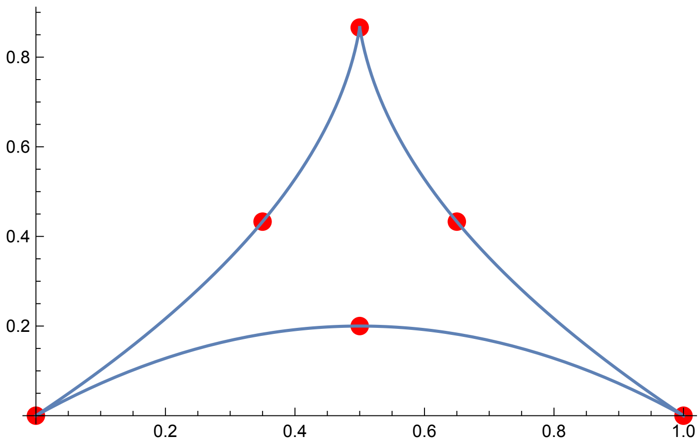

```
{x, y} /. ξ3 → (1 - ξ1 - ξ2) /. ξ2 → 0 // Expand
```

$\left\{0.5 - \frac{\xi 1}{2}, \frac{\sqrt{3}}{2} - \frac{\sqrt{3}\ \xi 1}{2}\right\}$

```
{x1, x2, x3, x4, x5, x6} = {0, 1, 1/2, 0.5, 0.5 (1 + 0.5) - 0.1, 1 / 4 + 0.1};
xx = %;
{y1, y2, y3, y4, y5, y6} = {0, 0, √3/2, 0.2, √3/4, √3/4};
yy = %;
a = ListPlot[Transpose[{xx, yy}],
   PlotMarkers → {Automatic, Medium}, PlotStyle → Red ];
N1[ξ1_, ξ2_, ξ3_] := ξ1 (2 ξ1 - 1)
N2[ξ1_, ξ2_, ξ3_] := ξ2 (2 ξ2 - 1)
N3[ξ1_, ξ2_, ξ3_] := ξ3 (2 ξ3 - 1)
N4[ξ1_, ξ2_, ξ3_] := 4 ξ1 ξ2
N5[ξ1_, ξ2_, ξ3_] := 4 ξ3 ξ2
N6[ξ1_, ξ2_, ξ3_] := 4 ξ1 ξ3
(* The shape functions are *)
NN = {N1[ξ1, ξ2, ξ3], N2[ξ1, ξ2, ξ3],
   N3[ξ1, ξ2, ξ3], N4[ξ1, ξ2, ξ3], N5[ξ1, ξ2, ξ3], N6[ξ1, ξ2, ξ3]};
x = NN.xx // Expand
y = NN.yy // Expand
(* Plotting the edges *)
b = ParametricPlot[{x, y} /. ξ3 → (1 - ξ1 - ξ2) /. ξ2 → 0, {ξ1, 0, 1}];
c = ParametricPlot[{x, y} /. ξ2 → (1 - ξ1 - ξ3) /. ξ3 → 0, {ξ1, 0, 1}];
d = ParametricPlot[{x, y} /. ξ2 → (1 - ξ1 - ξ3) /. ξ1 → 0, {ξ3, 0, 1}];
Show[a, b, c, d, PlotRange → All]
```

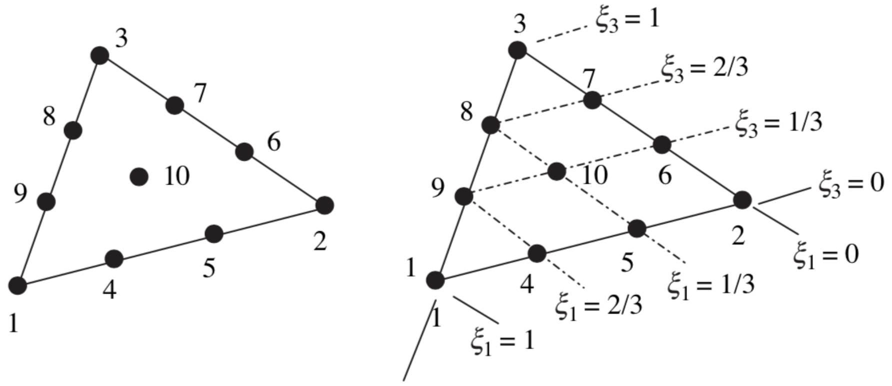
$$-\xi2 + 2.\ \xi1\,\xi2 + 2\,\xi2^2 - \frac{\xi3}{2} + 1.4\,\xi1\,\xi3 + 2.6\,\xi2\,\xi3 + \xi3^2$$

$$0.8\,\xi1\,\xi2 - \frac{\sqrt{3}\,\xi3}{2} + \sqrt{3}\,\xi1\,\xi3 + \sqrt{3}\,\xi2\,\xi3 + \sqrt{3}\,\xi3^2$$



Thus we can represent curved boundaries quite conveniently. Whenever, the edge boundaries remain straight the resulting elements are called as subparametric elements and have better convergence that the elements with curved boundaries. So the curved elements should not be used unless a curved boundary needs to be mimicked with a few number of elements.

Similar rules need to be implemented for higher order triangles:

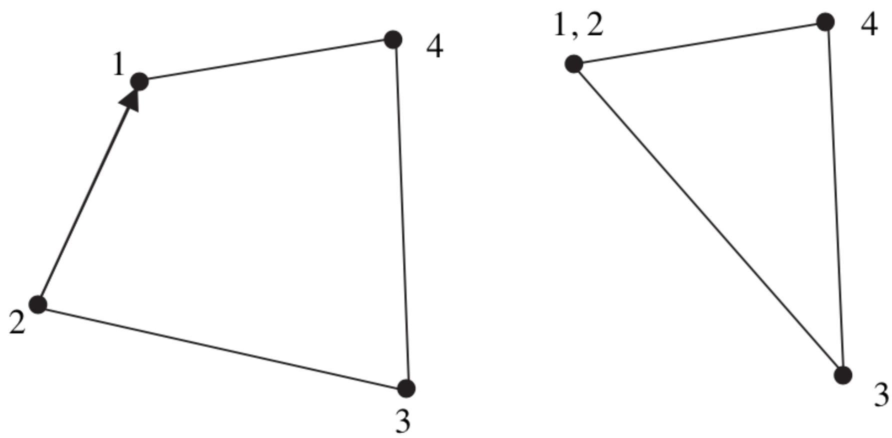


For a cubic element.

I	$\xi_1(x_I^e, y_I^e)$	$\xi_2(x_I^e, y_I^e)$	$\xi_3(x_I^e, y_I^e)$	$N_I^{10T}(\xi_1, \xi_2, \xi_3)$
1	1	0	0	$(9/2)\xi_1(\xi_1 - 1/3)(\xi_1 - 2/3)$
2	0	1	0	$(9/2)\xi_2(\xi_2 - 1/3)(\xi_2 - 2/3)$
3	0	0	1	$(9/2)\xi_3(\xi_3 - 1/3)(\xi_3 - 2/3)$
4	2/3	1/3	0	$(27/2)\xi_1\xi_2(\xi_1 - 1/3)$
5	1/3	2/3	0	$(27/2)\xi_1\xi_2(\xi_2 - 1/3)$
6	0	2/3	1/3	$(27/2)\xi_2\xi_3(\xi_2 - 1/3)$
7	0	1/3	2/3	$(27/2)\xi_2\xi_3(\xi_3 - 1/3)$
8	1/3	0	2/3	$(27/2)\xi_1\xi_3(\xi_3 - 1/3)$
9	2/3	0	1/3	$(27/2)\xi_1\xi_3(\xi_1 - 1/3)$
10	1/3	1/3	1/3	$27\xi_1\xi_2\xi_3$

Shape functions of 10 noded triangular element.

Triangular elements by collapsing the quadrilateral elements:



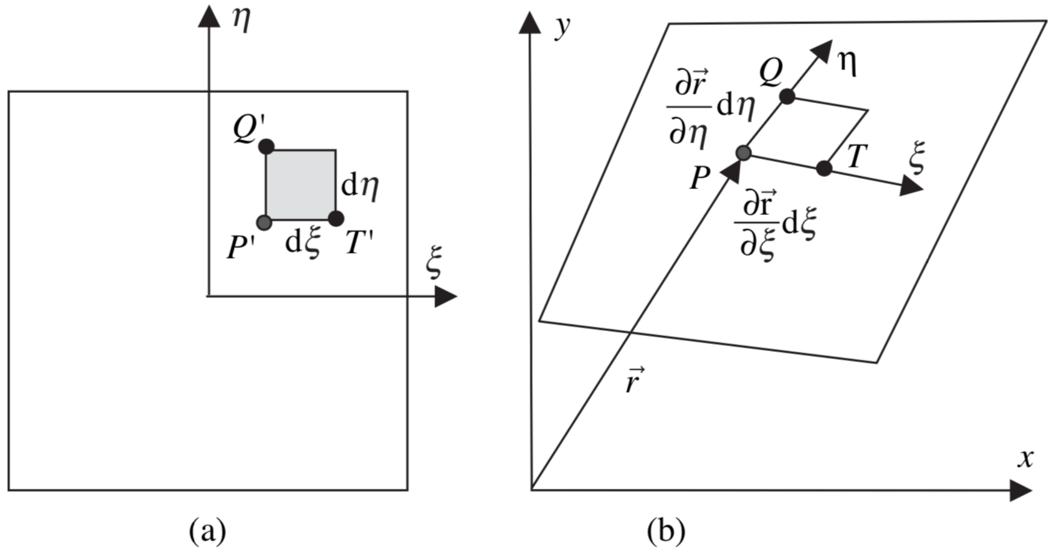
One can obtain triangle element shape functions by collapsing nodes for a quadrilateral element. This will, however, make the Jacobian zero at the boundaries because we have mapping of two points to just one point.

It can be shown that the Isoparametric elements are complete, i.e., they can at least represent a linear field.

Gauss quadrature for Quadrilateral elements.

The integration over a quadrilateral element is given as:

$$I = \int_{\Omega^e} f(\xi, \eta)\, \mathrm{d}\Omega.$$



Mapping of the element between parent and physical domain.

In the physical domain, we have:

$$\vec{a} = \frac{\partial \vec{r}}{\partial \xi}\, \mathrm{d}\xi = \left(\frac{\partial x}{\partial \xi}\vec{i} + \frac{\partial y}{\partial \xi}\vec{j}\right) \mathrm{d}\xi,$$

$$\vec{b} = \frac{\partial \vec{r}}{\partial \eta}\, \mathrm{d}\eta = \left(\frac{\partial x}{\partial \eta}\vec{i} + \frac{\partial y}{\partial \eta}\vec{j}\right) \mathrm{d}\eta.$$

The area $d\,\Omega$ of the physical domain, we get:

$$\mathrm{d}\Omega = \vec{k} \cdot (\vec{a} \times \vec{b}) = \vec{k} \cdot \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ \frac{\partial x}{\partial \xi}\,\mathrm{d}\xi & \frac{\partial y}{\partial \eta}\,\mathrm{d}\eta & 0 \\ \frac{\partial x}{\partial \xi}\,\mathrm{d}\xi & \frac{\partial y}{\partial \eta}\,\mathrm{d}\eta & 0 \end{bmatrix} = \underbrace{\det \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix}}_{|\mathbf{J}^e|} \mathrm{d}\xi\, \mathrm{d}\eta = |\mathbf{J}^e|\, \mathrm{d}\xi\, \mathrm{d}\eta,$$

Hence, the Integral described earlier is

$$I = \int_{\eta=-1}^{1} \int_{\xi=-1}^{1} |\mathbf{J}^e(\xi, \eta)| f(\xi, \eta)\, \mathrm{d}\xi\, \mathrm{d}\eta.$$

First integrating along ξ

$$I = \int_{\eta=-1}^{1} \left(\int_{\xi=-1}^{1} |\mathbf{J}^e(\xi, \eta)| f(\xi, \eta) d\xi \right) d\eta = \int_{r=-1}^{1} \sum_{i=1}^{n_{gp}} W_i |\mathbf{J}^e(\xi_i, \eta)| f(\xi_i, \eta) d\eta$$

Here, the Gauss quadrature is evaluated over the 1D Gauss points in the ξ direction. Then evaluating the Integral over η direction:

$$I = \int_{\eta=-1}^{1} \sum_{i=1}^{n_{gp}} W_i f(\xi_i, \eta)\, |\mathbf{J}^e(\xi_i, \eta)|\, d\eta = \sum_{i=1}^{n_{gp}} \sum_{j=1}^{n_{gp}} W_i W_j \left|\mathbf{J}^e(\xi_i, \eta_j)\right| f\left(\xi_i, \eta_j\right)$$

The points ξ_i and η_j are the Gauss points in 1-D. It is to be noted that $|\, \boldsymbol{J}^e(\xi, \eta)\, |\, f(\xi, \eta)$ is not necessarily a polynomial function of the variables ξ_i, η_i. Hence, Gauss quadrature need not be exact.

Integration over triangular elements.

The integral:

$$\int f\, \mathrm{d}\Omega = \sum_{i=1}^{n_{gp}} W_i\, f\left(\xi_1^{(i)}, \xi_2^{(i)}, \xi_3^{(i)}\right) |\, \boldsymbol{J}^e\left(\xi_1^{(i)}, \xi_2^{(i)}, \xi_3^{(i)}\right).$$

The Gauss-quadrature points are far more tricky in this case.

Integration order	Degree of precision	ξ_1	ξ_2	Weights
Three-point	2	0.1 666 666 666	0.1 666 666 666	0.1 666 666 666
		0.6 666 666 666	0.1 666 666 666	0.1 666 666 666
		0.1 666 666 666	0.6 666 666 666	0.1 666 666 666
		0.1 012 865 073	0.1 012 865 073	0.0 629 695 903
		0.7 974 269 853	0.1 012 865 073	0.0 629 695 903
		0.1 012 865 073	0.7 974 269 853	0.0 629 695 903
Seven-point	5	0.4 701 420 641	0.0 597 158 717	0.0 661 970 764
		0.4 701 420 641	0.4 701 420 641	0.0 661 970 764
		0.0 597 158 717	0.4 701 420 641	0.0 661 970 764
		0.3 333 333 333	0.3 333 333 333	0.1125

The degree of precision means that the quadrature is exact for a polynomial up to that degree. Note also that:

$$\xi_1 + \xi_2 + \xi_3 = 1.$$

For three node triangles (or a straight edge triangle) the Jacobian matrix is:

$$\mathbf{J}^e = \begin{bmatrix} x_1^e - x_3^e & y_1^e - y_3^e \\ x_2^e - x_3^e & y_2^e - y_3^e \end{bmatrix}.$$

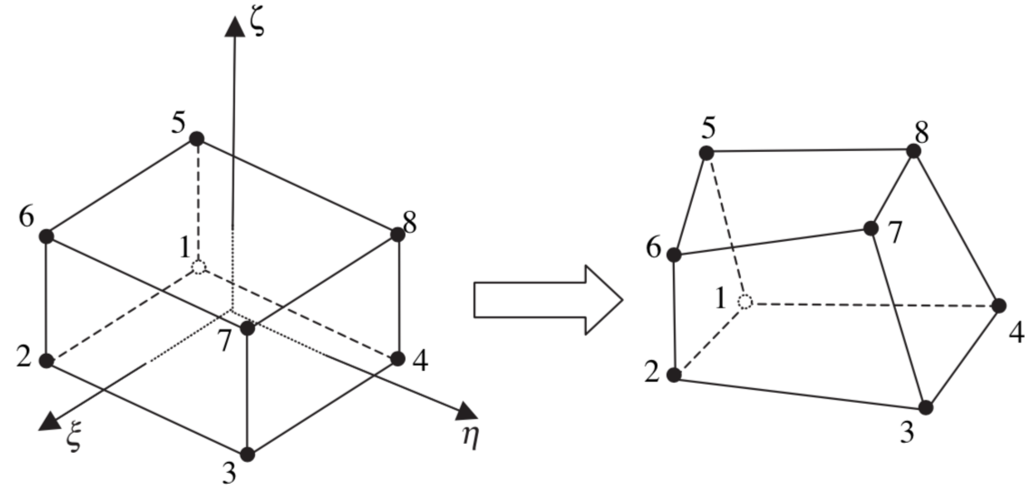
The resulting Jacobian (det(J)) is twice the area of the triangle with coordinates corresponding to 1, 2, 3. It is also the ratio of the area of the reference triangle and the parent triangle.

For straight edge triangle:

$$\int\limits_{\Omega^e} \xi_1^i \xi_2^j \xi_3^k \, \mathrm{d}\Omega = \frac{i!j!k!}{(i+j+k+2)!} 2A^e.$$

3D finite elements

Very similar ideas to described shape functions in 3D. Either tetrahedron (extension of triangle) or brick (extension of quadrilateral) element.



$$x(\xi,\eta,\zeta) = \mathbf{N}^{8\mathrm{H}}(\xi,\eta,\zeta)\mathbf{x}^e,$$
$$y(\xi,\eta,\zeta) = \mathbf{N}^{8\mathrm{H}}(\xi,\eta,\zeta)\mathbf{y}^e,$$
$$z(\xi,\eta,\zeta) = \mathbf{N}^{8\mathrm{H}}(\xi,\eta,\zeta)\mathbf{z}^e,$$

The shape function can be obtained from outer-product via 2-node 1D elements as:

$$N_L^{8\mathrm{H}}(\xi,\eta,\zeta) = N_I^{2\mathrm{L}}(\xi) N_J^{2\mathrm{L}}(\eta) N_K^{2\mathrm{L}}(\zeta).$$

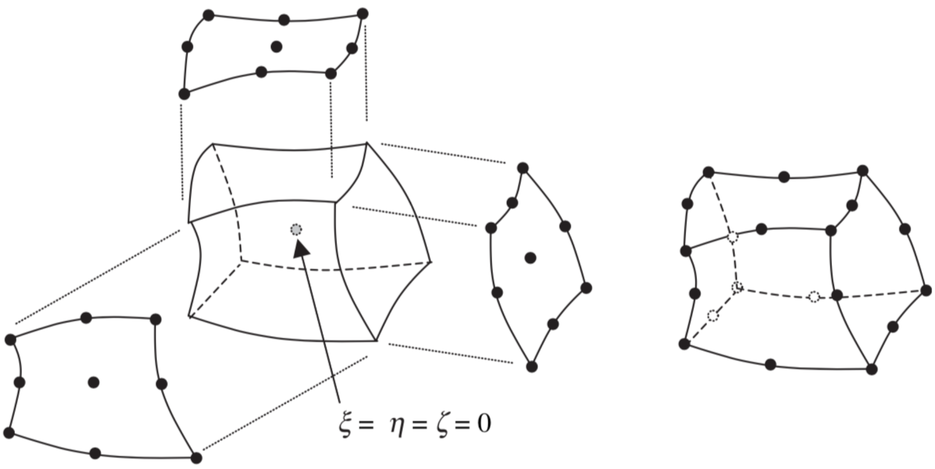
L	I	J	K
1	1	1	1
2	2	1	1
3	2	2	1
4	1	2	1
5	1	1	2
6	2	1	2
7	2	2	2
8	1	2	2

What are the I, J, K corresponding to every element number L of the quadrilateral.

The iso-parametric idea can be used to express any field:

$$\theta^e(\xi, \eta) = \mathbf{N}^{8\mathrm{H}}(\xi, \eta, \zeta)\mathbf{d}^e.$$

It is quite clear that for any of the faces, either ξ, η, ζ assumes a constant value and so the field θ^ebecomes a linear bilinear field in terms of the other two. Since a bilinear function can be uniquely defined in terms of the values at the four nodes, C^0continuity is automatically satisfied. Higher order element obtained in a similar manner by using outer-product of higher order 1D elements, or by using seredipity elements.



(a) (b)

The Jacobian in 3D is:

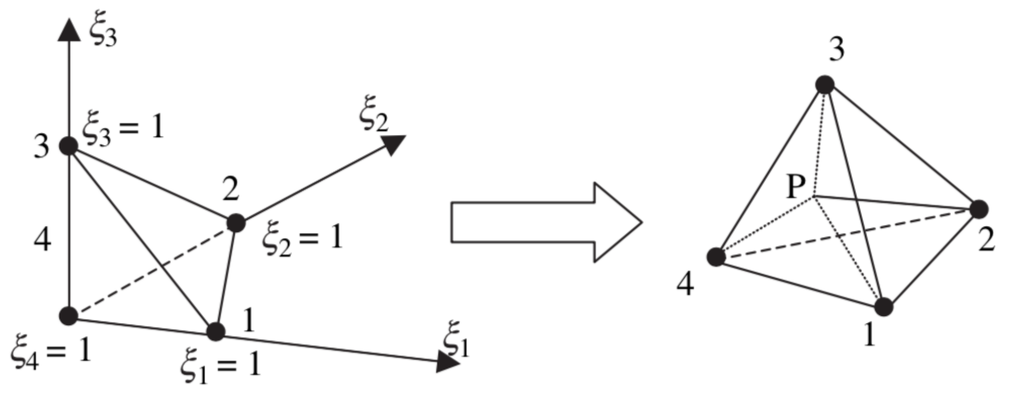
$$\mathbf{J}^e = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} & \frac{\partial z}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} & \frac{\partial z}{\partial \eta} \\ \frac{\partial x}{\partial \zeta} & \frac{\partial y}{\partial \zeta} & \frac{\partial z}{\partial \zeta} \end{bmatrix}.$$

and the integral can be expressed as:

$$I = \int_{\Omega^e} f(\xi, \eta, \zeta)\, \mathrm{d}\Omega = \int_{\xi=-1}^{1} \int_{\eta=-1}^{1} \int_{\zeta=-1}^{1} |\mathbf{J}^e(\xi, \eta, \zeta)| f(\xi, \eta, \zeta)\, \mathrm{d}\xi$$

$$= \sum_{i=1}^{n_{\mathrm{gp}}} \sum_{j=1}^{n_{\mathrm{gp}}} \sum_{k=1}^{n_{\mathrm{gp}}} W_i W_j W_k |\mathbf{J}^e(\xi_i, \eta_j, \zeta_k)| f(\xi_i, \eta_j,$$

Tetrahedral is similar to triangles, but a bit complex in terms of geometry.



(a) (b)

The Tetrahedral coordinates of point P are:

$$\xi_1 = \frac{\text{volume of P234}}{\Omega^e}, \qquad \xi_2 = \frac{\text{volume of P134}}{\Omega^e},$$
$$\xi_3 = \frac{\text{volume of P124}}{\Omega^e}, \qquad \xi_4 = \frac{\text{volume of P123}}{\Omega^e}.$$

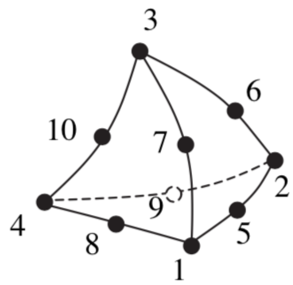
with

$$\xi_1 + \xi_2 + \xi_3 + \xi_4 = 1.$$

The shape functions of a four-node tetrahedron are:

$$N_1^{4\text{Tet}} = \xi_1,$$
$$N_2^{4\text{Tet}} = \xi_2,$$
$$N_3^{4\text{Tet}} = \xi_3,$$
$$N_4^{4\text{Tet}} = \xi_4 = 1 - \xi_1 - \xi_2 - \xi_3.$$

For a 10 node tetrahedron (quadratic) the shape functions are obtained by the same trick as for triangles:



A 10-node curved-face tetrahedral element.

I	$\xi_1(x_I^e, y_I^e)$	$\xi_2(x_I^e, y_I^e)$	$\xi_3(x_I^e, y_I^e)$	$\xi_4(x_I^e, y_I^e)$	$N_I^{10\text{Tet}}(\xi_1, \xi_2, \xi_3, \xi_3)$
1	1	0	0	0	$2\xi_1(\xi_1 - 1/2)$
2	0	1	0	0	$2\xi_2(\xi_2 - 1/2)$
3	0	0	1	0	$2\xi_3(\xi_3 - 1/2)$
4	0	0	0	1	$2\xi_4(\xi_4 - 1/2)$
5	1/2	1/2	0	0	$4\xi_1\xi_2$
6	0	1/2	1/2	0	$4\xi_2\xi_3$
7	1/2	0	1/2	0	$4\xi_1\xi_3$
8	1/2	0	0	1/2	$4\xi_1\xi_4$
9	0	1/2	0	1/2	$4\xi_2\xi_4$
10	0	0	1/2	1/2	$4\xi_3\xi_4$

We will now combine the ideas regarding the Boundary Value problem, Green's theorem and Weak-Form along with the triangular and quadrilateral elements and shape functions developed here to solve 2D BVP using the Finite Element.

Solution to BVP using the finite element method

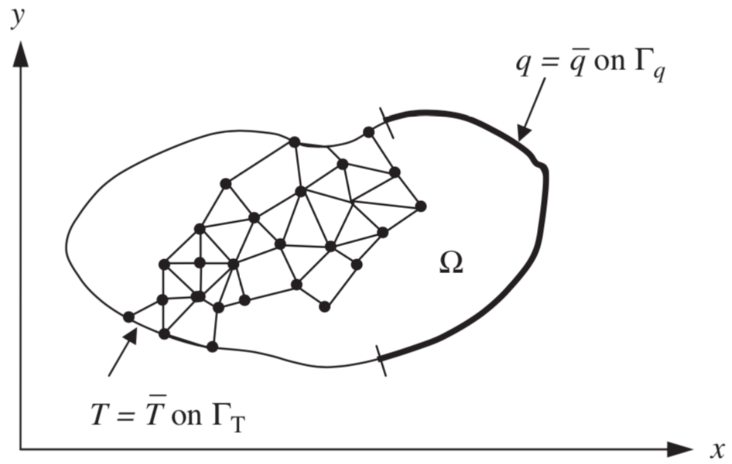
Finite element formulation for heat conduction problem

$$\text{find } T(x, y) \in U \text{ such that :}$$

$$\int_\Omega (\nabla w)^{\mathrm{T}} \mathbf{D} \nabla T \, d\Omega = -\int_{\Gamma_q} w^{\mathrm{T}} \bar{q} \, d\Gamma + \int_\Omega w^{\mathrm{T}} s \, d\Omega \qquad \forall w \in U_0,$$

where

$$\nabla T = \begin{bmatrix} \dfrac{\partial T}{\partial x} \\ \dfrac{\partial T}{\partial y} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} k_{xx} & k_{xy} \\ k_{xy} & k_{yy} \end{bmatrix}$$



Finite element mesh on the domain.

Integrals are replaced with the summation over the elements

$$\sum_{e=1}^{n_{el}} \left(\int_{\Omega^e} (\nabla w^e)^T \mathbf{D}^e (\nabla T^e)\, d\Omega + \int_{\Gamma_q^e} w^{eT} \bar{q}\, d\Gamma - \int_{\Omega^e} w^{eT} s\, d\Omega \right) = 0,$$

The finite element trial function and the weight functions is:

$$T(x,y) \approx T^e(x,y) = \mathbf{N}^e(x,y)\mathbf{d}^e = \sum_{I=1}^{n_{en}} N_I^e(x,y) T_I^e \quad (x,y) \in \Omega^e$$

$$w^T(x,y) \approx w^{eT}(x,y) = \mathbf{N}^e(x,y)\mathbf{w}^e = \sum_{I=1}^{n_{en}} N_I^e(x,y) w_I^e \quad (x,y) \in \Omega^e$$

Here $\boldsymbol{N}^e$ is the element shape function and

$$\mathbf{d}^e = [T_1^e \; T_2^e \; \cdots \; T_{n_{en}}^e]^T$$

and

$$\mathbf{w}^e = [\mathrm{w}_1^e \; \mathrm{w}_2^e \; \cdots \; w_{n_{en}}^e]^T$$

For isoparametric formulation the shape functions are expressed in terms of ξ and η. The element nodal degrees of freedom are expressed as

$$\mathbf{d}^e = \mathbf{L}^e\mathbf{d}.$$

Hence,

$$\text{(a)} \qquad T^e(x,y) = \mathbf{N}^e(x,y)\mathbf{L}^e\mathbf{d}$$

$$\text{(b)} \quad w^{e\mathrm{T}}(x,y) = (\mathbf{N}^e(x,y)\mathbf{w}^e)^{\mathrm{T}} = \mathbf{w}^{\mathrm{T}}\mathbf{L}^{e\mathrm{T}}\mathrm{N}^{e\mathrm{T}}(x,y)$$

The gradient becomes:

$$\nabla T^e = \begin{bmatrix} \dfrac{\partial T^e}{\partial x} \\ \dfrac{\partial T^e}{\partial y} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial N_1^e}{\partial x} T_1^e + \dfrac{\partial N_2^e}{\partial x} T_2^e + \cdots + \dfrac{\partial N_{n_{\mathrm{en}}}^e}{\partial x} T_{n_{\mathrm{en}}}^e \\ \dfrac{\partial N_1^e}{\partial y} T_1^e + \dfrac{\partial N_2^e}{\partial y} T_2^e + \cdots + \dfrac{\partial N_{n_{\mathrm{en}}}^e}{\partial y} T_{n_{\mathrm{en}}}^e \end{bmatrix} = \begin{bmatrix} \dfrac{\partial N_1^e}{\partial x} & \dfrac{\partial N_2^e}{\partial x} & \cdots & \dfrac{\partial N_{n_{\mathrm{en}}}^e}{\partial x} \\ \dfrac{\partial N_1^e}{\partial y} & \dfrac{\partial N_2^e}{\partial y} & \cdots & \dfrac{\partial N_{n_{\mathrm{en}}}^e}{\partial y} \end{bmatrix} \mathbf{d}^e$$

In a more compact notation, the gradient becomes:

$$\nabla T^e(x,y) = (\nabla \mathbf{N}^e(x,y))\mathbf{d}^e = \mathbf{B}^e(x,y)\mathbf{d}^e = \mathbf{B}^e(x,y)\mathbf{L}^e\mathbf{d},$$

where,

$$\mathbf{B}^e(x,y) = \nabla \mathbf{N}^e(x,y).$$

The gradient operator for the weight function will give

$$(\nabla w^e)^{\mathrm{T}} = (\mathbf{B}^e\mathbf{w}^e)^{\mathrm{T}} = \mathbf{w}^{e\mathrm{T}}\mathbf{B}^{e\mathrm{T}} = (\mathbf{L}^e\mathbf{w})^{\mathrm{T}}\mathbf{B}^{e\mathrm{T}} = \mathbf{w}^{\mathrm{T}}\mathbf{L}^{e\mathrm{T}}\mathbf{B}^{e\mathrm{T}},$$

The degrees of freedom are partitioned as:

$$\mathbf{d} = \left\{ \begin{matrix} \overline{\mathbf{d}}_{\mathrm{E}} \\ \mathbf{d}_{\mathrm{F}} \end{matrix} \right\}, \qquad \mathbf{w} = \left\{ \begin{matrix} \mathbf{w}_{\mathrm{E}} \\ \mathbf{w}_{\mathrm{F}} \end{matrix} \right\} = \left\{ \begin{matrix} 0 \\ \mathbf{w}_{\mathrm{F}} \end{matrix} \right\}.$$

Here, E corresponds to the essential boundaries where the values of the field are known and the rest are given by F. Substituting all the values, we obtain

$$\mathbf{w}^{\mathrm{T}} \left\{ \sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \left[\int_{\Omega^e} \mathbf{B}^{e\mathrm{T}}\mathbf{D}^e\mathbf{B}^e \, \mathrm{d}\Omega \, \mathbf{L}^e\mathbf{d} + \int_{\Gamma_q^e} \mathbf{N}^{e\mathrm{T}}\overline{q} \, \mathrm{d}\Gamma - \int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} s \, \mathrm{d}\Omega \right] \right\} = 0 \qquad \forall \, \mathbf{w}_{\mathrm{F}}.$$

where the element conductance matrix is:

$$\boxed{\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^{e\mathrm{T}} \mathbf{D}^e \mathbf{B}^e \, \mathrm{d}\Omega.}$$

and the element flux matrix is:

$$\boxed{\mathbf{f}^e = \underbrace{-\int_{\Gamma_q^e} \mathbf{N}^{e\mathrm{T}} \overline{q} \, \mathrm{d}\Gamma}_{\mathbf{f}_\Gamma^e} + \underbrace{\int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} s \, \mathrm{d}\Omega}_{\mathbf{f}_\Omega^e},}$$

Here f_Γ^e and f_Ω^e are the boundary flux and the domain source terms, respectively. The weak form can then be written as:

$$\mathbf{w}^T \left[\left(\sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{K}^e \mathbf{L}^e \right) \mathbf{d} - \left(\sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{f}^e \right) \right] = 0 \qquad \forall \, \mathbf{w}$$

This could be rewritten as:

$$\mathbf{w}^\mathrm{T} \mathbf{r} = \mathbf{0} \qquad \forall \, \mathbf{w}_\mathrm{F},$$

where,

$$\mathbf{r} = \mathbf{K}\mathbf{d} - \mathbf{f},$$

and the global matrices are assembled as in 1D.

$$\mathbf{K} = \sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{K}^e \mathbf{L}^e, \qquad \mathbf{f} = \sum_{e=1}^{n_{\mathrm{el}}} \mathbf{L}^{e\mathrm{T}} \mathbf{f}^e.$$

The partition can be done as follows

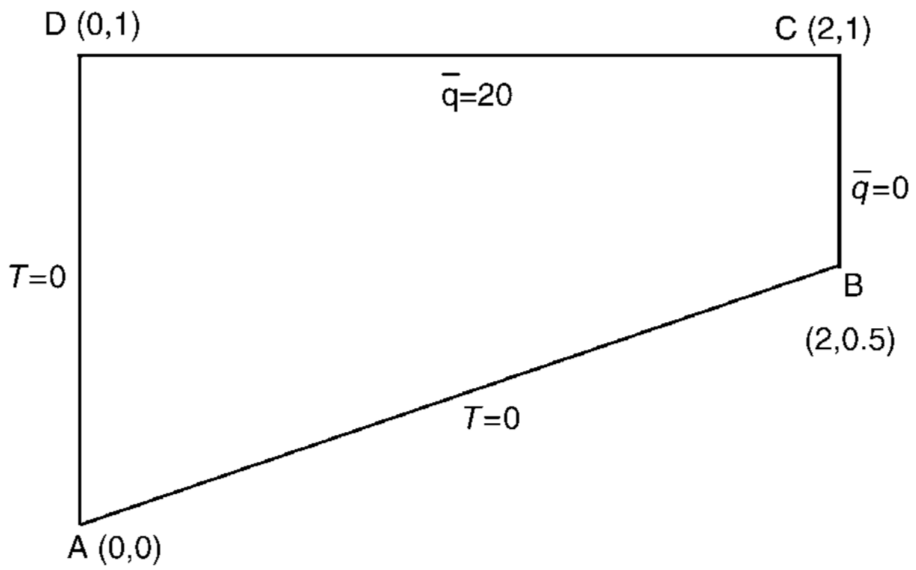
$$\mathbf{w}_\mathrm{F}^\mathrm{T} \mathbf{r}_\mathrm{F} + \mathbf{w}_\mathrm{E}^\mathrm{T} \mathbf{r}_\mathrm{E} = \mathbf{0} \qquad \forall \mathbf{w}_\mathrm{F}$$

However, since $w_E = 0$ at the essential boundaries, this reduces to $r_F = 0$. The final partitioned equations can be written as:

$$\mathbf{r} = \begin{bmatrix} \mathbf{r}_\mathrm{E} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{K}_\mathrm{E} & \mathbf{K}_\mathrm{EF} \\ \mathbf{K}_\mathrm{EF}^\mathrm{T} & \mathbf{K}_\mathrm{F} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{d}}_\mathrm{E} \\ \mathbf{d}_\mathrm{F} \end{bmatrix} - \begin{bmatrix} \mathbf{f}_\mathrm{E} \\ \mathbf{f}_\mathrm{F} \end{bmatrix},$$ which could be rewritten as:

$$\begin{bmatrix} \mathbf{K}_\mathrm{E} & \mathbf{K}_\mathrm{EF} \\ \mathbf{K}_\mathrm{EF}^\mathrm{T} & \mathbf{K}_\mathrm{F} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{d}}_\mathrm{E} \\ \mathbf{d}_\mathrm{F} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_\mathrm{E} + \mathbf{r}_\mathrm{E} \\ \mathbf{f}_\mathrm{F} \end{bmatrix}$$

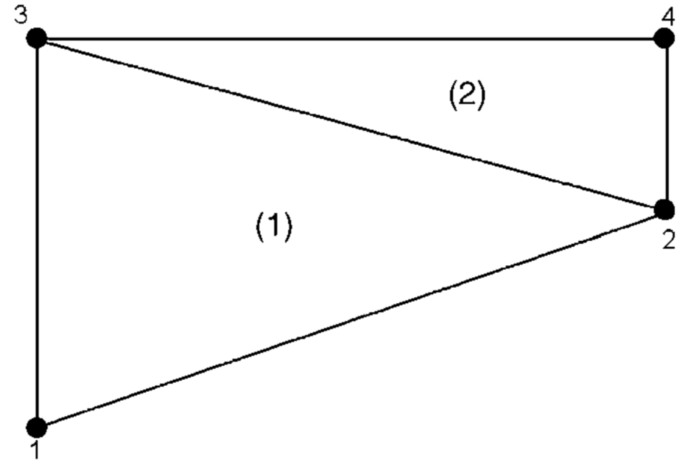
An example is provided as follows:



Problem:

Consider the heat conduction problem depicted in Figure 8.2. The coordinates are given in meters. The conductivity is isotropic, with $\mathbf{D} = k\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, and $k = 5\,\mathrm{W\,^\circ C^{-1}}$. The temperature $T = 0$ is prescribed along edges AB and AD. The heat fluxes $\bar{q} = 0$ and $\bar{q} = 20\,\mathrm{W\,m^{-1}}$ are prescribed on edges BC and CD, respectively. A constant heat source $s = 6\,\mathrm{Wm^{-2}}$ is applied over the plate.

The above problem can be broken into two elements:



The nodes at the intersection of essential and natural boundaries should satisfy the essential boundaries and should have lower numbering. The B_e for the matrix is given as:

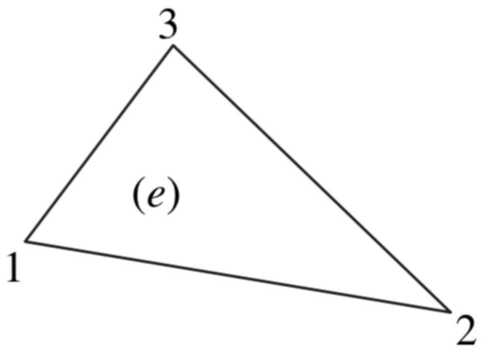
$$\mathbf{B}^e = \frac{1}{2A^e}\begin{bmatrix} (y_2^e - y_3^e) & (y_3^e - y_1^e) & (y_1^e - y_2^e) \\ (x_3^e - x_2^e) & (x_1^e - x_3^e) & (x_2^e - x_1^e) \end{bmatrix},$$

And the area of the element is given as:

$$2A^e = (x_2^e y_3^e - x_3^e y_2^e) - (x_1^e y_3^e - x_3^e y_1^e) + (x_1^e y_2^e - x_2^e$$

In the simple case where **D =** k **I,** the conductance matrix greatly simplifies.

$$\mathbf{K}^e = \int_{\Omega^e} \mathbf{B}^{eT}\mathbf{D}^e\mathbf{B}^e \, d\Omega = \int_{\Omega^e} \mathbf{B}^{eT}\mathbf{B}^e k \, d\Omega = \mathbf{B}^{eT}\mathbf{B}^e k \int_{\Omega^e} d\Omega$$



4 A counterclockwise numbering of element nodes.

Local node numbering and coordinates of element-1.

$$\mathbf{K}^{\mathrm{e}} = kA^{e}\mathbf{B}^{e\mathrm{T}}\mathbf{B}^{e}.$$

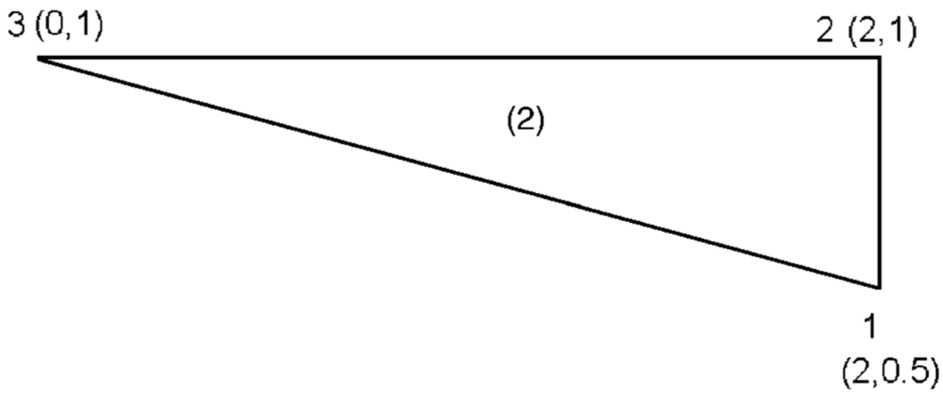
The area $A^{(1)} = 1$ and the resulting B matrix is:

$$\mathbf{B}^{(1)} = \frac{1}{2}\begin{bmatrix} -0.5 & 1 & -0.5 \\ -2 & 0 & 2 \end{bmatrix}.$$

Hence, the conductance is:

$$\mathbf{K}^{(1)} = kA^{(1)}\mathbf{B}^{(1)\mathrm{T}}\mathbf{B}^{(1)} = \begin{array}{c} \begin{bmatrix} 5.3125 & -0.625 & -4.6875 \\ -0.625 & 1.25 & -0.625 \\ -4.6875 & -0.625 & 5.3125 \end{bmatrix} \begin{array}{c} [1] \\ [2] \\ [3] \end{array} \\ \begin{array}{ccc} [1] & \qquad [2] & \qquad [3] \end{array} \end{array}$$

Similarly, for element 2 $A^{(2)} = 0.5$, and the conductance matrix is:



Local node numbering for element 2.
The conductance matrix for element 2 is

$$\mathbf{K}^{(2)} = kA^{(2)}\mathbf{B}^{(2)\mathrm{T}}\mathbf{B}^{(2)} = \begin{array}{c} \begin{bmatrix} 10 & -10 & 0 \\ -10 & 10.625 & -0.625 \\ 0 & -0.625 & 0.625 \end{bmatrix} \begin{array}{c} [2] \\ [4] \\ [3] \end{array} \\ \begin{array}{ccc} [2] & \qquad [3] & \qquad [4] \end{array} \end{array}.$$

The global conductance is given as

i

$$\mathbf{K} = \begin{array}{cc} \begin{bmatrix} 5.3125 & -0.625 & -4.6875 & 0 \\ -0.625 & 11.25 & -0.625 & -10 \\ -4.6875 & -0.625 & 5.9375 & -0.625 \\ 0 & -10 & -0.625 & 10.625 \end{bmatrix} & \begin{array}{c} {[1]} \\ {[2]} \\ {[3]} \\ {[4]} \end{array} \\ \begin{array}{cccc} {[1]} & \qquad {[2]} & \qquad {[3]} & \qquad {[4]} \end{array} & \end{array} .$$

Now, consider the source term:

$$\mathbf{f}_{\Omega}^{e} = \int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} s \, \mathrm{d}\Omega,$$

where the element shape functions are:

$$N_1^e = \frac{1}{2A^e}\left(x_2^e y_3^e - x_3^e y_2^e + (y_2^e - y_3^e)x + (x_3^e - x_2^e)y\right),$$
$$N_2^e = \frac{1}{2A^e}\left(x_3^e y_1^e - x_1^e y_3^e + (y_3^e - y_1^e)x + (x_1^e - x_3^e)y\right),$$
$$N_3^e = \frac{1}{2A^e}\left(x_1^e y_2^e - x_2^e y_1^e + (y_1^e - y_2^e)x + (x_2^e - x_1^e)y\right).$$

In the special case when the source terms is constant, using

$\int_{\Omega^e} N_I^e \, \mathrm{d}\Omega = A^e/3\,,$

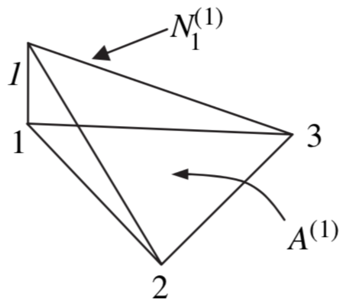
and s the flux terms from the source

$$\mathbf{f}_{\Omega}^{e} = s \int_{\Omega^e} \mathbf{N}^{e\mathrm{T}} \, \mathrm{d}\Omega = \frac{sA^e}{3}\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}.$$

$$\mathbf{f}_\Omega^{(1)} = \frac{sA^{(1)}}{3}\begin{bmatrix}1\\1\\1\end{bmatrix} = \frac{6\times 1}{3}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}2\\2\\2\end{bmatrix}\begin{matrix}[1]\\ [2]\\ [3]\end{matrix},$$

$$\mathbf{f}_\Omega^{(2)} = \frac{sA^{(2)}}{3}\begin{bmatrix}1\\1\\1\end{bmatrix} = \frac{6\times 0.5}{3}\begin{bmatrix}1\\1\\1\end{bmatrix} = \begin{bmatrix}1\\1\\1\end{bmatrix}\begin{matrix}[2]\\ [4]\\ [3]\end{matrix}.$$

The integral can be evaulated by noting that:



and the volume of this ‘’pyramid’’ is $\frac{1}{3}A^{(1)} \times 1$.
The direct assembly of the element source matrices

$$\mathbf{f}_\Omega = \begin{bmatrix}2\\2+1\\2+1\\1\end{bmatrix} = \begin{bmatrix}2\\3\\3\\1\end{bmatrix}\begin{matrix}[1]\\ [2]\\ [3]\\ [4]\end{matrix}.$$

To obtain the boundary flux we note that element-1 has two edges on the essential boundary condition where the temperature is specified and one interior edge, but no edge on the flux boundary condition. So only element 2 contributes to the boundary flux.

For element-2, q = 20 only on the top boundary. On the vertical boundary the flux is zero.
The boundary flux is obtained as follows:

$$\mathbf{f}_{\Gamma}^{e} = -\int_{\Gamma_q^e} \mathbf{N}^{e\mathrm{T}}\overline{q}\,\mathrm{d}\Gamma.$$

The shape function on the top boundary is evaluated as follows.

$$\mathbf{N}^{(2)}\Big|_{y=1} = \begin{bmatrix} \frac{1}{2A^{(2)}}\left[x_2^{(2)}y_3^{(2)} - x_3^{(2)}y_2^{(2)} + \left(y_2^{(2)} - y_3^{(2)}\right)x + \left(x_3^{(2)} - x_2^{(2)}\right)y\right] \\ \frac{1}{2A^{(2)}}\left[x_3^{(2)}y_1^{(2)} - x_1^{(2)}y_3^{(2)} + \left(y_3^{(2)} - y_1^{(2)}\right)x + \left(x_1^{(2)} - x_3^{(2)}\right)y\right] \\ \frac{1}{2A^{(2)}}\left[x_1^{(2)}y_2^{(2)} - x_2^{(2)}y_1^{(2)} + \left(y_1^{(2)} - y_2^{(2)}\right)x + \left(x_2^{(2)} - x_1^{(2)}\right)y\right] \end{bmatrix}_{y=1} = \begin{bmatrix} 0 \\ 0.5x \\ -0.5x + 1.0 \end{bmatrix}.$$

This value is then subsituted and integrated to give the boundary flux.

$$\mathbf{f}_{\Gamma}^{(2)} = -20\int_{x=0}^{x=2} \begin{bmatrix} 0 \\ 0.5x \\ -0.5x + 1 \end{bmatrix} \mathrm{d}x = \begin{bmatrix} 0 \\ -20 \\ -20 \end{bmatrix} \begin{matrix} [2] \\ [4] \\ [3] \end{matrix}.$$

The total boundary flux is then obtained as.

$$\mathbf{f}_{\Gamma} = \begin{bmatrix} 0 \\ 0 \\ -20 \\ -20 \end{bmatrix}.$$

Finally, the right hand side matrix of the matrix:

$$\begin{bmatrix} \mathbf{K}_{\mathrm{E}} & \mathbf{K}_{\mathrm{EF}} \\ \mathbf{K}_{\mathrm{EF}}^{\mathrm{T}} & \mathbf{K}_{\mathrm{F}} \end{bmatrix} \begin{bmatrix} \bar{\mathbf{d}}_{\mathrm{E}} \\ \mathbf{d}_{\mathrm{F}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_{\mathrm{E}} + \mathbf{r}_{\mathrm{E}} \\ \mathbf{f}_{\mathrm{F}} \end{bmatrix}$$

is assembled to give:

$$\mathbf{f}_{\Gamma} + \mathbf{f}_{\Omega} + \mathbf{r} = \begin{bmatrix} 2 \\ 3 \\ -17 \\ -19 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ 0 \end{bmatrix}.$$

The resulting system of equations is given as:

$$\begin{bmatrix} 5.3125 & -0.625 & -4.6875 & 0 \\ -0.625 & 11.25 & -0.625 & -10 \\ -4.6875 & -0.625 & 5.9375 & -0.625 \\ 0 & -10 & -0.625 & 10.625 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ T_4 \end{bmatrix} = \begin{bmatrix} r_1 + 2 \\ r_2 + 3 \\ r_3 - 17 \\ -19 \end{bmatrix}.$$

They can be solved to give:

$$T_4 = -19/10.625 = -1.788.$$

The resulting global and element temperature matrices are:

$$\mathbf{d} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -1.788 \end{bmatrix}, \qquad \mathbf{d}^{(1)} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \begin{matrix} [1] \\ [2] \\ [3] \end{matrix}, \qquad \mathbf{d}^{(2)} = \begin{bmatrix} 0 \\ -1.788 \\ 0 \end{bmatrix} \begin{matrix} [2] \\ [4] \\ [3] \end{matrix}$$

The flux matrices are:

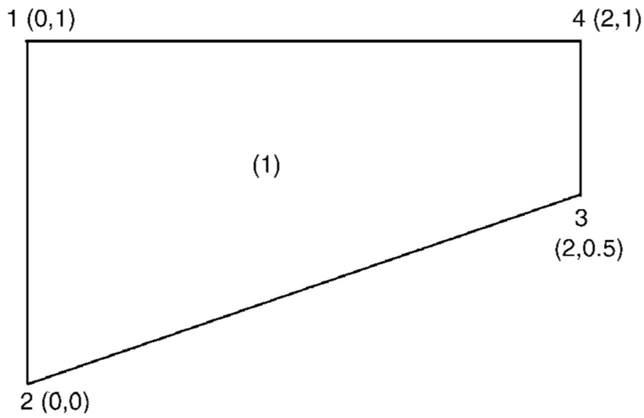
$$\mathbf{q}^{(1)} = -k\mathbf{IB}^{(1)}\mathbf{d}^{(1)} = -k\mathbf{B}^{(1)}\mathbf{d}^{(1)} = -5\frac{1}{2}\begin{bmatrix} -0.5 & 1 & -0.5 \\ -2 & 0 & 2 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

$$\mathbf{q}^{(2)} = -k\mathbf{B}^{(2)}\mathbf{d}^{(2)} = -5\begin{bmatrix} 0 & 0.5 & -0.5 \\ -2 & 2 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ -1.788 \\ 0 \end{bmatrix} = \begin{bmatrix} 4.47 \\ 17.88 \end{bmatrix}.$$

Note that the flux is obtained not for the node, but for the element. In general, we need to ‘project’ the flux from the element back to the nodes.

We will do the same problem by using quadratic element and isoparametric element formulation.

Solve the same problem using a single quadratic element. The integration is to be performed using the 2 × 2 Gauss quadrature developed earlier



Note, that the numbering is done in an anti-clockwise manner. The element coordinate matrix is:

$$[\mathbf{x}^e \quad \mathbf{y}^e] = \begin{bmatrix} x_1^e & y_1^e \\ x_2^e & y_2^e \\ x_3^e & y_3^e \\ x_4^e & y_4^e \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 2 & 0.5 \\ 2 & 1 \end{bmatrix}.$$

The shape functions for the four-node quadrilateral element are:

$$\begin{aligned}
N_1^{4Q}(\xi,\eta) &= \frac{\xi-\xi_2}{\xi_1-\xi_2}\frac{\eta-\eta_4}{\eta_1-\eta_4} = \frac{1}{4}(1-\xi)(1-\eta),\\
N_2^{4Q}(\xi,\eta) &= \frac{\xi-\xi_1}{\xi_2-\xi_1}\frac{\eta-\eta_4}{\eta_1-\eta_4} = \frac{1}{4}(1+\xi)(1-\eta),\\
N_3^{4Q}(\xi,\eta) &= \frac{\xi-\xi_1}{\xi_2-\xi_1}\frac{\eta-\eta_1}{\eta_4-\eta_1} = \frac{1}{4}(1+\xi)(1+\eta),\\
N_4^{4Q}(\xi,\eta) &= \frac{\xi-\xi_2}{\xi_1-\xi_2}\frac{\eta-\eta_1}{\eta_4-\eta_1} = \frac{1}{4}(1-\xi)(1+\eta).
\end{aligned}$$

The gradients in the **parent** domain are:

$$\mathbf{GN}^{4Q} = \begin{bmatrix} \dfrac{\partial N_1^{4Q}}{\partial \xi} & \dfrac{\partial N_2^{4Q}}{\partial \xi} & \dfrac{\partial N_3^{4Q}}{\partial \xi} & \dfrac{\partial N_4^{4Q}}{\partial \xi} \\ \dfrac{\partial N_1^{e}}{\partial \eta} & \dfrac{\partial N_2^{e}}{\partial \eta} & \dfrac{\partial N_3^{e}}{\partial \eta} & \dfrac{\partial N_4^{e}}{\partial \eta} \end{bmatrix} = \frac{1}{4}\begin{bmatrix} \eta-1 & 1-\eta & 1+\eta & -\eta-1 \\ \xi-1 & -\xi-1 & 1+\xi & 1-\xi \end{bmatrix}.$$

These gradients are converted from the parent to the physical or material domain via the **Jacobian** matrix which is obtained as:

$$\mathbf{J}^{(1)} = (\mathbf{GN}^{4Q})[\mathbf{x}^{(1)}\ \mathbf{y}^{(1)}] = \frac{1}{4}\begin{bmatrix} \eta-1 & 1-\eta & 1+\eta & -\eta-1 \\ \xi-1 & -\xi-1 & 1+\xi & 1-\xi \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 0 & 0 \\ 2 & 0.5 \\ 2 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0.125\eta-0.375 \\ 1 & 0.125\xi+0.125 \end{bmatrix},$$

$$\det \mathbf{J}^{(1)} \equiv \left|\mathbf{J}^{(1)}\right| = -0.125\eta + 0.375,$$

$$(\mathbf{J}^{(1)})^{-1} = \begin{bmatrix} \dfrac{1+\xi}{3-\eta} & 1 \\ \dfrac{8}{\eta-3} & 0 \end{bmatrix}.$$

Using this Jacobian matrix we can obtain the gradients in the material space or the global coordinates as

$$\mathbf{B}^{(1)} = (\mathbf{J}^{(1)})^{-1}(\mathbf{GN}^{4Q}) = (\mathbf{J}^{(1)})^{-1}\frac{1}{4}\begin{bmatrix} \eta-1 & 1-\eta & 1+\eta & -\eta-1 \\ \xi-1 & -\xi-1 & 1+\xi & 1-\xi \end{bmatrix}.$$

The **conductance** matrix is given as

$$\mathbf{K} = \mathbf{K}^{(1)} = \int_{\Omega} \mathbf{B}^{e\mathrm{T}}\mathbf{D}^e\mathbf{B}^e \, \mathrm{d}\Omega = k\int_{-1}^{1}\int_{-1}^{1} \mathbf{B}^{(1)\mathrm{T}}\mathbf{B}^{(1)}\left|\mathbf{J}^{(1)}\right| \mathrm{d}\xi\, \mathrm{d}\eta$$

Note that the matrix **K** above is not a polynomial. It will look as:

```
GN = 1/4 {{η - 1, 1 - η, 1 + η, -η - 1}, {ξ - 1, -ξ - 1, 1 + ξ, 1 - ξ}};
MatrixForm[%]
xy = {{0, 1}, {0, 0}, {2, 0.5}, {2, 1}};
MatrixForm[%]
(* Jacobian Matrix for the element *)
J = GN.xy // FullSimplify;
MatrixForm[%]
(* Determinant of the Jacobian *)
detJ = Det[J]
(* The B matrix is *)
B = Inverse[J].GN // FullSimplify;
MatrixForm[B]
(* The Integrand [kB^T B detJ] to obtain the K matrix *)
detJ * (Transpose[B].B) // FullSimplify;
MatrixForm[%]
```

$$\begin{pmatrix} \frac{1}{4}(-1+\eta) & \frac{1-\eta}{4} & \frac{1+\eta}{4} & \frac{1}{4}(-1-\eta) \\ \frac{1}{4}(-1+\xi) & \frac{1}{4}(-1-\xi) & \frac{1+\xi}{4} & \frac{1-\xi}{4} \end{pmatrix}$$

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \\ 2 & 0.5 \\ 2 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 0 & -0.375+0.125\,\eta \\ 1 & 0.125+0.125\,\xi \end{pmatrix}$$

$$0.375-0.125\,\eta$$

$$\begin{pmatrix} \frac{1.-0.5\,\eta-0.5\,\xi}{-3.+1.\,\eta} & \frac{0.5+0.5\,\xi}{-3.+1.\,\eta} & \frac{1.+1.\,\xi}{3.-1.\,\eta} & \frac{0.5-0.5\,\eta-1.\,\xi}{3.-1.\,\eta} \\ 2.-\frac{4.}{3.-1.\,\eta} & -2.+\frac{4.}{3.-1.\,\eta} & -\frac{1+\eta}{1.5-0.5\,\eta} & -2.+\frac{8.}{3.-1.\,\eta} \end{pmatrix}$$

$$\begin{pmatrix} \frac{-0.625-0.53125\,\eta^2+\eta\,(1.125-0.0625\,\xi)+(0.125-0.03125\,\xi)\,\xi}{-3.+1.\,\eta} & \frac{0.4375+0.5\,\eta^2+\eta\,(-0.96875+0.03125\,\xi)+(-0.03125+0.0}{-3.+1.\,\eta} \\ \frac{0.4375+0.5\,\eta^2+\eta\,(-0.96875+0.03125\,\xi)+(-0.03125+0.03125\,\xi)\,\xi}{-3.+1.\,\eta} & \frac{-0.53125+(1.-0.5\,\eta)\,\eta+(-0.0625-0.03125\,\xi)}{-3.+1.\,\eta} \\ \frac{0.625-0.5\,\eta^2+\eta\,(-0.0625-0.0625\,\xi)+(0.0625-0.0625\,\xi)\,\xi}{-3.+1.\,\eta} & \frac{-0.4375+0.5\,\eta^2+(0.125+0.0625\,\xi)\,\xi}{-3.+1.\,\eta} \\ \frac{-0.4375+0.53125\,\eta^2+\eta\,(-0.09375+0.09375\,\xi)+(-0.15625+0.0625\,\xi)\,\xi}{-3.+1.\,\eta} & \frac{0.53125-0.5\,\eta^2+\eta\,(-0.03125-0.03125\,\xi)+(-0.03125-0.}{-3.+1.\,\eta} \end{pmatrix}$$

This explicit calculation to only demonstrate that you may not encounter perfect polynomials. Else, you need to figure out what the Gauss points are and can evaluate individual terms at those points. In the current case using 2 × 2 Gauss quadrature with following sampling points and weights:

$$\xi_1 = -\frac{1}{\sqrt{3}}, \qquad \xi_2 = \frac{1}{\sqrt{3}}, \qquad \eta_1 = -\frac{1}{\sqrt{3}}, \qquad \eta_2 = \frac{1}{\sqrt{3}}, \qquad W_1 = W_2 = 1.$$

The conductance matrix becomes:

$$\mathbf{K} = \mathbf{K}^{(1)} = \int_\Omega \mathbf{B}^{e\mathrm{T}}\mathbf{D}^e\mathbf{B}^e \,\mathrm{d}\Omega = k\int_{-1}^{1}\int_{-1}^{1} \mathbf{B}^{(1)\mathrm{T}}\mathbf{B}^{(1)}\left|\mathbf{J}^{(1)}\right| \mathrm{d}\xi\,\mathrm{d}\eta$$

$$= k\sum_{i=1}^{2}\sum_{j=1}^{2} W_iW_j\left|\mathbf{J}^{(1)}(\xi_i,\eta_j)\right|\mathbf{B}^{(1)\mathrm{T}}(\xi_i,\eta_j)\mathbf{B}^{(1)}(\xi_i,\eta_j).$$

After plugging in the values, we get:

$$\mathbf{K}=\begin{bmatrix} 4.76 & -3.51 & -2.98 & 1.73 \\ -3.51 & 4.13 & 1.73 & -2.36 \\ -2.98 & 1.73 & 6.54 & -5.29 \\ 1.73 & -2.36 & -5.29 & 5.91 \end{bmatrix}.$$

For comparision, the exact Integral that is obtained analytically is:

```
Integrate[5 detJ * (Transpose[B].B), {ξ, -1, 1}, {η, -1, 1}] // MatrixForm
```

$$\begin{pmatrix} 4.77675 & -3.52675 & -2.94649 & 1.69649 \\ -3.52675 & 4.15175 & 1.69649 & -2.32149 \\ -2.94649 & 1.69649 & 6.60702 & -5.35702 \\ 1.69649 & -2.32149 & -5.35702 & 5.98202 \end{pmatrix}$$

Which is quite comparable with that obtained using the four point Gauss quadrature.

Now, to obtain the contribution from the source term to the element flux. The shape functions are:

$$N1 = \frac{1}{4} (1 - \xi) (1 - \eta);$$
$$N2 = \frac{1}{4} (1 + \xi) (1 - \eta);$$
$$N3 = \frac{1}{4} (1 + \xi) (1 + \eta);$$
$$N4 = \frac{1}{4} (1 - \xi) (1 + \eta);$$

Hence, the source matrix is simply:

$$\mathbf{f}_\Omega = \int_{\Omega^e} s(\mathbf{N}^{4Q})^{\mathrm{T}}\, \mathrm{d}\Omega = \int_{-1}^{1}\int_{-1}^{1} s(\mathbf{N}^{4Q})^{\mathrm{T}} \left|\mathbf{J}^{(1)}\right| \mathrm{d}\xi\, \mathrm{d}\eta$$

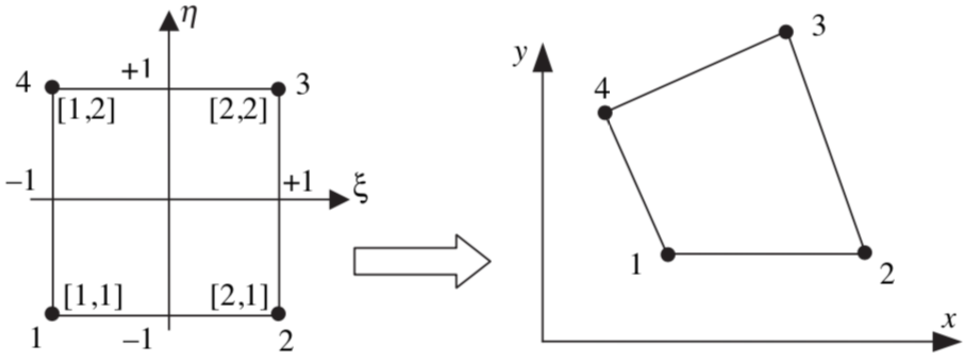
$$= \int_{-1}^{1}\int_{-1}^{1} 6 \begin{bmatrix} N_1^{4Q}(\xi,\eta) \\ N_2^{4Q}(\xi,\eta) \\ N_3^{4Q}(\xi,\eta) \\ N_4^{4Q}(\xi,\eta) \end{bmatrix} (-0.125\eta + 0.375)\, \mathrm{d}\xi\, \mathrm{d}\eta = \begin{bmatrix} 2.5 \\ 2.5 \\ 2 \\ 2 \end{bmatrix}.$$

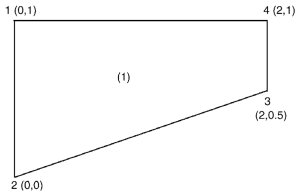
```
fΩ = Integrate[6 {N1, N2, N3, N4} detJ, {ξ, -1, 1}, {η, -1, 1}];
MatrixForm[%]
```

$$\begin{pmatrix} 2.5 \\ 2.5 \\ 2. \\ 2. \end{pmatrix}$$

Since the integrand above was a perfect polynomial in ξ, η with appropriate degree (2) in both, 2×2 quadrature was sufficient to get the exact answer. Finally, we need to obtain the flux from the top horizontal boundary.

The mapping is as follows:





If one goes from 1-4, it corresponds to positive η and from 1-2 correspond to positive ξ.

The boundary flux can be integrate analytically or using one-point gauss quadrature. On the top boundary 1-4, ξ = -1 and so:

$$\mathbf{f}_\Gamma = -\int_{\Gamma^{CD}} \bar{q}(\mathbf{N}^{4Q})^{\mathrm{T}}\,\mathrm{d}\Gamma = -\int_{x=0}^{x=2} \bar{q}\mathbf{N}^{4Q}(\xi=-1,\eta)^{\mathrm{T}}\,\mathrm{d}x$$

$$= -\underbrace{\frac{b-a}{2}}_{1}\bar{q}\int_{-1}^{1}\mathbf{N}^{4Q}(\xi=-1,\eta)^{\mathrm{T}}\,\mathrm{d}\eta = -20\int_{-1}^{1}\begin{bmatrix}\frac{1}{2}(1-\eta)\\ 0\\ 0\\ \frac{1}{2}(1+\eta)\end{bmatrix}\mathrm{d}\eta = \begin{bmatrix}-20\\ 0\\ 0\\ -20\end{bmatrix}.$$

The resulting RHS matrix is given by:

$$\mathbf{f}_\Omega + \mathbf{f}_\Gamma + \mathbf{r} = \begin{bmatrix} r_1 - 17.5\\ r_2 + 2.5\\ r_3 + 2\\ -18\end{bmatrix}.$$

The global system of equations is:

$$\begin{bmatrix} 4.76 & -3.51 & -2.98 & 1.73\\ -3.51 & 4.13 & 1.73 & -2.36\\ -2.98 & 1.73 & 6.54 & -5.29\\ 1.73 & -2.36 & -5.29 & 5.91\end{bmatrix}\begin{bmatrix}0\\0\\0\\T_4\end{bmatrix} = \begin{bmatrix} r_1 - 17.5\\ r_2 + 2.5\\ r_3 + 2\\ -18\end{bmatrix},$$

which yields $T_4 = -3.04$. The global temperature matrix is:

$$\mathbf{d} = \mathbf{d}^{(1)} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ T_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -3.04 \end{bmatrix}.$$

The resulting flux matrix is generally computed at the Gauss points and is given as:

$$\mathbf{q}^{(1)} = -k\nabla\theta = -k\mathbf{B}^{(1)}\mathbf{d}^{(1)},$$

$$\mathbf{q}^{(1)}(\xi_1, \eta_1) = -k\mathbf{B}^{(1)}(\xi_1, \eta_1)\mathbf{d}^{(1)} = \begin{bmatrix} 0.90 \\ 3.60 \end{bmatrix},$$

$$\mathbf{q}^{(1)}(\xi_2, \eta_2) = -k\mathbf{B}^{(1)}(\xi_2, \eta_2)\mathbf{d}^{(1)} = \begin{bmatrix} -2.3 \\ 19.8 \end{bmatrix},$$

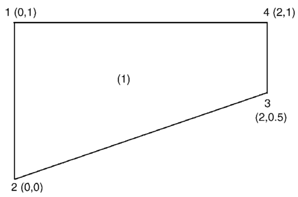
$$\mathbf{q}^{(1)}(\xi_3, \eta_3) = -k\mathbf{B}^{(1)}(\xi_3, \eta_3)\mathbf{d}^{(1)} = \begin{bmatrix} 4.95 \\ 19.8 \end{bmatrix},$$

$$\mathbf{q}^{(1)}(\xi_4, \eta_4) = -k\mathbf{B}^{(1)}(\xi_4, \eta_4)\mathbf{d}^{(1)} = \begin{bmatrix} 5.81 \\ 3.60 \end{bmatrix}.$$

Note, that unlike for the 3-node triangle element, the flux in this case is not constant.

As expected, the values that we get for the triangle and the quadrilateral element are not the same. However, they will converge to the real solution when the element size becomes small.

Side Note:



1) If the integral has to be performed on the inclined boundary 2-3, then 2-3 corresponds to $\xi = 1$ and positive η. For this element, it is immediately, clear that:
$d\Gamma = L_{23}\, d\eta$, where $L_{23} = \sqrt{2^2 + 0.5^2} \approx 2.06$.

2) If the integral was to be performed on 1-2, then it corresponded to +ve ξ. In that case, $\eta = -1$ and :
$d\Gamma = -dy = L_{12}\, d\xi$, and the limits of integration are from $\xi = -1$ to +1, and $L_{12} = 1$.

3) If the integral was to be performed on 3-4, then it corresponded to -ve ξ and $\eta = 1$. In this case:
$d\Gamma = dy = -L_{34}\, d\xi$, where $L_{34} = 0.5$. Now the integral will
$\int_\Gamma f\, d\Gamma = \int_0^{0.5} f\, dy = -\int_1^{-1} f\, L_{34}\, d\xi = \int_{-1}^{1} f\, L_{34}\, d\xi$.

The goal of this long exercise is to establish how to take the boundary integral. If the boundary is curved, then the relation between $d\Gamma$ and $d\xi$ is more complex.

$\sqrt{2^2 + 0.5^2}$

```
2.06155
```

A short detour: some side calculations:

Shape functions on the boundary $\xi = 1$.

$$n1 = \frac{\eta\ (\eta - 1)}{(-1 - 0)\ (-1 - 1)}$$
$$n2 = \frac{(\eta + 1)\ (\eta - 1)}{(0 + 1)\ (0 - 1)}$$
$$n3 = \frac{\eta\ (\eta + 1)}{(1 - 0)\ (1 + 1)}$$

$$\frac{1}{2}\ (-1 + \eta)\ \eta$$

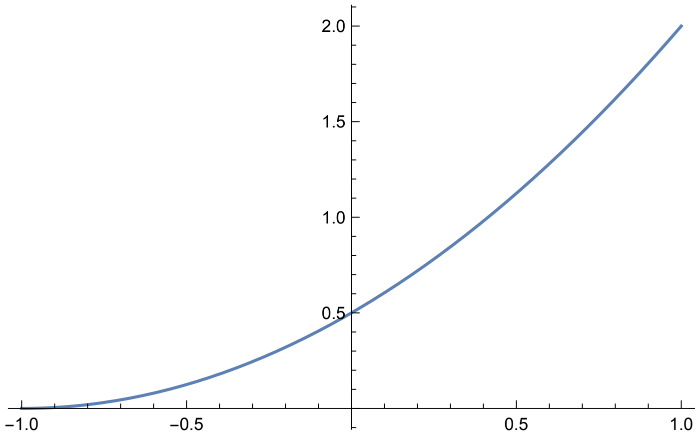
$$-(-1 + \eta)\ (1 + \eta)$$

$$\frac{1}{2}\ \eta\ (1 + \eta)$$

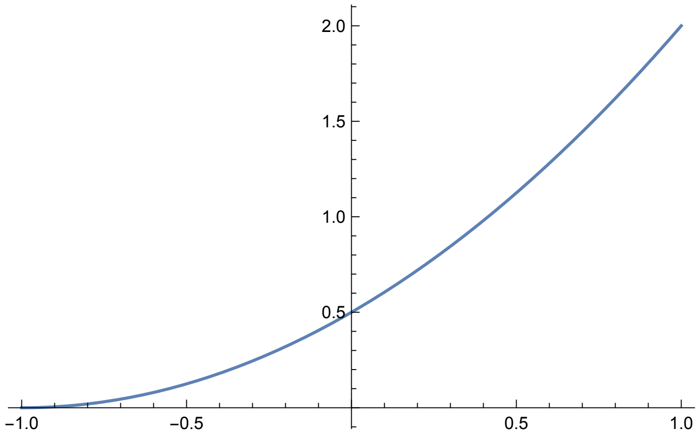
If the points on the edge are (0, 0), (0.5, 0.5), (2, 2), then the middle point is not exactly at the center of the edge. In that case, the mapping between the parent and the physical edge is non-linear.

```
x = n1 0 + n2 0.5 + n3 2 // Expand
Plot[x, {η, -1, 1}]
y = n1 0 + n2 0.5 + n3 2 // Expand
Plot[y, {η, -1, 1}]
```

$0.5 + \eta + 0.5\ \eta^2$

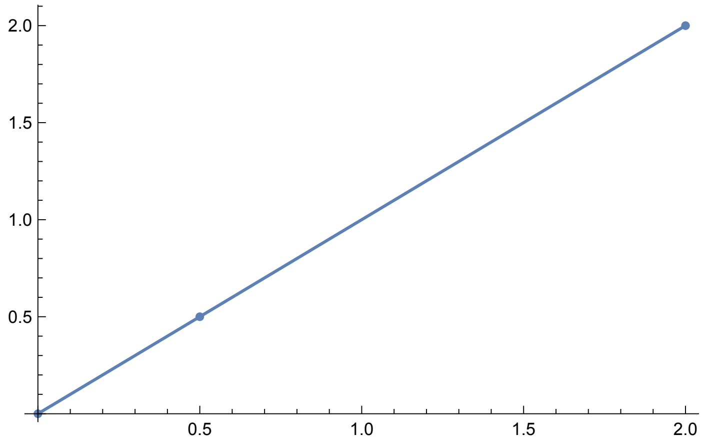


$0.5 + \eta + 0.5\ \eta^2$



If, we now plot how the edge looks like then:

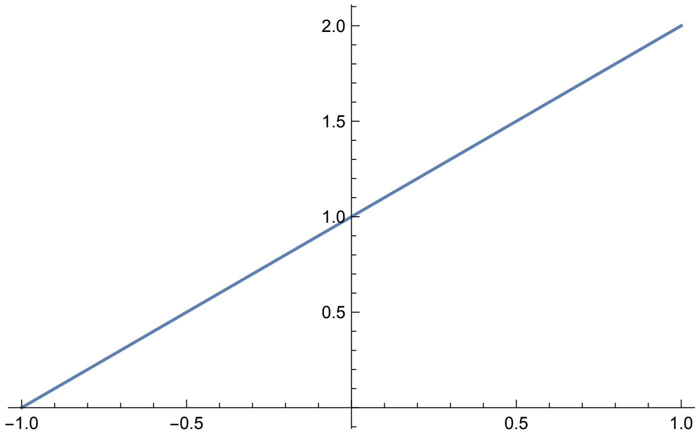
```
ParametricPlot[{x, y}, {η, -1, 1}];
ListPlot[{{x, y} /. η → -1, {x, y} /. η → 0, {x, y} /. η → 1}];
Show[%, %%]
```



If, however, the middle point is at the exact center of the edge, then, the middle point is (1, 1) and the mapping on the edge is:

```
x = n1 0 + n2  + n3 2 // Expand
Plot[x, {η, -1, 1}]
y = n1 0 + n2  + n3 2 // Expand
Plot[y, {η, -1, 1}]
```

$1 + \eta$



$1 + \eta$