Hat–Function

defined on a triangulation of the domain D

Hat–Function

defined on a triangulation of the domain D

- B_i, $i \in I$: basis for piecewise linear functions

Hat–Function

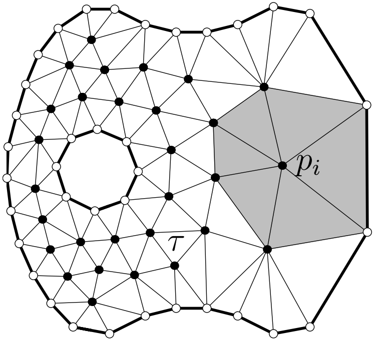
defined on a triangulation of the domain D

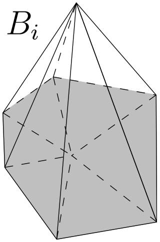
- B_i, $i \in I$: basis for piecewise linear functions
- $B_i(p_i) = 1$, 0 at other vertices p_j

Hat–Function

defined on a triangulation of the domain D

- B_i, $i \in I$: basis for piecewise linear functions
- $B_i(p_i) = 1$, 0 at other vertices p_j

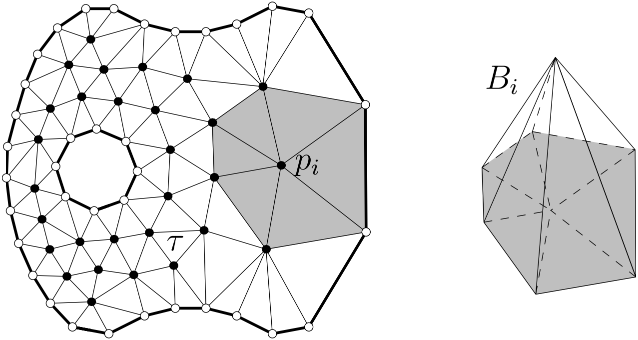




Hat–Function

defined on a triangulation of the domain D

- B_i, $i \in I$: basis for piecewise linear functions
- $B_i(p_i) = 1$, 0 at other vertices p_j



approximation, determined by Lagrange data

$$u_h = \sum_{i \in I} u_i B_i, \quad u_i = u_h(p_i)$$

Ritz-Galerkin System for Hat–Functions

system entries: sum contributions from each triangle $\tau = [p_i, p_j, p_k]$

$$g_{\ell,m} = \sum_{\tau} \int_{\tau} \operatorname{grad} B_\ell \, \operatorname{grad} B_m, \quad f_\ell = \sum_{\tau} \int_{\tau} f \, B_\ell$$

Ritz-Galerkin System for Hat–Functions

system entries: sum contributions from each triangle $\tau = [p_i, p_j, p_k]$

$$g_{\ell,m} = \sum_\tau \int_\tau \operatorname{grad} B_\ell \operatorname{grad} B_m, \quad f_\ell = \sum_\tau \int_\tau f\, B_\ell$$

compute gradients via directional derivatives

$$\underbrace{\begin{pmatrix} \operatorname{grad} B_i \\ \operatorname{grad} B_j \\ \operatorname{grad} B_k \end{pmatrix}}_{G_\tau} \begin{pmatrix} p_j - p_i & p_k - p_j \end{pmatrix} = R, \quad R = \begin{pmatrix} -1 & 0 \\ 1 & -1 \\ 0 & 1 \end{pmatrix}$$

G: add submatrix of

$$\text{area}(\tau)\, G_\tau G_\tau^t = \frac{|\det P|}{2}\, RP^{-1}(P^t)^{-1}R^t\,,$$

corresponding to inner vertices

G: add submatrix of

$$\operatorname{area}(\tau)\, G_\tau G_\tau^t = \frac{|\det P|}{2}\, R P^{-1} (P^t)^{-1} R^t\,,$$

corresponding to inner vertices

F: add subvector of

$$\frac{|\det P|}{6} \begin{pmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{pmatrix} \begin{pmatrix} f_i \\ f_j \\ f_k \end{pmatrix}$$

corresponding to inner vertices

Details

Details

exact Taylor approximation for linear functions

$$B(q) = B(p) + \underbrace{\operatorname{grad} B\,(q-p)}_{\text{directional derivative}}$$

Details

exact Taylor approximation for linear functions

$$B(q) = B(p) + \underbrace{\operatorname{grad} B\,(q-p)}_{\text{directional derivative}}$$

quadrature formula for triangle $\tau = [p_i, p_j, p_k]$

$$\int_\tau g = \frac{\operatorname{area}\tau}{3}\,\left[g((p_i+p_j)/2) + g((p_j+p_k)/2) + g((p_k+p_i)/2)\right]$$

Details

exact Taylor approximation for linear functions

$$B(q) = B(p) + \underbrace{\operatorname{grad} B\,(q-p)}_{\text{directional derivative}}$$

quadrature formula for triangle $\tau = [p_i, p_j, p_k]$

$$\int_\tau g = \frac{\operatorname{area}\tau}{3}\;[g((p_i+p_j)/2) + g((p_j+p_k)/2) + g((p_k+p_i)/2)]$$

exact for quadratic polynomials $\rightsquigarrow$ error $O(h^5)$, $h = \operatorname{diam}\tau$

Details

exact Taylor approximation for linear functions

$$B(q) = B(p) + \underbrace{\operatorname{grad} B\,(q-p)}_{\text{directional derivative}}$$

quadrature formula for triangle $\tau = [p_i, p_j, p_k]$

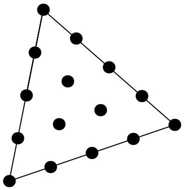
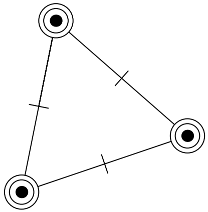
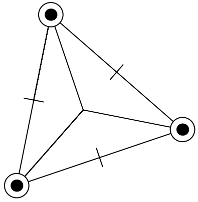
$$\int_\tau g = \frac{\operatorname{area}\tau}{3}\;[g((p_i+p_j)/2) + g((p_j+p_k)/2) + g((p_k+p_i)/2)]$$

exact for quadratic polynomials $\leadsto$ error $O(h^5)$, $h = \operatorname{diam}\tau$
apply with $g = fB_\ell$ and linear interpolation of f and $B_\ell \leadsto$

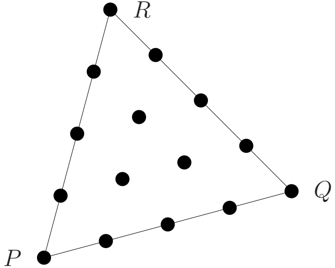
$$[\ldots] = f_\ell/2 + f_m/4 + f_{m'}/4, \quad m, m' \neq \ell\,,$$

since $g((p_\ell + p_m)/2) = ((f_\ell + f_m)/2)\,((1+0)/2)$

Bivariate Finite Elements

		
Lagrange	Argyris	Clough–Tocher
degree 4, $\in C^0$ dimension 15	degree 5, $\in C^1$ dimension 21	degree 3, $\in C^1$ dimension 12

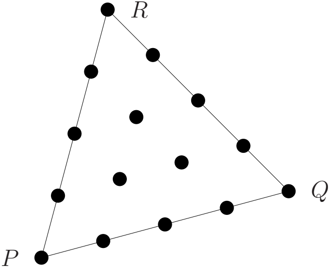
Lagrange Elements



nodes, labeled (i, j, k) with $i + j + k = n$

$$\frac{i}{n}P + \frac{j}{n}Q + \frac{k}{n}R$$

Lagrange Elements



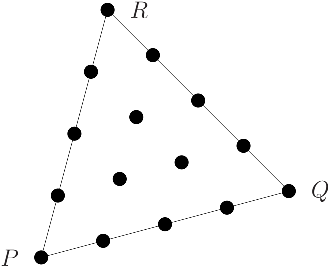
nodes, labeled (i, j, k) with $i + j + k = n$

$$\frac{i}{n}P + \frac{j}{n}Q + \frac{k}{n}R$$

basis functions

$$B_{i,j,k} = \binom{n}{i} u^0 \cdots u^{i-1} \binom{n}{j} v^0 \cdots v^{j-1} \binom{n}{k} w^0 \cdots w^{k-1}$$

Lagrange Elements



nodes, labeled (i, j, k) with $i + j + k = n$

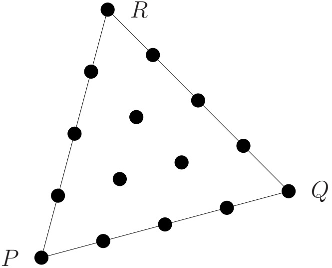
$$\frac{i}{n}P + \frac{j}{n}Q + \frac{k}{n}R$$

basis functions

$$B_{i,j,k} = \binom{n}{i} u^0 \cdots u^{i-1} \binom{n}{j} v^0 \cdots v^{j-1} \binom{n}{k} w^0 \cdots w^{k-1}$$

u^ℓ linear, with $u^\ell(P) = 1$ and $u^\ell = 0$ on nodes (ℓ, α, β)
(v^ℓ, w^ℓ defined similarly)

Lagrange Elements



nodes, labeled (i, j, k) with $i + j + k = n$

$$\frac{i}{n}P + \frac{j}{n}Q + \frac{k}{n}R$$

basis functions

$$B_{i,j,k} = \binom{n}{i} u^0 \cdots u^{i-1} \binom{n}{j} v^0 \cdots v^{j-1} \binom{n}{k} w^0 \cdots w^{k-1}$$

u^ℓ linear, with $u^\ell(P) = 1$ and $u^\ell = 0$ on nodes (ℓ, α, β)
(v^ℓ, w^ℓ defined similarly)
$\rightsquigarrow$ interpolation of Lagrange data

Details

node

$$X = \frac{i}{n} P + \frac{j}{n} Q + \frac{k}{n} R$$

Details

node

$$X = \frac{i}{n} P + \frac{j}{n} Q + \frac{k}{n} R$$

$u^\ell((n/n)P) = 1,\ u^\ell((\ell/n)P + \cdots) = 0 \implies$

$$u^\ell\left(\frac{i}{n} P + \cdots\right) = \frac{i - \ell}{n - \ell}$$

(view i as variable, ranging from ℓ to n)

Details

node

$$X = \frac{i}{n} P + \frac{j}{n} Q + \frac{k}{n} R$$

$u^\ell((n/n)P) = 1,\ u^\ell((\ell/n)P + \cdots) = 0 \implies$

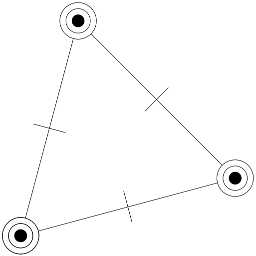
$$u^\ell\left(\frac{i}{n} P + \cdots\right) = \frac{i-\ell}{n-\ell}$$

(view i as variable, ranging from ℓ to n)
value of $u^0 u^1 \cdots u^{i-1}$ at X

$$\frac{i}{n} \frac{i-1}{n-1} \cdots \frac{1}{n-i+1} = \binom{n}{i}^{-1}$$

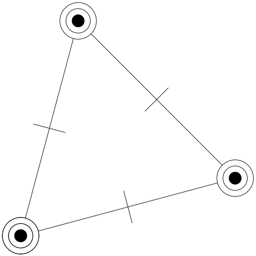
$\implies B_{i,j,k}(X) = 1$

Argyris Triangle



degree 5, $\in C^1$
dimension 21

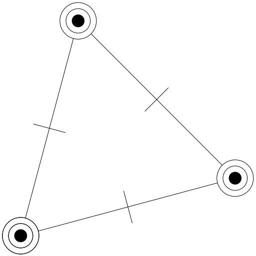
Argyris Triangle



degree 5, $\in C^1$
dimension 21

defining data

Argyris Triangle

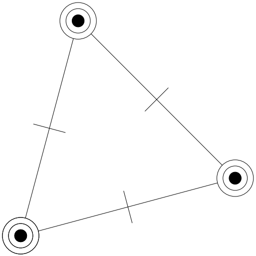


degree 5, $\in C^1$
dimension 21

defining data

- partial derivatives of order ≤ 2 at vertices

Argyris Triangle

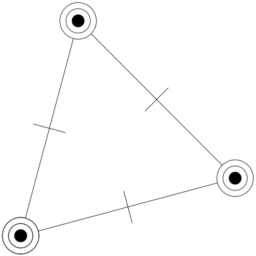


degree 5, $\in C^1$
dimension 21

defining data

- partial derivatives of order ≤ 2 at vertices
- normal derivatives at edge mid-points

Argyris Triangle



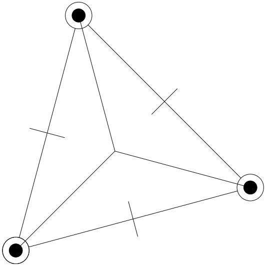
degree 5, $\in C^1$
dimension 21

defining data

- partial derivatives of order ≤ 2 at vertices
- normal derivatives at edge mid-points

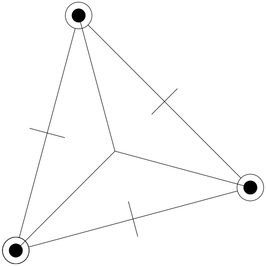
$\Longrightarrow$ values and derivatives prescribed at triangle boundaries
(quintic and quartic polynomials, respectively)

Clough-Tocher Macro-Triangle



degree 3, $\in C^1$
dimension 12

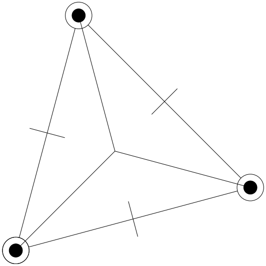
Clough-Tocher Macro-Triangle



degree 3, $\in C^1$
dimension 12

defining data

Clough-Tocher Macro-Triangle

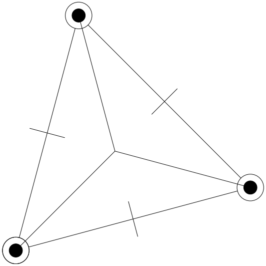


degree 3, $\in C^1$
dimension 12

defining data

- values and gradients at vertices

Clough-Tocher Macro-Triangle

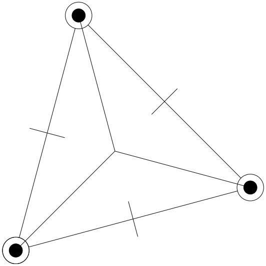


degree 3, $\in C^1$
dimension 12

defining data

- values and gradients at vertices
- normal derivatives at edge mid-points

Clough-Tocher Macro-Triangle



degree 3, $\in C^1$
dimension 12

defining data

- values and gradients at vertices
- normal derivatives at edge mid-points

$\Longrightarrow$ values and derivatives prescribed along the outer boundaries (cubic and quadratic polynomials, respectively)

Properties of Finite Elements

The basis functions B_i of standard mesh-based finite element subspaces are piecewise polynomials of degree $\leq n$ with support on few neighboring mesh cells. They are at least continuous and compatible with homogeneous boundary conditions on piecewise linear boundaries.

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.
- Using quadratic or higher degree leads to excessively large systems.

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.
- Using quadratic or higher degree leads to excessively large systems.
- Only moderately accurate approximations are possible.

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.
- Using quadratic or higher degree leads to excessively large systems.
- Only moderately accurate approximations are possible.
- Boundary conditions are merely approximated for general free-form domains.

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.
- Using quadratic or higher degree leads to excessively large systems.
- Only moderately accurate approximations are possible.
- Boundary conditions are merely approximated for general free-form domains.

Weighted spline-based finite elements overcome these difficulties:

Disadvantages of Standard Mesh-Based Finite Elements

principal drawbacks

- Generating good meshes can be difficult, time-consuming, and might require user interaction.
- Using quadratic or higher degree leads to excessively large systems.
- Only moderately accurate approximations are possible.
- Boundary conditions are merely approximated for general free-form domains.

Weighted spline-based finite elements overcome these difficulties:
No mesh generation is required, accurate smooth approximations are possible with relatively low-dimensional finite element subspaces, and boundary conditions are satisfied exactly.