Last time – Finite Elements, part 1

Last time – Finite Elements, part 1

1. Representation

$$f = \sum f_i \phi_i(x)$$

e.g. ϕ_i linear over $\triangle$ (localised)

Last time – Finite Elements, part 1

1. Representation

$$f = \sum f_i \phi_i(x)$$

e.g. ϕ_i linear over $\triangle$ (localised)

2. Variational Statement

$$\nabla^2 f = \rho \qquad \equiv \qquad \delta I = 0, \quad I = \int \tfrac{1}{2} |\nabla f|^2 + \rho f$$

so

$$K_{ij} f_j = r_i$$

with

$$K_{ij} = \int \nabla \phi_i \cdot \nabla \phi_j \qquad r_i = \int \rho \phi_i$$

Last time – Finite Elements, part 1

1. Representation

$$f = \sum f_i \phi_i(x)$$

e.g. ϕ_i linear over $\triangle$ (localised)

2. Variational Statement

$$\nabla^2 f = \rho \qquad \equiv \qquad \delta I = 0, \quad I = \int \tfrac{1}{2}|\nabla f|^2 + \rho f$$

so

$$K_{ij} f_j = r_i$$

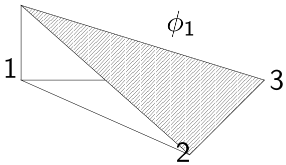
with

$$K_{ij} = \int \nabla\phi_i \cdot \nabla\phi_j \qquad r_i = \int \rho\phi_i$$

This time – Finite Elements, part 2

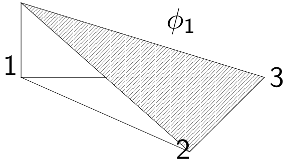
Details in 2D with linear triangular elements

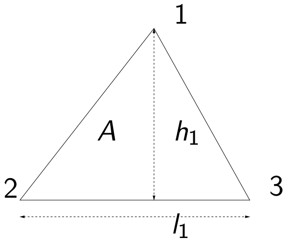
Consider one triangle
$|\nabla\phi_1| = 1/h_1$



Details in 2D with linear triangular elements

Consider one triangle $|\nabla \phi_1| = 1/h_1$

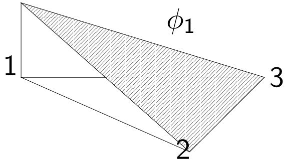


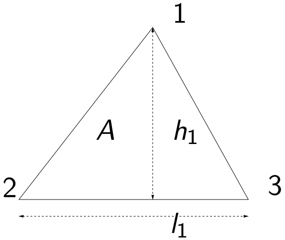


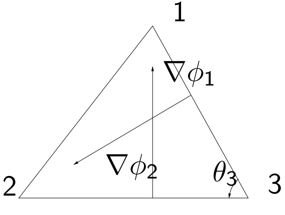
$$K_{11} = \int \nabla \phi_1 \cdot \nabla \phi_1 = \frac{A}{h_1^2}$$

Details in 2D with linear triangular elements

Consider one triangle $|\nabla\phi_1| = 1/h_1$







$$K_{11} = \int \nabla\phi_1 \cdot \nabla\phi_1 = \frac{A}{h_1^2} \quad K_{12} = \int \nabla\phi_1 \cdot \nabla\phi_2 = -\frac{A\cos\theta_3}{h_1 h_2}$$

further manipulations

$$h_1 = \ell_2 \sin\theta_3 \quad \text{and} \quad h_2 = \ell_1 \sin\theta_3.$$

and

$$A = \tfrac{1}{2}\ell_1\ell_2 \sin\theta_3.$$

further manipulations

$$h_1 = \ell_2 \sin\theta_3 \quad \text{and} \quad h_2 = \ell_1 \sin\theta_3.$$

and

$$A = \tfrac{1}{2}\ell_1\ell_2 \sin\theta_3.$$

Hence

$$K_{12} = -\frac{\cos\theta_3 A}{h_1 h_2} = -\tfrac{1}{2}\cot\theta_3.$$

$$\ell_1 = h_1 \cot\theta_3 + h_1 \cot\theta_2.$$

further manipulations

$$h_1 = \ell_2 \sin\theta_3 \quad \text{and} \quad h_2 = \ell_1 \sin\theta_3.$$

and

$$A = \tfrac{1}{2}\ell_1\ell_2 \sin\theta_3.$$

Hence

$$K_{12} = -\frac{\cos\theta_3 A}{h_1 h_2} = -\tfrac{1}{2}\cot\theta_3.$$

$$\ell_1 = h_1 \cot\theta_3 + h_1 \cot\theta_2.$$

Hence

$$K_{11} = \frac{A}{h_1^2} = \tfrac{1}{2}\left(\cot\theta_3 + \cot\theta_2\right).$$

further manipulations

$$h_1 = \ell_2 \sin\theta_3 \quad \text{and} \quad h_2 = \ell_1 \sin\theta_3.$$

and

$$A = \tfrac{1}{2}\ell_1\ell_2 \sin\theta_3.$$

Hence

$$K_{12} = -\frac{\cos\theta_3 A}{h_1 h_2} = -\tfrac{1}{2}\cot\theta_3.$$

$$\ell_1 = h_1 \cot\theta_3 + h_1 \cot\theta_2.$$

Hence

$$K_{11} = \frac{A}{h_1^2} = \tfrac{1}{2}\left(\cot\theta_3 + \cot\theta_2\right).$$

Note

$$K_{11} + K_{12} + K_{13} = 0.$$

further manipulations

$$h_1 = \ell_2 \sin\theta_3 \quad \text{and} \quad h_2 = \ell_1 \sin\theta_3.$$

and

$$A = \tfrac{1}{2}\ell_1\ell_2 \sin\theta_3.$$

Hence

$$K_{12} = -\frac{\cos\theta_3 A}{h_1 h_2} = -\tfrac{1}{2}\cot\theta_3.$$

$$\ell_1 = h_1 \cot\theta_3 + h_1 \cot\theta_2.$$

Hence

$$K_{11} = \frac{A}{h_1^2} = \tfrac{1}{2}\left(\cot\theta_3 + \cot\theta_2\right).$$

Note

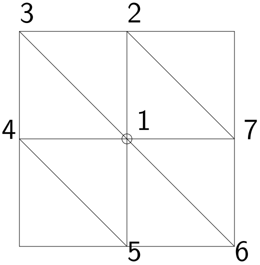
$$K_{11} + K_{12} + K_{13} = 0.$$

Because

$$\nabla\phi_1 \cdot \nabla\left(\phi_1 + \phi_2 + \phi_3 \equiv 1\right) \equiv 0.$$

Assembling contributions from different triangles

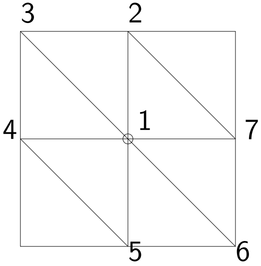
Special grid:



For the 123-triangle,

Assembling contributions from different triangles

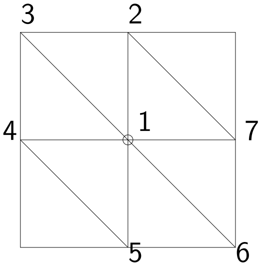
Special grid:



For the 123-triangle, $\theta_2 = \frac{\pi}{2}$ so $K_{13} = 0$

Assembling contributions from different triangles

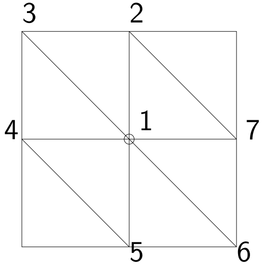
Special grid:



For the 123-triangle, $\theta_2 = \frac{\pi}{2}$ so $K_{13} = 0$
$\theta_3 = \frac{\pi}{4}$ so $K_{12} = -\frac{1}{2}$,

Assembling contributions from different triangles

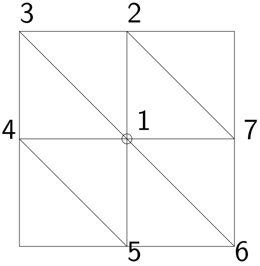
Special grid:



For the 123-triangle, $\theta_2 = \frac{\pi}{2}$ so $K_{13} = 0$
$\theta_3 = \frac{\pi}{4}$ so $K_{12} = -\frac{1}{2}$, and so contribution to K_{11} is $\frac{1}{2}$

Assembling contributions from different triangles

Special grid:

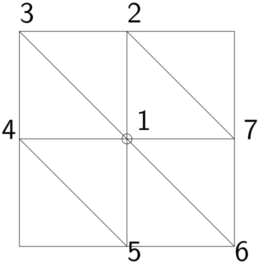


For the 123-triangle, $\theta_2 = \frac{\pi}{2}$ so $K_{13} = 0$
$\theta_3 = \frac{\pi}{4}$ so $K_{12} = -\frac{1}{2}$, and so contribution to K_{11} is $\frac{1}{2}$

For the 172-triangle, $K_{17} = K_{12} = -\frac{1}{2}\cot\frac{\pi}{4} = -\frac{1}{2}$,
and so contribution to K_{11} is 1

Assembling contributions from different triangles

Special grid:



For the 123-triangle, $\theta_2 = \frac{\pi}{2}$ so $K_{13} = 0$
$\theta_3 = \frac{\pi}{4}$ so $K_{12} = -\frac{1}{2}$, and so contribution to K_{11} is $\frac{1}{2}$

For the 172-triangle, $K_{17} = K_{12} = -\frac{1}{2}\cot\frac{\pi}{4} = -\frac{1}{2}$,
and so contribution to K_{11} is 1

Assembling from all triangles

$$K_{11} = 4, \quad K_{12} = K_{14} = K_{15} = K_{17} = -1, \quad K_{13} = K_{16} = 0.$$

Forcing

$$r_i = \int \rho \phi_i = \frac{1}{3} A \rho_i$$

by linear variation of ϕ_i on each of the 6 triangle (so $A = 3h^2$).

Forcing

$$r_i = \int \rho\phi_i = \frac{1}{3}A\rho_i$$

by linear variation of ϕ_i on each of the 6 triangle (so $A = 3h^2$).

Hence FE Poisson problem on special grid is

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} f + h^2\rho_i = 0,$$

Forcing

$$r_i = \int \rho \phi_i = \frac{1}{3} A \rho_i$$

by linear variation of ϕ_i on each of the 6 triangle (so $A = 3h^2$).

Hence FE Poisson problem on special grid is

$$\begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} f + h^2 \rho_i = 0,$$

identical to FD!

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)
- List of triangles: T as vertices P_T, Q_T, R_T

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)
- List of triangles: T as vertices P_T, Q_T, R_T
- Inverse list of triangles containing given vertex

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)
- List of triangles: T as vertices P_T, Q_T, R_T
- Inverse list of triangles containing given vertex
-    or list of neighbouring points
-    or list of edges E joining points P_E, Q_E

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)
- List of triangles: T as vertices P_T, Q_T, R_T
- Inverse list of triangles containing given vertex
- or list of neighbouring points
- or list of edges E joining points P_E, Q_E
- Search to find triangle containing given point (x, y)

Programming challenge

On more general unstructured grids need

- List of points: P at (x_P, y_P)
- List of triangles: T as vertices P_T, Q_T, R_T
- Inverse list of triangles containing given vertex
- or list of neighbouring points
- or list of edges E joining points P_E, Q_E
- Search to find triangle containing given point (x, y)

Can use list of triangles to assemble sparse matrix K_{ij}.

Galerkin formulation or "Weak" formulation

Not all physics is $\delta(\mathrm{Action}) = 0$.

Galerkin formulation or “Weak” formulation

Not all physics is $\delta(\mathrm{Action}) = 0$.
Consider nonlinear pde for $u(\mathbf{x}, t)$

$$A(u) = f$$

Galerkin formulation or "Weak" formulation

Not all physics is $\delta(\mathrm{Action}) = 0$.
Consider nonlinear pde for $u(\mathbf{x}, t)$

$$A(u) = f$$

Use Finite Element representation

$$u(\mathbf{x}, t) = \sum^{N} u_i(t)\phi_i(\mathbf{x})$$

Galerkin formulation or "Weak" formulation

Not all physics is $\delta(\mathrm{Action}) = 0$.
Consider nonlinear pde for $u(\mathbf{x}, t)$

$$A(u) = f$$

Use Finite Element representation

$$u(\mathbf{x}, t) = \sum^{N} u_i(t)\phi_i(\mathbf{x})$$

For vector space of basis functions, define inner product

$$\langle a, b \rangle = \int a(\mathbf{x}) b(\mathbf{x}) dV$$

Galerkin formulation or "Weak" formulation

Not all physics is $\delta(\mathrm{Action}) = 0$.
Consider nonlinear pde for $u(\mathbf{x}, t)$

$$A(u) = f$$

Use Finite Element representation

$$u(\mathbf{x}, t) = \sum^{N} u_i(t)\phi_i(\mathbf{x})$$

For vector space of basis functions, define inner product

$$\langle a, b \rangle = \int a(\mathbf{x})b(\mathbf{x})dV$$

Require residual to be OG all N basis functions

$$\langle A(u) - f, \phi_j \rangle = 0 \quad \text{all } j$$

Automatic conservative property

$$\langle A(u) - f, \phi_j \rangle = 0 \quad \text{all } j$$

Automatic conservative property

$$\langle A(u) - f, \phi_j \rangle = 0 \quad \text{all } j$$

Now (with possible selection)

$$\sum \phi \equiv 1$$

Automatic conservative property

$$\langle A(u) - f, \phi_j \rangle = 0 \quad \text{all } j$$

Now (with possible selection)

$$\sum \phi \equiv 1$$

So

$$\langle A(u) - f, 1 \rangle = 0$$

Automatic conservative property

$$\langle A(u) - f, \phi_j \rangle = 0 \quad \text{all } j$$

Now (with possible selection)

$$\sum \phi \equiv 1$$

So

$$\langle A(u) - f, 1 \rangle = 0$$

i.e.

$$\int A(u) = \int f$$

Eg diffusion equation

part of Navier-Stokes

$$u_t = \nabla^2 u$$

Eg diffusion equation

part of Navier-Stokes

$$u_t = \nabla^2 u$$

Galerkin after integration by parts

$$\langle u_t, \phi_j \rangle = -\langle \nabla u, \nabla \phi_j \rangle \quad \text{all } j$$

Eg diffusion equation

part of Navier-Stokes

$$u_t = \nabla^2 u$$

Galerkin after integration by parts

$$\langle u_t, \phi_j \rangle = -\langle \nabla u, \nabla \phi_j \rangle \quad \text{all } j$$

Substitute FE representation $u = \sum u_i(t)\phi_i(x)$

$$\sum \dot{u}_i \langle \phi_i, \phi_j \rangle = -\sum u_i \langle \nabla \phi_i, \nabla \phi_j \rangle$$

Eg diffusion equation

part of Navier-Stokes

$$u_t = \nabla^2 u$$

Galerkin after integration by parts

$$\langle u_t, \phi_j \rangle = -\langle \nabla u, \nabla \phi_j \rangle \quad \text{all } j$$

Substitute FE representation $u = \sum u_i(t)\phi_i(x)$

$$\sum \dot{u}_i \langle \phi_i, \phi_j \rangle = -\sum u_i \langle \nabla \phi_i, \nabla \phi_j \rangle$$

i.e.

$$M_{ij}\dot{u}_i = -K_{ij}u_i$$

with 'Mass' $\quad M_{ij} = \langle \phi_i, \phi_j \rangle \quad$ and 'Stiffness' $\quad K_{ij} = \langle \nabla \phi_i, \nabla \phi_j \rangle$

b. In 1D

Using linear elements on equal intervals h

b. In 1D

Using linear elements on equal intervals h

$$M_{ij} = \begin{cases} \frac{2}{3}h & i = j \\ \frac{1}{6}h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad K_{ij} = \begin{cases} 2/h & i = j \\ -1/h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

b. In 1D

Using linear elements on equal intervals h

$$M_{ij} = \begin{cases} \frac{2}{3}h & i = j \\ \frac{1}{6}h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad K_{ij} = \begin{cases} 2/h & i = j \\ -1/h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

Hence

$$h\left(\tfrac{1}{6}\dot{u}_{i-1} + \tfrac{2}{3}\dot{u}_i + \tfrac{1}{6}\dot{u}_{i+1}\right) = \frac{1}{h}\left(u_{i-1} - 2u_i + u_{i+1}\right).$$

b. In 1D

Using linear elements on equal intervals h

$$M_{ij} = \begin{cases} \frac{2}{3}h & i = j \\ \frac{1}{6}h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad K_{ij} = \begin{cases} 2/h & i = j \\ -1/h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

Hence

$$h\left(\tfrac{1}{6}\dot{u}_{i-1} + \tfrac{2}{3}\dot{u}_i + \tfrac{1}{6}\dot{u}_{i+1}\right) = \frac{1}{h}\left(u_{i-1} - 2u_i + u_{i+1}\right).$$

Remark Linear algebra to find $\dot{u}_i$ – tridiagonal matrix fast to invert

b. In 1D

Using linear elements on equal intervals h

$$M_{ij} = \begin{cases} \frac{2}{3}h & i = j \\ \frac{1}{6}h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases} \quad \text{and} \quad K_{ij} = \begin{cases} 2/h & i = j \\ -1/h & i = j \pm 1 \\ 0 & \text{otherwise} \end{cases}$$

Hence

$$h\left(\tfrac{1}{6}\dot{u}_{i-1} + \tfrac{2}{3}\dot{u}_i + \tfrac{1}{6}\dot{u}_{i+1}\right) = \frac{1}{h}\left(u_{i-1} - 2u_i + u_{i+1}\right).$$

Remark Linear algebra to find $\dot{u}_i$ – tridiagonal matrix fast to invert

Remark Time step this “semi-discretised” form with any FD (NOT FE) algorithm, e.g.

$$u_i^{n+1} = u_i^n + \Delta t \dot{u}_i$$

c. In 2D

Use linear triangular elements on special grid.

c. In 2D

Use linear triangular elements on special grid.
Assemble contributions to M and K from different triangles

$$M_{ij} = \begin{cases} \frac{1}{12} h^2 & i = j \\ \frac{1}{24} h^2 & i \neq j, \end{cases} \qquad K_{ij} \quad \text{as before}$$

c. In 2D

Use linear triangular elements on special grid.
Assemble contributions to M and K from different triangles

$$M_{ij} = \begin{cases} \frac{1}{12}h^2 & i = j \\ \frac{1}{24}h^2 & i \neq j, \end{cases} \qquad K_{ij} \quad \text{as before}$$

So

$$\frac{1}{2}h^2\left(\dot{u}_1 + \frac{1}{6}(\dot{u}_2 + \dot{u}_3 + \dot{u}_4 + \dot{u}_5 + \dot{u}_6 + \dot{u}_7)\right) = u_2 + u_4 + u_5 + u_7 - 4u_1$$

with linear problem to find $\dot{u}_i$

Navier-Stokes

Navier-Stokes

a. Weak formulation

Use FE representation

$$\mathbf{u}(\mathbf{x},t) = \sum_i \mathbf{u}_i(t)\phi_i(\mathbf{x}),$$
$$p(\mathbf{x},t) = \sum_i p_i(t)\psi_i(\mathbf{x}),$$

Need different ϕ_i and ψ_i.

Navier-Stokes

a. Weak formulation

Use FE representation

$$\begin{aligned}\mathbf{u}(\mathbf{x},t) &= \sum_i \mathbf{u}_i(t)\phi_i(\mathbf{x}),\\ p(\mathbf{x},t) &= \sum_i p_i(t)\psi_i(\mathbf{x}),\end{aligned}$$

Need different ϕ_i and ψ_i.

Galerkin

$$\left\langle \rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) + \nabla p - \mu\nabla^2\mathbf{u}, \phi_j \right\rangle = 0 \quad \text{all } \phi_j,$$

and incompressibility constraint

$$\langle \nabla\cdot\mathbf{u}, \psi_j\rangle = 0 \quad \text{all } \psi_j.$$

Integration by parts

Integration by parts

$$\rho\left(M_{ij}\dot{\mathbf{u}}_j + Q_{ijk}\mathbf{u}_j\mathbf{u}_k\right) = -B_{ji}p_j - \mu K_{ij}\mathbf{u}_j,$$

and

$$-B_{ij}\mathbf{u}_j = 0,$$

with mass M and stiffness K as before, and two new coupling matrices

$$Q_{ijk} = \langle \phi_i \nabla \phi_j, \phi_k \rangle \quad \text{and} \quad B_{ij} = \langle \nabla \psi_i, \phi_j \rangle = -\langle \psi_i, \nabla \phi_j \rangle.$$

b. Time integration

Time step semi-discretised form with any FD algorithm

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \Delta t \dot{\mathbf{u}}_i$$

Time step semi-discretised form with any FD algorithm

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \Delta t \dot{\mathbf{u}}_i$$

Incompressible by projection split step

$$\begin{aligned} \mathbf{u}^* &= \mathbf{u}_i^n + \Delta t \,(\dot{\mathbf{u}}_i^n \quad \text{without the } p \text{ term})\,, \\ \mathbf{u}^{n+1} &= \mathbf{u}^* + \Delta t \,(\dot{\mathbf{u}}_i^n \quad \text{with just the } p \text{ term})\,, \end{aligned}$$

b. Time integration

Time step semi-discretised form with any FD algorithm

$$\mathbf{u}_i^{n+1} = \mathbf{u}_i^n + \Delta t \dot{\mathbf{u}}_i$$

Incompressible by projection split step

$$\begin{aligned} \mathbf{u}^* &= \mathbf{u}_i^n + \Delta t \, (\dot{\mathbf{u}}_i^n \quad \text{without the } p \text{ term}) \, , \\ \mathbf{u}^{n+1} &= \mathbf{u}^* + \Delta t \, (\dot{\mathbf{u}}_i^n \quad \text{with just the } p \text{ term}) \, , \end{aligned}$$

with p chosen so the incompressibility at the end of the step

$$B\mathbf{u}^{n+1} = 0.$$

Problems with pressure – Locking

Consider triangles with velocity linear and pressure constant

Problems with pressure – Locking

Consider triangles with velocity linear and pressure constant
Then

$$\langle \nabla \cdot \mathbf{u}, \psi_j \rangle = 0 \quad \text{all } j,$$

gives

Problems with pressure – Locking

Consider triangles with velocity linear and pressure constant
Then

$$\langle \nabla \cdot \mathbf{u}, \psi_j \rangle = 0 \quad \text{all } j,$$

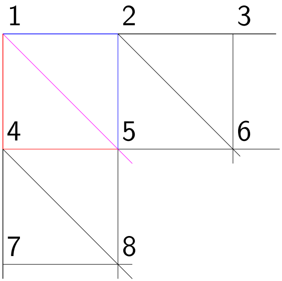
gives

$$\oint_{\Delta_j} u_n = 0,$$

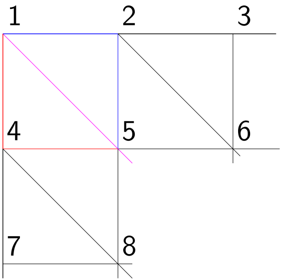
i.e. no net volume flux out of triangle Δ_j.

. . . locking

Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



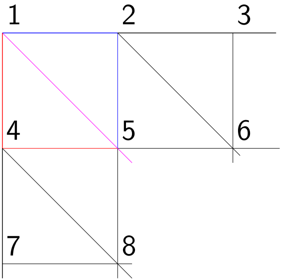
Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,

...locking

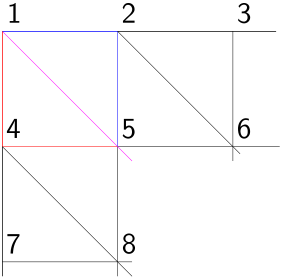
Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,
flux in over edge 45 is $\frac{1}{2} h v_5$,

. . . locking

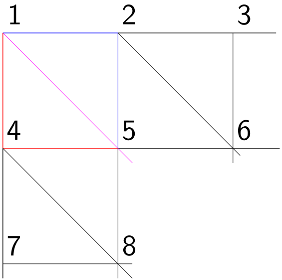
Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,
flux in over edge 45 is $\frac{1}{2}hv_5$, flux out over edge 15 is $\frac{1}{2}h(u_5 + v_5)$

. . . locking

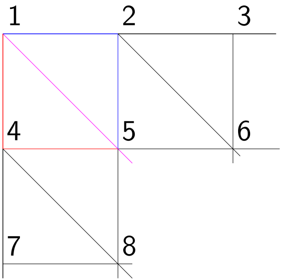
Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,
flux in over edge 45 is $\frac{1}{2}hv_5$, flux out over edge 15 is $\frac{1}{2}h(u_5 + v_5)$
Hence $u_5 = 0$,

. . . locking

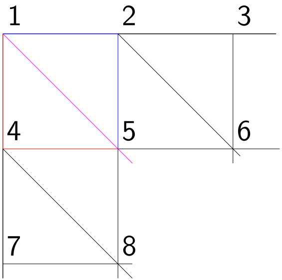
Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,
flux in over edge 45 is $\frac{1}{2}hv_5$, flux out over edge 15 is $\frac{1}{2}h(u_5 + v_5)$
Hence $u_5 = 0$, by symmetry (triangle 125) $v_5 = 0$.

...locking

Consider top corner, with $\mathbf{u} = 0$ on boundary (74123).



For triangle 145,
flux in over edge 45 is $\frac{1}{2}hv_5$, flux out over edge 15 is $\frac{1}{2}h(u_5 + v_5)$
Hence $u_5 = 0$, by symmetry (triangle 125) $v_5 = 0$.

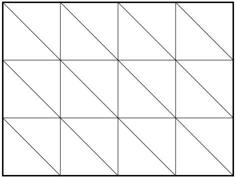
Then $\mathbf{u}_6 = 0$ and $\mathbf{u}_8 = 0$, so $\mathbf{u} \equiv 0$.

. . . locking

For one triangle there are $1p + 3u + 3v$ variables.

. . . locking

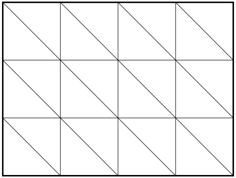
For one triangle there are $1p + 3u + 3v$ variables.
But on a 4×3 grid



there are $24p + 6u + 6v$ variables.

. . . locking

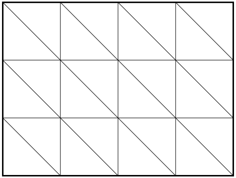
For one triangle there are $1p + 3u + 3v$ variables.
But on a 4×3 grid



there are $24p + 6u + 6v$ variables. Too many p

. . . locking

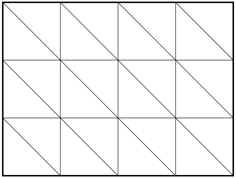
For one triangle there are $1p + 3u + 3v$ variables.
But on a 4×3 grid



there are $24p + 6u + 6v$ variables. Too many p
Create more u & v with bubble functions (vanish on boundaries of elements),

. . . locking

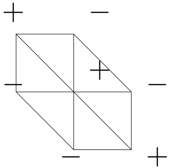
For one triangle there are $1p + 3u + 3v$ variables.
But on a 4×3 grid



there are $24p + 6u + 6v$ variables. Too many p
Create more u & v with bubble functions (vanish on boundaries of elements), or reduce number of pressure

Spurious pressure modes

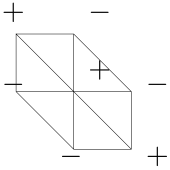
if have p linear over triangle



As in Algorithm 2 of driven cavity, above pressure drives no flow

Spurious pressure modes

if have p linear over triangle



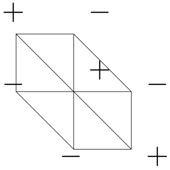
As in Algorithm 2 of driven cavity, above pressure drives no flow

$$B_{ji} p_j = 0.$$

has eigensolutions.

Spurious pressure modes

if have p linear over triangle



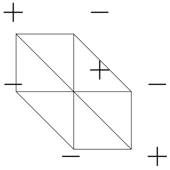
As in Algorithm 2 of driven cavity, above pressure drives no flow

$$B_{ji}p_j = 0.$$

has eigensolutions. Avoid by choosing ϕ_i and ψ_i to satisfy the Babuška-Brezzi condition.

Spurious pressure modes

if have p linear over triangle



As in Algorithm 2 of driven cavity, above pressure drives no flow

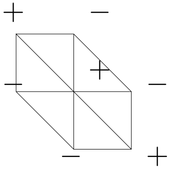
$$B_{ji} p_j = 0.$$

has eigensolutions. Avoid by choosing ϕ_i and ψ_i to satisfy the Babuška-Brezzi condition.
Alternatively, replace incompressibility by

$$\nabla \cdot \mathbf{u} = -\beta h^2 p, \quad \text{with optimal } \beta = 0.025$$

Spurious pressure modes

if have p linear over triangle



As in Algorithm 2 of driven cavity, above pressure drives no flow

$$B_{ji} p_j = 0.$$

has eigensolutions. Avoid by choosing ϕ_i and ψ_i to satisfy the Babuška-Brezzi condition.
Alternatively, replace incompressibility by

$$\nabla \cdot \mathbf{u} = -\beta h^2 p, \quad \text{with optimal } \beta = 0.025$$

Weak formulation

$$B_{ij} \mathbf{u}_j - \beta h^2 p_i = 0.$$

Also upwinding

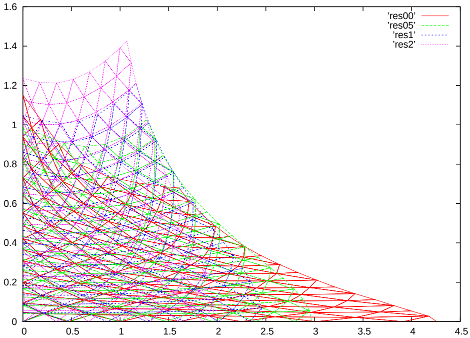
- Petrov-Galerkin: Add upstream bias to weight functions,

Also upwinding

- Petrov-Galerkin: Add upstream bias to weight functions, but adds artificial numerical streamwise diffusion

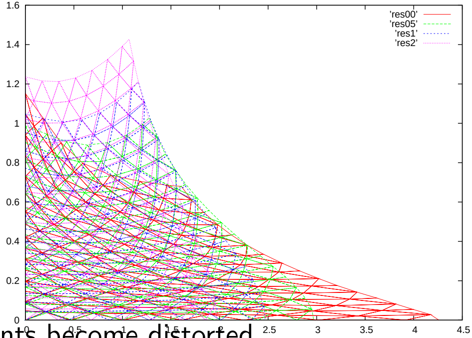
Also upwinding

- Petrov-Galerkin: Add upstream bias to weight functions, but adds artificial numerical streamwise diffusion
- Lagrangian Finite Elements – elements advected with flow,



Also upwinding

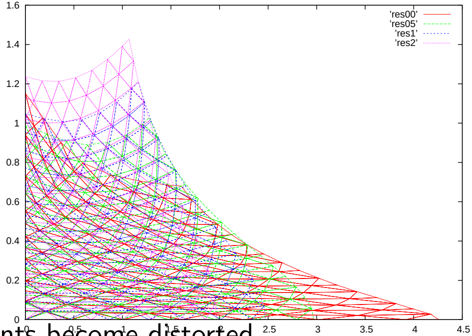
- Petrov-Galerkin: Add upstream bias to weight functions,
  but adds artificial numerical streamwise diffusion
- Lagrangian Finite Elements – elements advected with flow,



but elements become distorted

Also upwinding

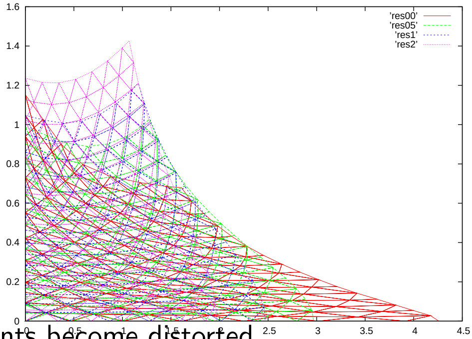
- Petrov-Galerkin: Add upstream bias to weight functions,
  but adds artificial numerical streamwise diffusion
- Lagrangian Finite Elements – elements advected with flow,



but elements become distorted
$\rightarrow$ re-griding

Also upwinding

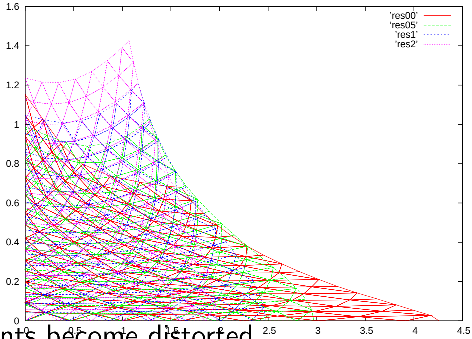
- Petrov-Galerkin: Add upstream bias to weight functions,
  but adds artificial numerical streamwise diffusion
- Lagrangian Finite Elements – elements advected with flow,



but elements become distorted
$\rightarrow$ re-griding, e.g. diagonal swapping.

Also upwinding

- Petrov-Galerkin: Add upstream bias to weight functions,
  but adds artificial numerical streamwise diffusion
- Lagrangian Finite Elements – elements advected with flow,



  but elements become distorted
  → re-griding, e.g. diagonal swapping.
- ALE – somewhere between Lagrangian and Eulerian.