Hello and Welcome Everyone!

Special welcome to my committee:

Professor Randolph E. Bank, Chair
Professor Scott B. Baden
Professor David J. Benson
Professor Michael Holst
Professor Melvin Leok

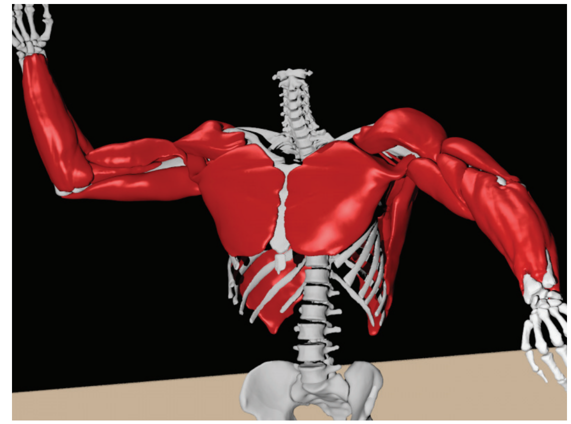
University of California, San Diego

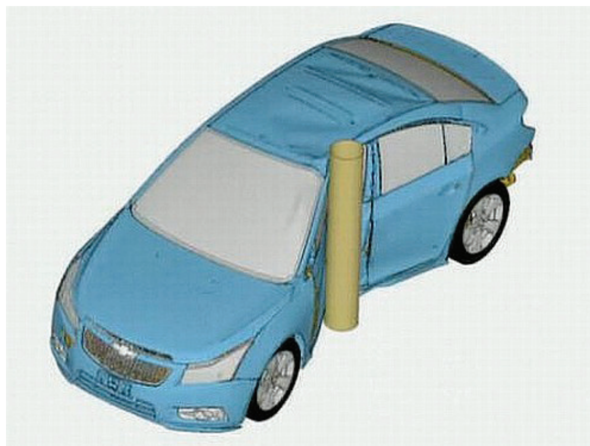
Domain Partitioning Methods for Elliptic Partial Differential Equations

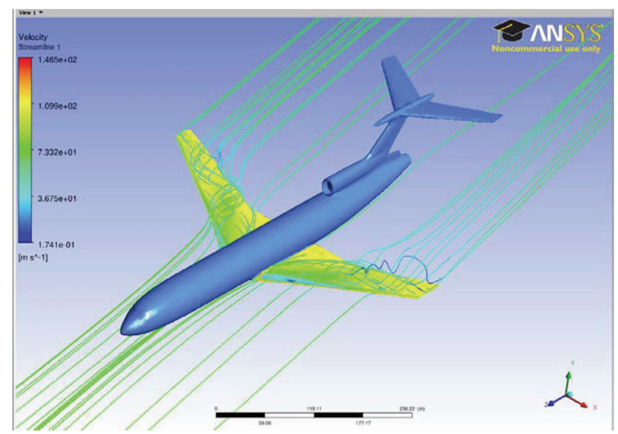
A dissertation submitted in partial satisfaction of the requirements for the degree
Doctor in Philosophy
in
Mathematics w/ Specialization Computational Science
by
Christopher G. Deotte

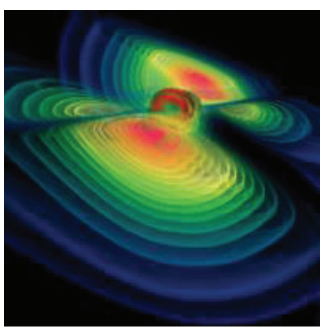
Committee in charge:
Professor Randolph E. Bank, Chair
Professor Scott B. Baden
Professor David J. Benson
Professor Michael Holst
Professor Melvin Leok

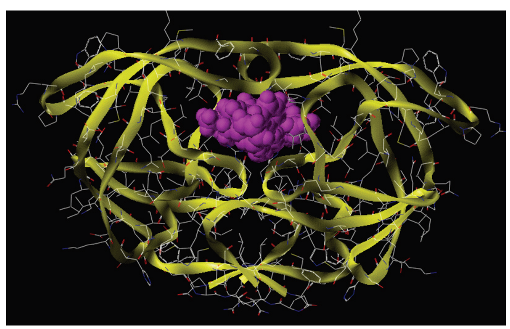
PDEs are the equations that describe the world around us. Their solutions predict and model reality.











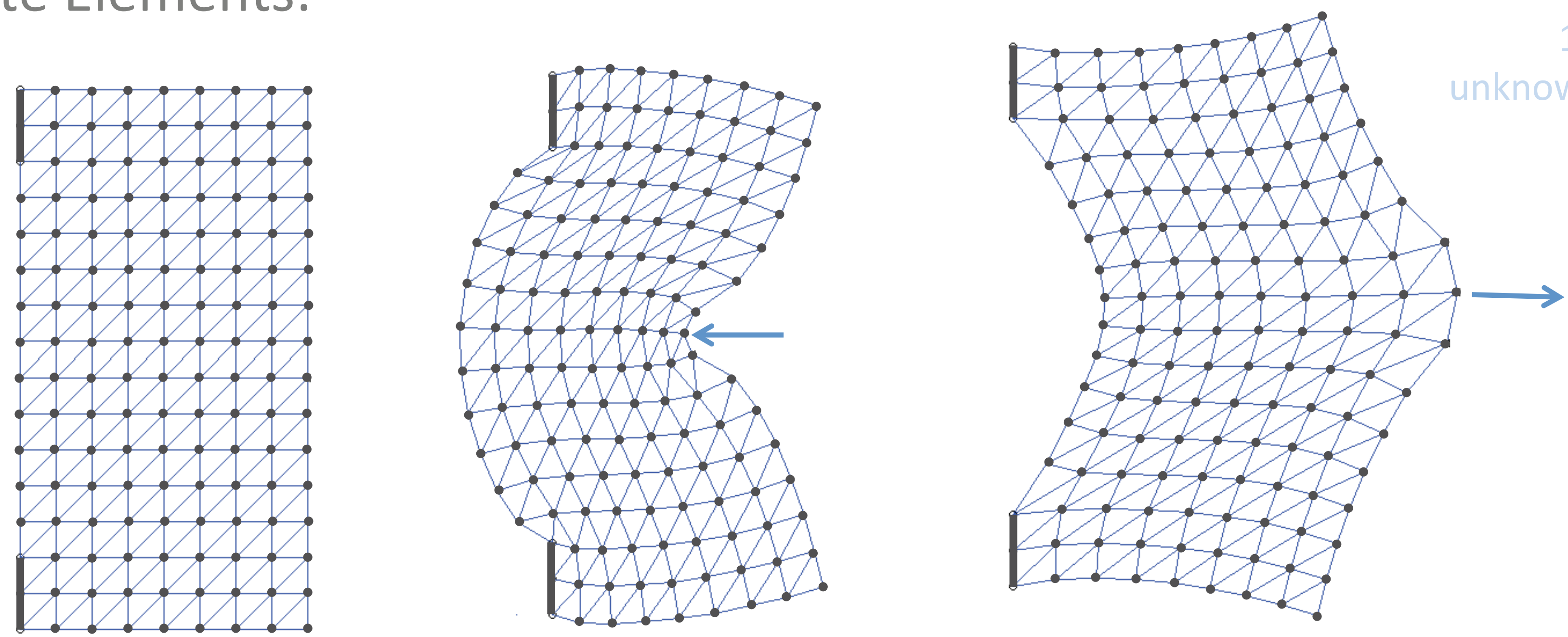
Calculus 101:

$$\frac{\mathrm{d}^2 u(t)}{\mathrm{d}t^2} = -9.8$$

$$u(t) = -4.9t^2 + v_0 t + u_0$$

Finite Elements:

138
unknowns



$$-10.4\nabla^2 u - 16.4\nabla \cdot \varepsilon(u) = 0$$

$$u(x, y) = ?$$

Our Second Order Elliptic Partial Differential Equation:

See pg 1-8
for more info

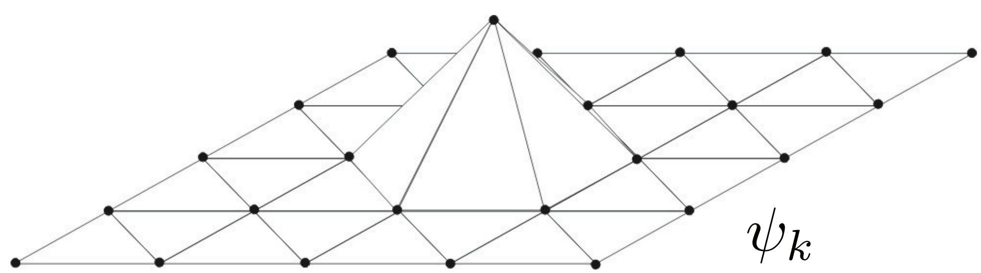
Find $u \in H^2(\Omega)$ s.t.

$$\begin{aligned} -\nabla \cdot (a(x,y)\nabla u) + b(x,y) \cdot \nabla u + c(x,y)u &= f(x,y) \quad &&\text{in } \Omega \\ a(x,y)\nabla u \cdot n &= g_N(x,y) \quad &&\text{on } \partial\Omega_N \\ u &= g_D(x,y) \quad &&\text{on } \partial\Omega_D \end{aligned}$$

Here $\Omega \in \mathbb{R}^d$ is a bounded domain, n is the unit normal vector, a is a $d \times d$ spd matrix, b is a vector of length d, and $[a]_{i,j}$, $[b]_i$, c, f, g_N, and g_D are scalar functions.

→ weak form → triangle finite elements → linear Lagrange basis functions →

Find $u_h \in S_h \subset H^1(\Omega)$ s.t.
$u_h = u_D + \sum_{k=1}^{n} U_k \psi_k$

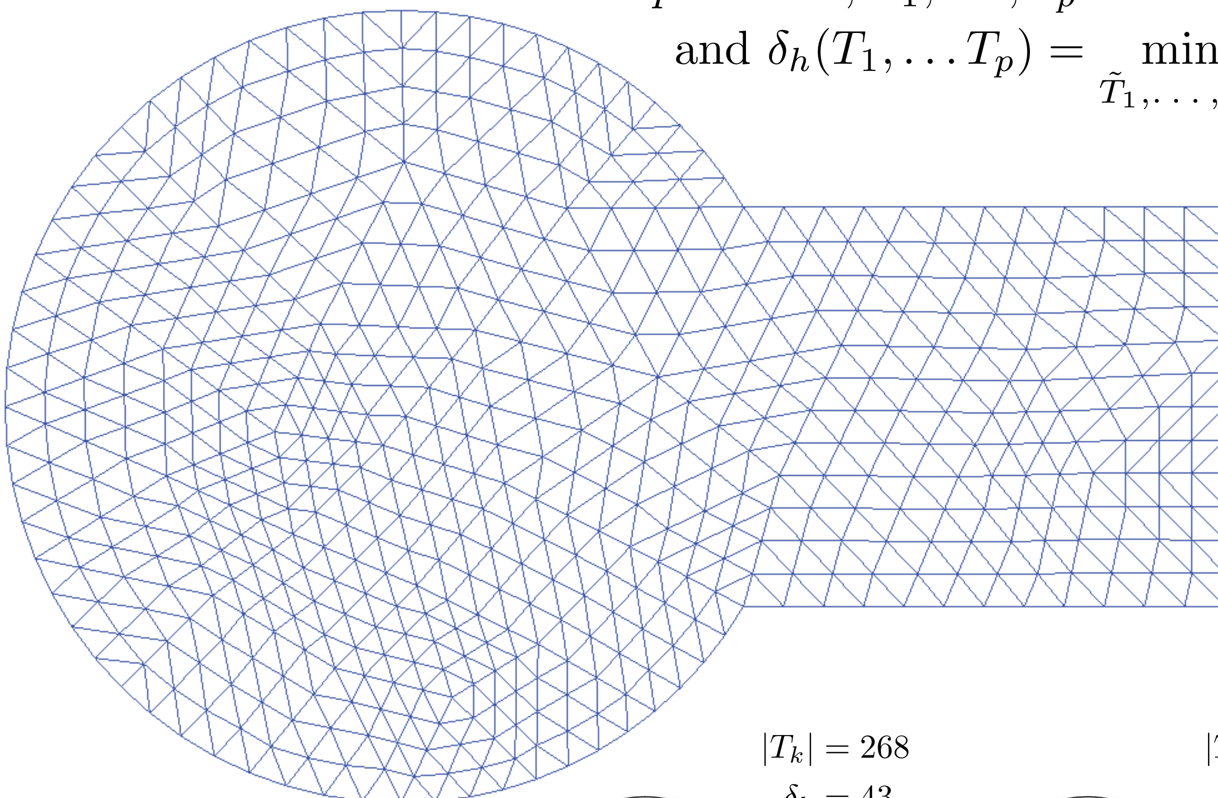


Graph Partitioning

See pg 21-32
for more info

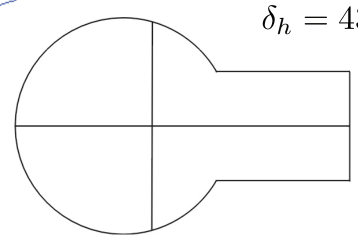
583 (1072 triangles)
unknowns

Partition $\Omega \equiv \bigcup_{i=1}^{n} t_i$ triangles into p subsets, $T_1, \ldots, T_p$ such that $|T_i| \approx |T_j| \, \forall i, j$ and $\delta_h(T_1, \ldots T_p) = \min_{\tilde{T}_1, \ldots, \tilde{T}_p} \delta_h(\tilde{T}_1, \ldots, \tilde{T}_p)$

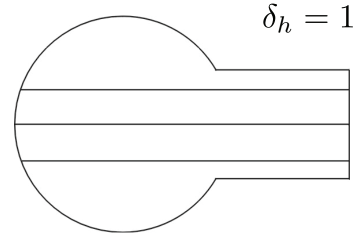


· Equal parts
· Minimize cut

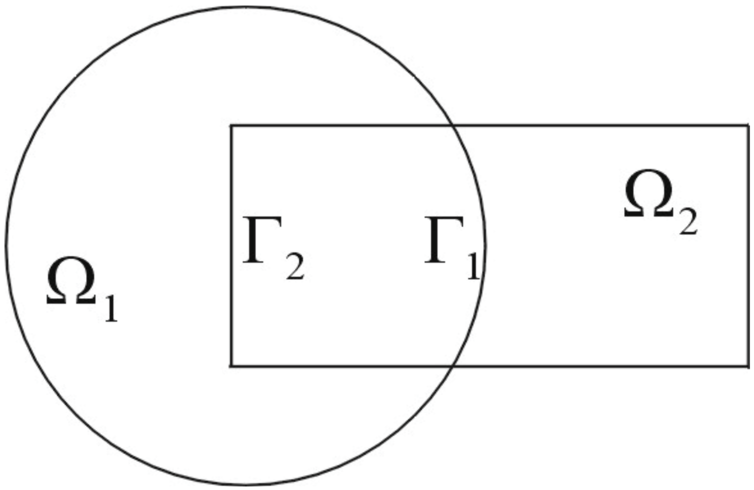
$|T_k| = 268$
$\delta_h = 43$



$|T_k| = 268$
$\delta_h = 111$



Schwarz Framework



Define L as the differential operator

$$Lu \equiv -\nabla \cdot (a(x,y)\nabla u) + b(x,y)\cdot \nabla u + c(x,y)u$$

Define B as the boundary operator

$$Bu \equiv \begin{cases} a(x,y)\nabla u \cdot n \text{ on } \Omega_N \\ u \text{ on } \Omega_D \end{cases}$$

- Find $u_1 \in H^2(\Omega_1)$ s.t.

$$Lu_1 = f_1 \text{ in } \Omega_1$$
$$Bu_1 = g_1 \text{ on } \partial\Omega_1 \backslash \Gamma_1$$
$$u_1 = u_2 \text{ on } \Gamma_1$$

- Find $u_2 \in H^2(\Omega_2)$ s.t.

$$Lu_2 = f_2 \text{ in } \Omega_2$$
$$Bu_2 = g_2 \text{ on } \partial\Omega_2 \backslash \Gamma_2$$
$$u_2 = u_1 \text{ on } \Gamma_2$$

Communicate Γ_1 and Γ_2.

Matrix Form: Additive and Multiplicative Schwarz

$$\begin{bmatrix} A_1 & A_{1,2} \\ A_{2,1} & A_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}$$

$$\downarrow$$

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^k = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ A_{2,1} & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^{k-1} - \begin{bmatrix} 0 & A_{1,2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^{k-1}$$

$$DU^k = (E+F)U^{k-1} + B$$

$$U^k = \left(D^{-1}(E+F)\right) U^{k-1} + C$$

Additive: Block Jacobi

or $\downarrow$

$$\begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^k = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ A_{2,1} & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^{k} - \begin{bmatrix} 0 & A_{1,2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix}^{k-1}$$

$$(D-E)U^k = FU^{k-1} + B$$

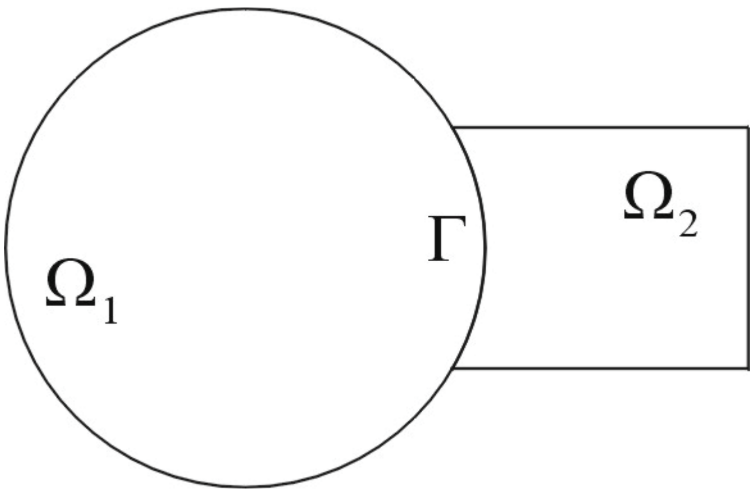
$$U^k = \left((D-E)^{-1}F\right) U^{k-1} + C$$

Multiplicative: Block Gauss-Seidel

Domain Decomposition: Non-overlapping subdomains

See pg 9-20 for more info

Lagrange Multiplier Framework



- Find $u_1 \in H^2(\Omega_1)$ s.t.

$$Lu_1 = f_1 \text{ in } \Omega_1$$
$$Bu_1 = g_1 \text{ on } \partial\Omega_1 \backslash \Gamma$$
$$a\nabla u_1 \cdot n = \lambda \text{ on } \Gamma$$

- Find $u_2 \in H^2(\Omega_2)$ s.t.

$$Lu_2 = f_2 \text{ in } \Omega_1$$
$$Bu_2 = g_2 \text{ on } \partial\Omega_2 \backslash \Gamma$$
$$a\nabla u_2 \cdot n = -\lambda \text{ on } \Gamma$$

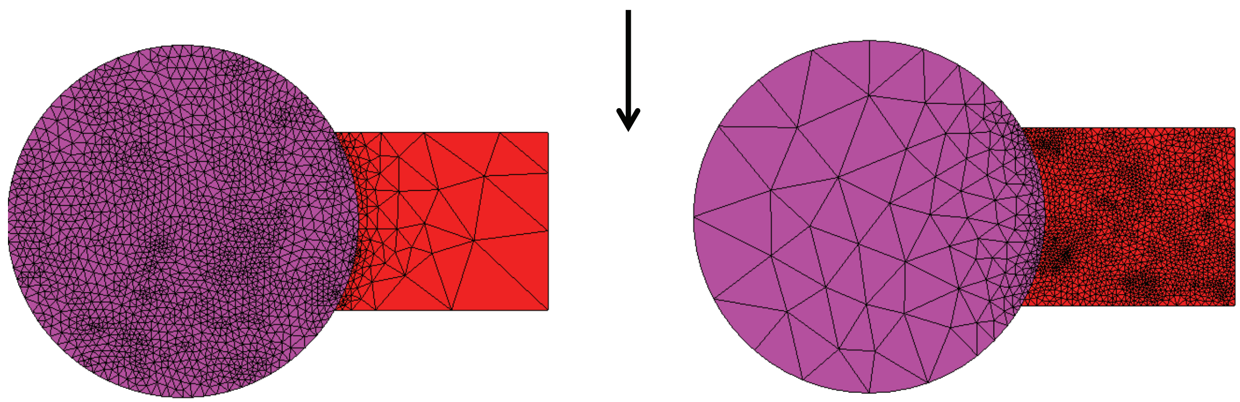
- Constrain $u_1 = u_2$ on Γ
  Update λ

Communicate λ

Communicate Γ

Matrix Form: Bank-Holst Paradigm DD Solver

$$
\begin{bmatrix}
A_1^{II} & A_1^{IB} & 0 & 0 & 0 \\
A_1^{BI} & A_1^{BB} & 0 & 0 & I \\
0 & 0 & \bar{A}_2^{II} & \bar{A}_2^{IB} & 0 \\
0 & 0 & \bar{A}_2^{BI} & \bar{A}_2^{BB} & -I \\
0 & I & 0 & -I & 0
\end{bmatrix}
\begin{bmatrix}
\delta U_1^I \\
\delta U_1^B \\
\delta \bar{U}_2^I \\
\delta \bar{U}_2^B \\
\lambda
\end{bmatrix}
=
\begin{bmatrix}
R_1^I \\
R_1^B \\
R_2^I \\
R_2^B \\
U_2^B - U_1^B
\end{bmatrix}
$$



1800

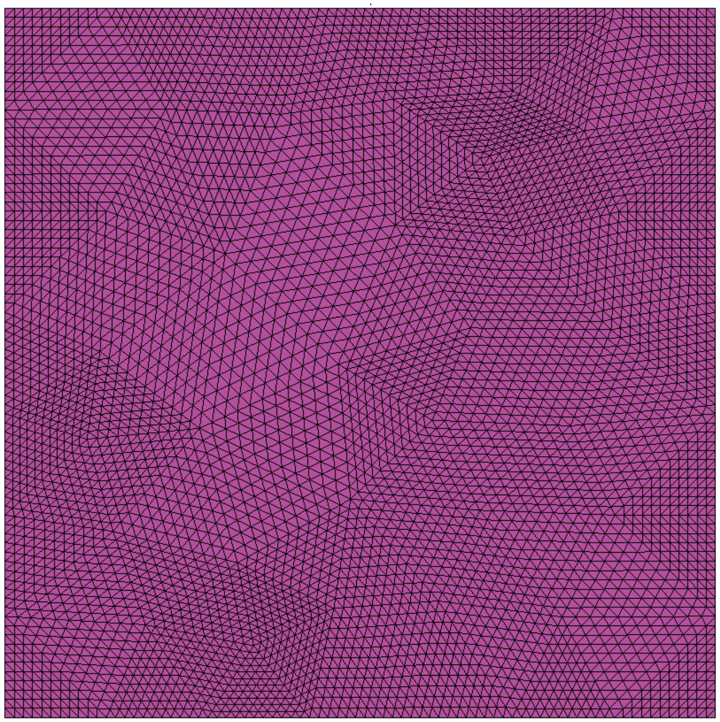
Unknowns

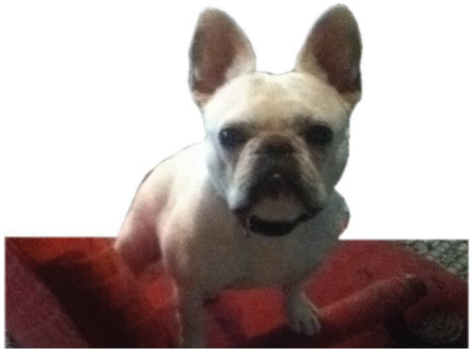
=100+1700

$$
\begin{bmatrix}
A_1^{II} & A_1^{IB} & 0 \\
A_1^{BI} & A_1^{BB} + \bar{A}_2^{BB} & \bar{A}_2^{BI} \\
0 & \bar{A}_2^{IB} & \bar{A}_2^{II}
\end{bmatrix}
\begin{bmatrix}
\delta U_1^I \\
\delta U_1^B \\
\delta \bar{U}_2^I
\end{bmatrix}
=
\begin{bmatrix}
R_1^I \\
R_1^B + R_2^B + \bar{A}_2^{BB}(U_2^B - U_1^B) \\
\bar{A}_2^{IB}(U_2^B - U_1^B)
\end{bmatrix}
$$

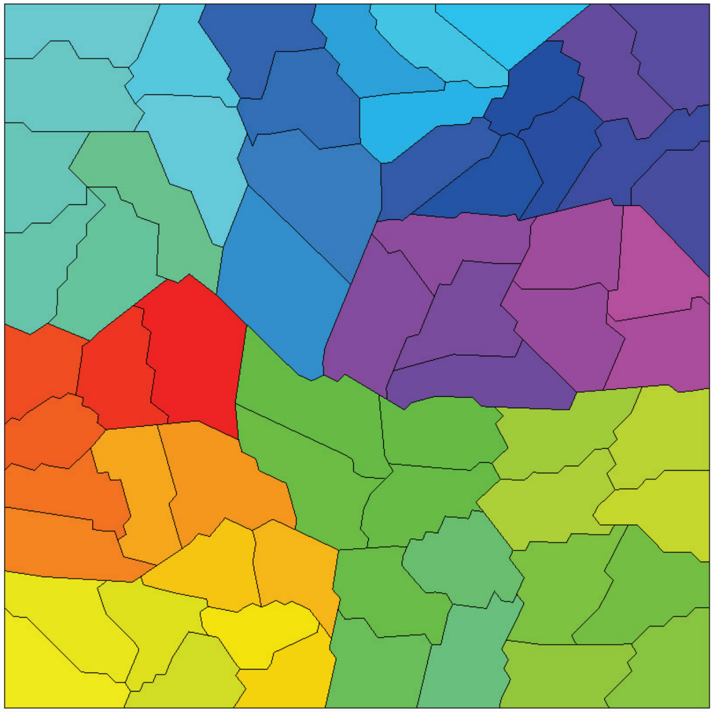
Metis 5.1.0: Graph Partitioner by George Karypis

4341 unknowns (8400 triangles)





- Equal parts
- Minimize cut



Background summary

<u>Solving a PDE</u>

(1) Find n unknowns instead of $u(x,y)$

(2) Partition with equal parts and minimize cut.

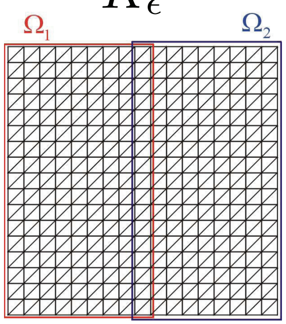
(3) Domain Decomposition Method with iterations.

Our contribution

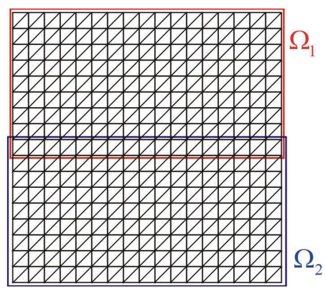
When a PDE has strong convection or anistropic diffusion, directional dependence exists and a partition that favors this direction is desirable. Incorporating this information into (2) reduces DD iterations.

See pg 86-102 for more info

$K_\epsilon^{(1)}$



$K_\epsilon^{(2)}$



Finite Elements with DD Additive Schwarz

$$U^k = \left(D^{-1}(E+F)\right) U^{k-1} + C$$

	$\rho(D_{(1)}^{-1}(E_{(1)}+F_{(1)}))$	?	$\rho(D_{(2)}^{-1}(E_{(2)}+F_{(2)}))$
$-\triangle u - 1 = 0$	λ	$=$	λ
$-\nabla \cdot \begin{bmatrix} \beta & 0 \\ 0 & 1 \end{bmatrix} \nabla u - 1 = 0$ $\beta > 1$	$\dfrac{U_{\frac{m}{2}-1}\left(1+\frac{1}{\beta}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}{U_{\frac{m}{2}}\left(1+\frac{1}{\beta}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}$	$>$	$\dfrac{U_{\frac{m}{2}-1}\left(1+\beta\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}{U_{\frac{m}{2}}\left(1+\beta\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}$
$-\triangle u + \begin{bmatrix} \beta \\ 0 \end{bmatrix} \nabla u - 1 = 0$ $\beta h > 1$	$\dfrac{U_{\frac{m}{2}-1}\left(1+\frac{(1-\sqrt{\beta h})^2}{2\sqrt{\beta h}}+\frac{1}{\sqrt{\beta h}}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}{U_{\frac{m}{2}}\left(1+\frac{(1-\sqrt{\beta h})^2}{2\sqrt{\beta h}}+\frac{1}{\sqrt{\beta h}}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}$	$>$	$\dfrac{U_{\frac{m}{2}-1}\left(1+\frac{(1-\sqrt{\beta h})^2}{2\sqrt{\beta h}}+\sqrt{\beta h}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}{U_{\frac{m}{2}}\left(1+\frac{(1-\sqrt{\beta h})^2}{2\sqrt{\beta h}}+\sqrt{\beta h}\left(1+\cos\left(\frac{m\pi}{m+1}\right)\right)\right)}$

where $U_k(\cdot)$ is the k^{th} Chebyshev polynomial of the second kind

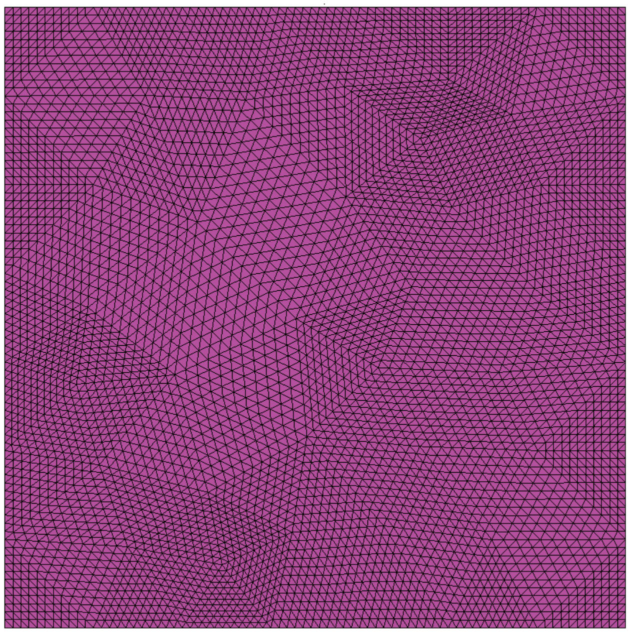
Encourage Rectangles

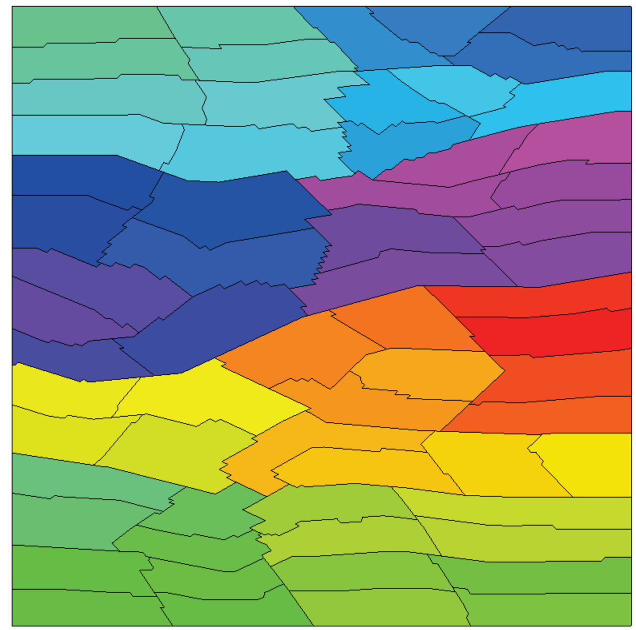
See pg 33-39
for more info

$$\beta = \frac{\delta_h(\tilde{K}_\epsilon^{\text{rect}})}{\delta_h(\tilde{K}_\epsilon^{\text{square}})} \approx \frac{1+\theta}{2\sqrt{\theta}}$$

Aspect ratio θ versus growth factor β

θ	1	2	3	4	5	6	10	14	25
β	1.00	1.06	1.15	1.25	1.34	1.43	1.74	2.00	2.60

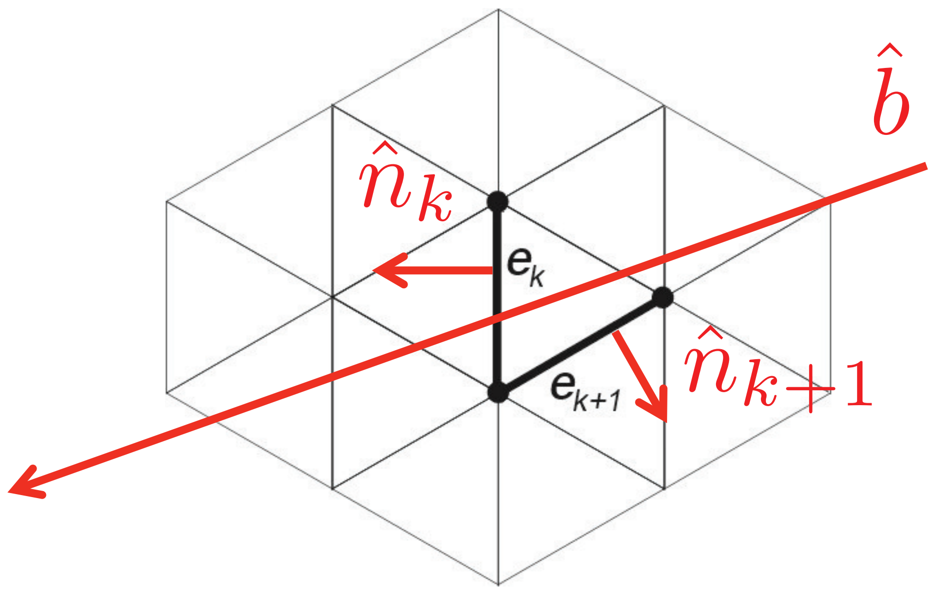




$$\text{rectangle DD time increase} = \frac{\alpha+\beta}{\alpha+1} \quad \text{where } \alpha = \frac{T^{\text{comp}}}{T^{\text{comm}}}$$

See pg 40-45 for more info

Define edge weight $w(e_k) = q_s(\hat{n}_k \cdot \hat{b})$ with $q_s(r) = (sr)^s + 1$
and $\hat{b}$ is the direction of convection or ∇u
and $\hat{n}_k$ is the unit normal to triangle side k.

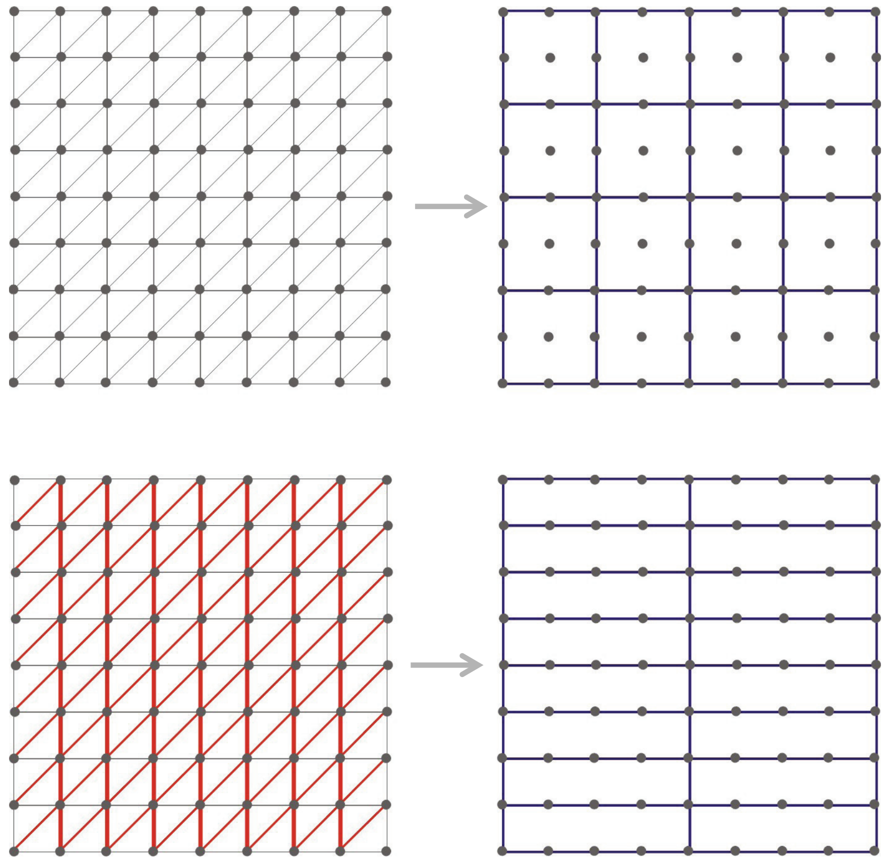


$$w(e_k) = q_s(0.94)$$
$$w(e_{k+1}) = q_s(0.17)$$

Example of Convection Weighting

16 parts

Unweighted
128 triangles
(81 unknowns)



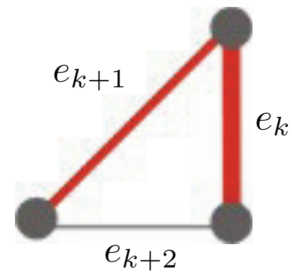
Weighted
128 triangles
(81 unknowns)

16 parts

Convection

Convection

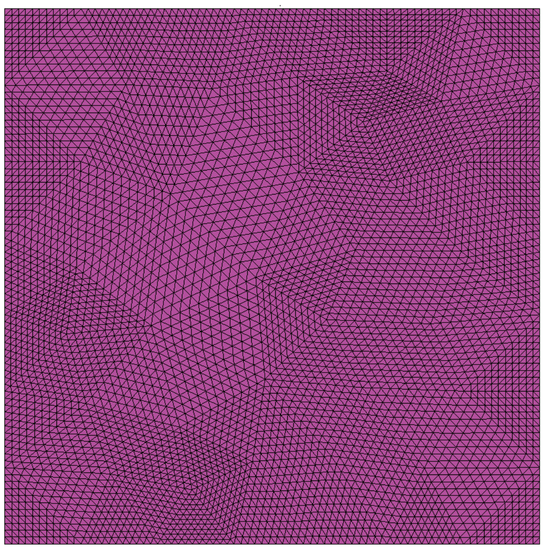
with $s = 1.83$



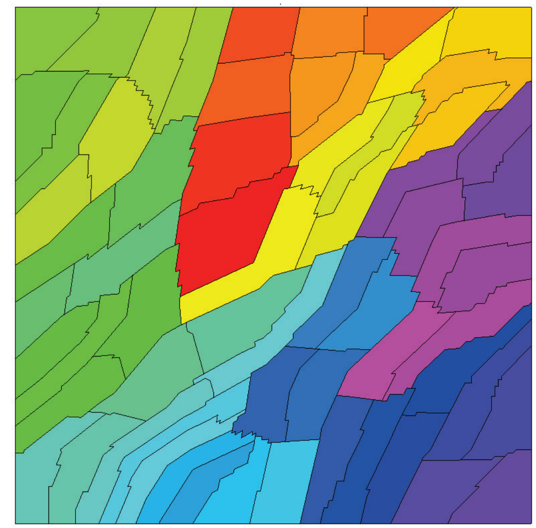
$$w(e_k) = 4$$
$$w(e_{k+1}) = 2.6$$
$$w(e_{k+2}) = 1$$

More Examples of Convection Weighting

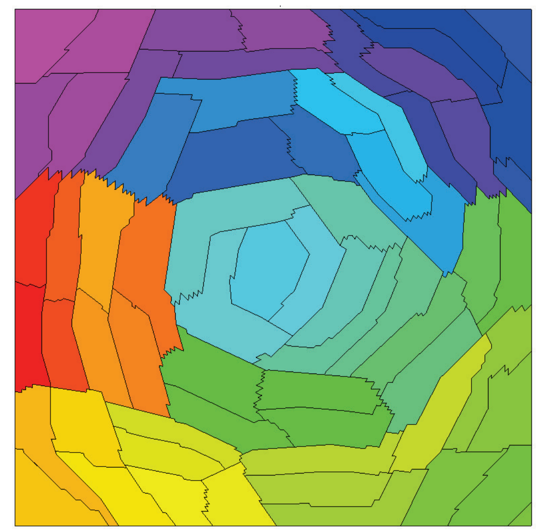
Weighted
8400 triangles
(4341 unknowns)



$$b = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$



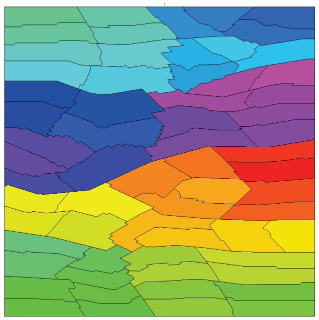
$$b = \begin{bmatrix} 0.5 - y \\ x - 0.5 \end{bmatrix}$$



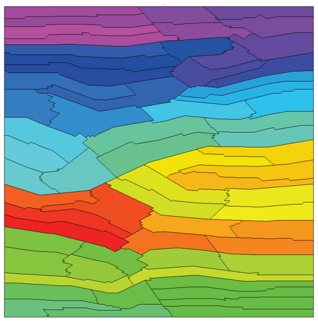
$$-\triangle u + b \cdot \nabla u = 1 \text{ on } \Omega = [0,1] \times [0,1] \in \mathbb{R}^2$$

Convection Weighting Parameter *s*

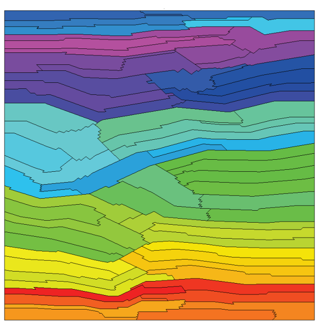
$$w(e_k) = q_s(\hat{n}_k \cdot \hat{b})$$
$$q_s(r) = (sr)^s + 1$$



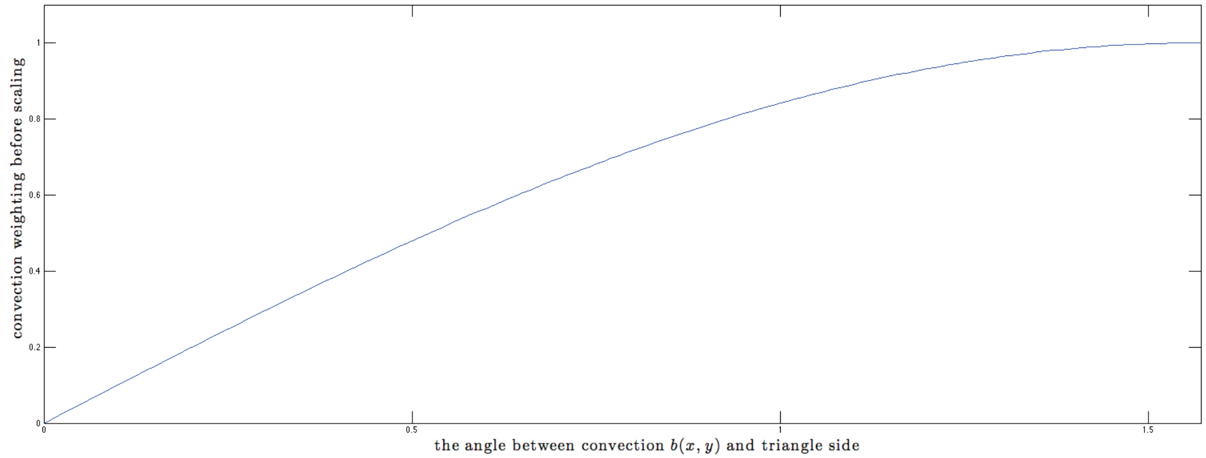
$s = 2$

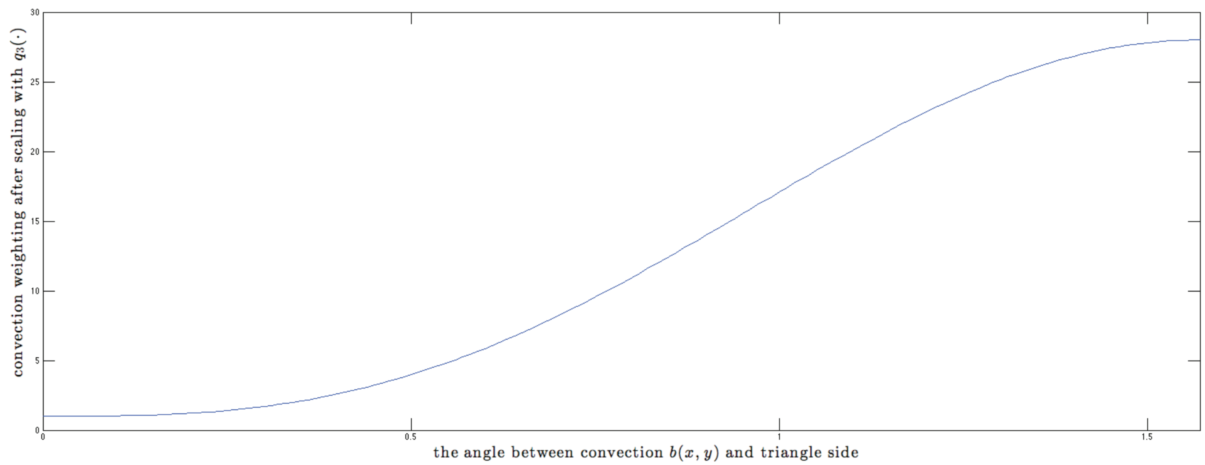


$s = 2.75$



$s = 4$

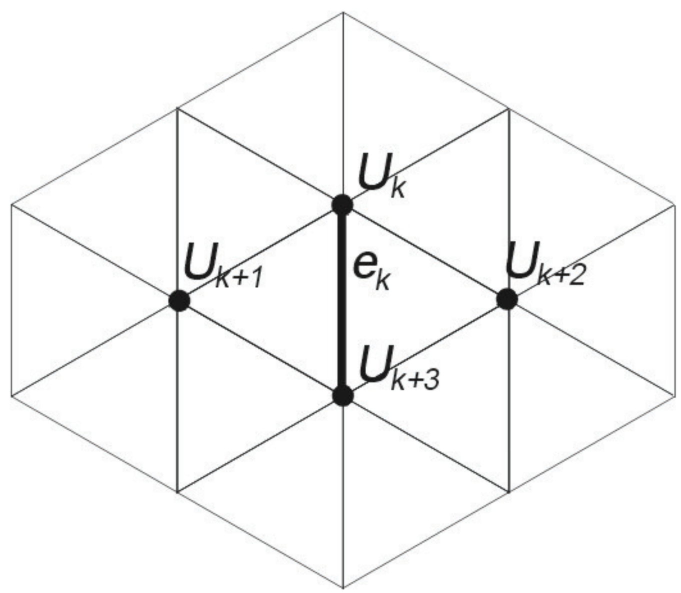




See pg 44-55, 57-66 for more info

$$\text{Define } w(e_k) = \max\left\{ \frac{|A_{k+1,k}| + |A_{k+1,k+3}|}{2|A_{k+1,k+1}|}, \frac{|A_{k,k+1}|}{2|A_{k,k}|} + \frac{|A_{k+3,k+1}|}{2|A_{k+3,k+3}|} \right\}$$
$$+ \max\left\{ \frac{|A_{k+2,k}| + |A_{k+2,k+3}|}{2|A_{k+2,k+2}|}, \frac{|A_{k,k+2}|}{2|A_{k,k}|} + \frac{|A_{k+3,k+2}|}{2|A_{k+3,k+3}|} \right\}$$

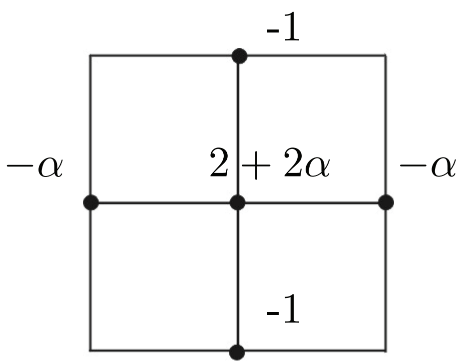
where A is the Finite Element Stiffness Matrix
and indices $k, k+1, k+2, k+3$ are defined in the Figure below.



Stiffness Matrix Weighting (on a square mesh)

See pg 44-55, 57-66 for more info

$$-\nabla \cdot \begin{bmatrix} \alpha & 0 \\ 0 & 1 \end{bmatrix} \nabla u - 1 = 0$$



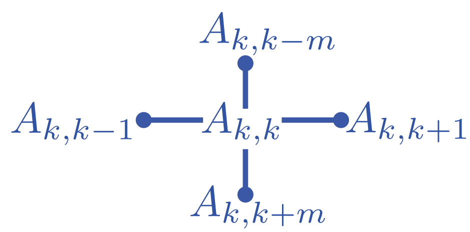
$$w(e_y) = \frac{2\alpha}{2+2\alpha} \to 1$$

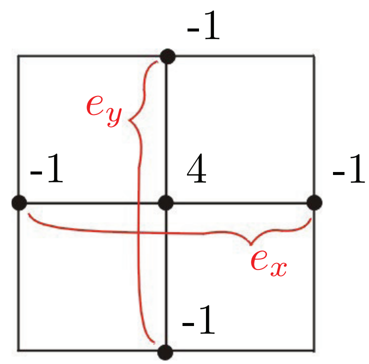
$$w(e_x) = \frac{2}{2+2\alpha} \to 0$$

$$\text{(for triangles } w(e_x) \to \frac{1}{4}\text{)}$$

Convection

$$w(e_y) = \frac{|A_{k,k-1}|+|A_{k,k+1}|}{|A_{k,k}|}$$





$$w(e_y) = \frac{1}{2}$$

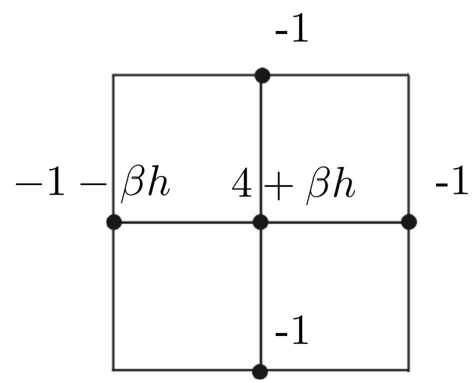
$$w(e_x) = \frac{1}{2}$$

$$-\triangle u - 1 = 0$$

$$w(e_y) = \frac{2+\beta h}{4+\beta h} \to 1$$

$$w(e_x) = \frac{2}{4+\beta h} \to 0$$

$$\text{(for triangles } w(e_x) \to \frac{1}{4}\text{)}$$

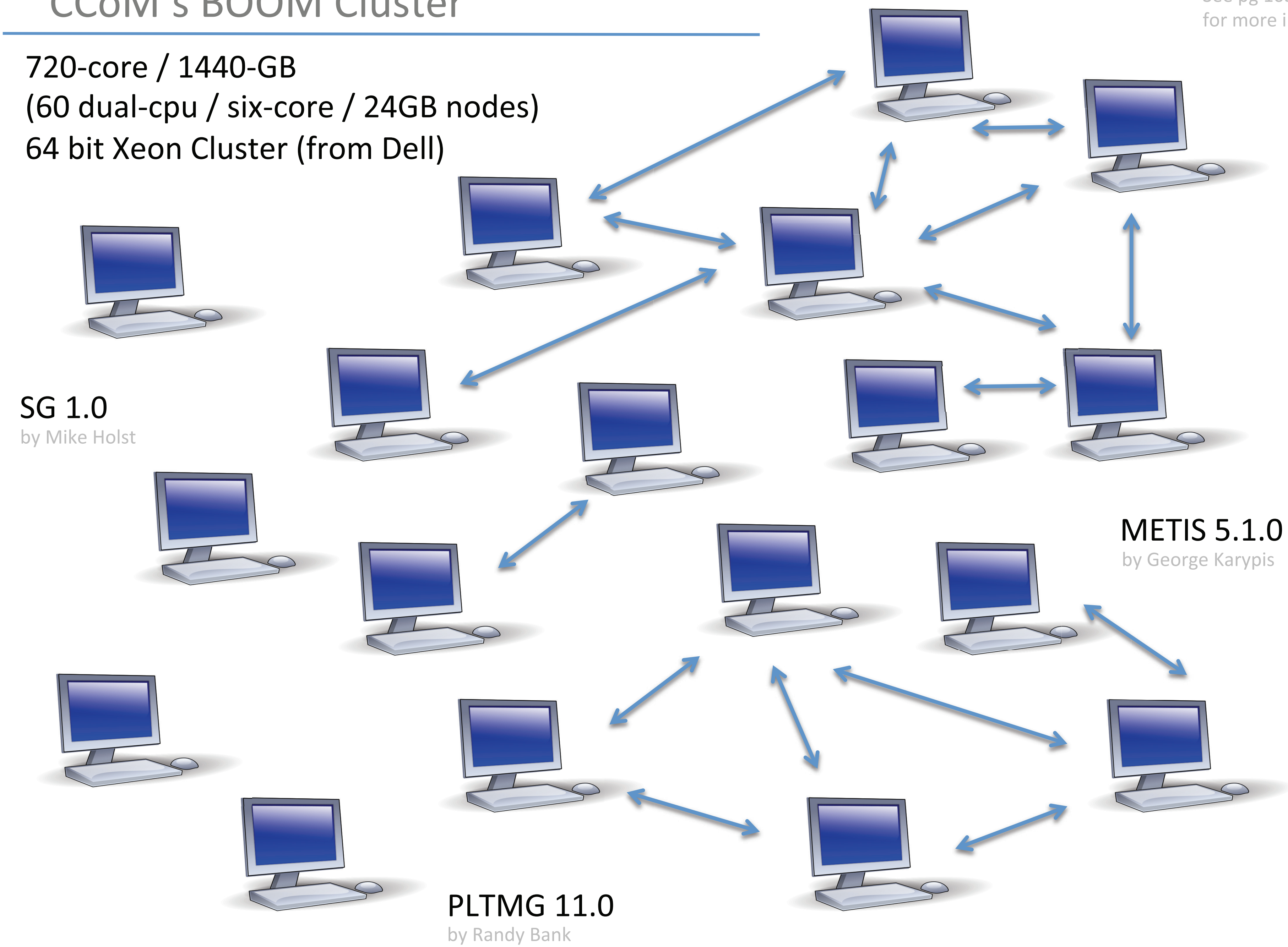


$$-\triangle u + \begin{bmatrix} \beta \\ 0 \end{bmatrix} \nabla u - 1 = 0$$

CCoM's BOOM Cluster

See pg 103-104
for more info

720-core / 1440-GB
(60 dual-cpu / six-core / 24GB nodes)
64 bit Xeon Cluster (from Dell)



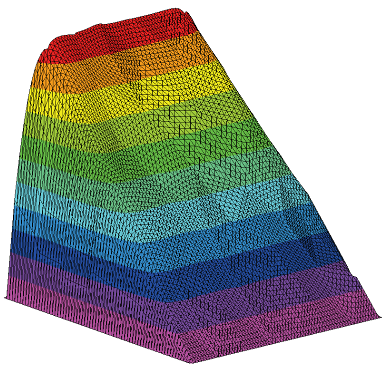
Fortran 90 Code:

See pg 104-108
for more info

```
      function geta(i,j,nb,ndf,map,a,ja,jap)
c
          use mthdef
          implicit real(kind=rknd) (a-h,o-z)
          implicit integer(kind=iknd) (i-n)
          integer(kind=iknd), dimension(*) :: ja,jap
          integer(kind=iknd), dimension(ndf) :: map
          real(kind=rknd), dimension(*) :: a
c
c     this routine returns A(i,j)
c
          nb_low = map(i)
          nb_high = map(j)
          if (nb_low == nb_high) then
            geta = a(nb_low)
          endif
          l_shift = 0
          if (nb_low > nb_high) then
            l_shift = ja(nb+1)-ja(1)
            nb_low = map(j)
            nb_high = map(i)
          endif
          if (nb_low /= nb_high) then
            n_index = -1
            do k=ja(nb_low),ja(nb_low+1)-1
              if (nb_high == ja(k)) n_index=k
            enddo
            if (n_index /= -1) then
              geta = a(jap(n_index)+l_shift)
            else
              geta = 0.0
            endif
          endif
      return
      end
```

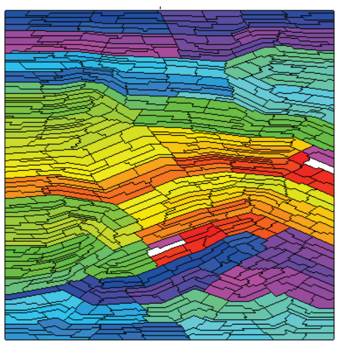
```
      do i=1,ntf
        i1=itdof(1,i)
        i2=itdof(2,i)
        i3=itdof(3,i)
        alpha=100.0
        beta=0.5
        wght(1,i)=
     +    MAX((abs(geta(i1,i2,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i1,i3,nb,ndf,map,a,ja,jns)))
     +    /2.0/abs(geta(i1,i1,nb,ndf,map,a,ja,jns)),
     +    abs(geta(i2,i1,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i2,i2,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i3,i1,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i3,i3,nb,ndf,map,a,ja,jns)))
        if (wght(1,i)>1.0) wght(1,i)=1.0
        wght(1,i)=alpha*wght(1,i)+beta

        wght(2,i)=
     +    MAX((abs(geta(i2,i1,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i2,i3,nb,ndf,map,a,ja,jns)))
     +    /2.0/abs(geta(i2,i2,nb,ndf,map,a,ja,jns)),
     +    abs(geta(i1,i2,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i1,i1,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i3,i2,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i3,i3,nb,ndf,map,a,ja,jns)))
        if (wght(2,i)>1.0) wght(2,i)=1.0
        wght(2,i)=alpha*wght(2,i)+beta

        wght(3,i)=
     +    MAX((abs(geta(i3,i2,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i3,i1,nb,ndf,map,a,ja,jns)))
     +    /2.0/abs(geta(i3,i3,nb,ndf,map,a,ja,jns)),
     +    abs(geta(i2,i3,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i2,i2,nb,ndf,map,a,ja,jns))
     +    + abs(geta(i1,i3,nb,ndf,map,a,ja,jns))
     +    /2.0/abs(geta(i1,i1,nb,ndf,map,a,ja,jns)))
        if (wght(3,i)>1.0) wght(3,i)=1.0
        wght(3,i)=alpha*wght(3,i)+beta
      enddo
```

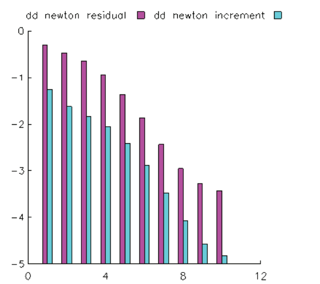
```
      real(kind=4) :: ubvec
      real(kind=4), dimension(nproc) :: tpwgts
      integer(kind=4) :: nvtxs,ncon,nparts,objval
      integer(kind=4), dimension(ntf+1) :: xadj,vwgt,part,vs
      integer(kind=4), dimension(3*ntf) :: adjncy,adjwgt
      integer(kind=4), dimension(40) :: options
      external METIS_METIS_SetDefaultOptions
      external METIS_PartGraphRecursive
      common /chris/wght(3,400000)


      call METIS_SetDefaultOptions(options)
      options(8) = 10
      index = 1
      nvtxs = ntf
      ncon = 1
      nparts = nproc
      ubvec = 1.001
      do i=1,nproc
        tpwgts(i) = 1.0/nproc
      enddo
      do i=1,ntf
        do j=1,3
          if (itedge(j,i)/4>0) then
            adjncy(index)=itedge(j,i)/4 - 1
            n=itedge(j,i)/4
            m=itedge(j,i)-4*n
            adjwgt(index)=wght(j,i)+wght(m,n)
            index=index+1
          endif
        enddo
        xadj(i+1)=index - 1
        vwgt(i)= 1
      enddo
      call METIS_PartGraphRecursive(nvtxs,ncon,xadj,adjncy,
     +  vwgt,vs,adjwgt,nparts,tpwgts,ubvec,options,objval,part)
      go to 50
```



$$\text{Find } u \in H^2(\Omega) \text{ s.t.}$$
$$-\triangle u - [10^4 \ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1] \times [0,1] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega$$

¼ Billion unknowns split among 512 processors!

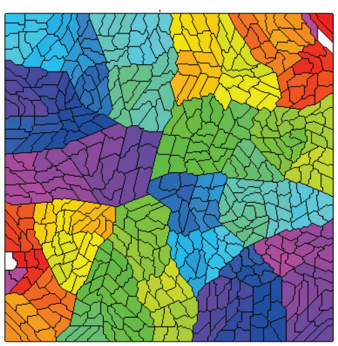


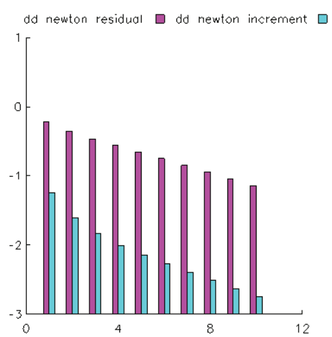



```
dd residual -- dd increment
iteration:   1    -0.30    -1.25
iteration:   2    -0.47    -1.62
iteration:   3    -0.65    -1.84
iteration:   4    -0.94    -2.06
iteration:   5    -1.36    -2.41
iteration:   6    -1.86    -2.89
iteration:   7    -2.43    -3.47
iteration:   8    -2.95    -4.07
iteration:   9    -3.27    -4.57
iteration:  10    -3.42    -4.83
convergence factors:    0.45     0.40
```

Weighted:
1.3 hours
$13.00 electricity

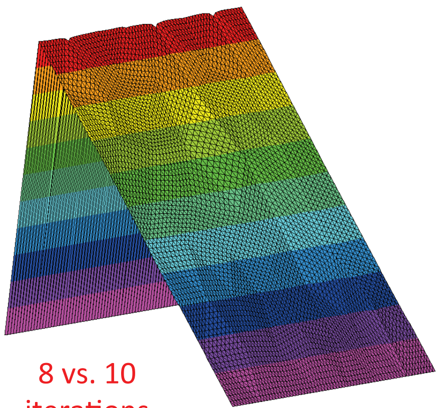
YES






```
dd residual -- dd increment
iteration:   1    -0.22    -1.25
iteration:   2    -0.36    -1.60
iteration:   3    -0.46    -1.83
iteration:   4    -0.56    -2.00
iteration:   5    -0.65    -2.14
iteration:   6    -0.75    -2.27
iteration:   7    -0.84    -2.39
iteration:   8    -0.94    -2.51
iteration:   9    -1.04    -2.63
iteration:  10    -1.14    -2.74
convergence factors:    0.79     0.68
```

Unweighted:
2.8 hours
$28.00 electricity

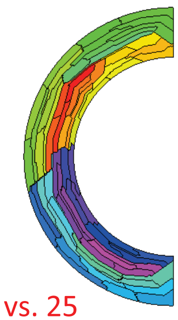


8 vs. 10 iterations

Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - [10^6 \ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1] \times [0,1] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega_D = \{0,1\} \times [0,1]$$
$$n \cdot \nabla u = 0 \text{ on } \partial\Omega_N = [0,1] \times \{0,1\}$$

DD convergence of $||r_k||$ and $||\delta u_k||$ were
Weighted: 0.29 and 0.28
Unweighted: 0.50 and 0.44

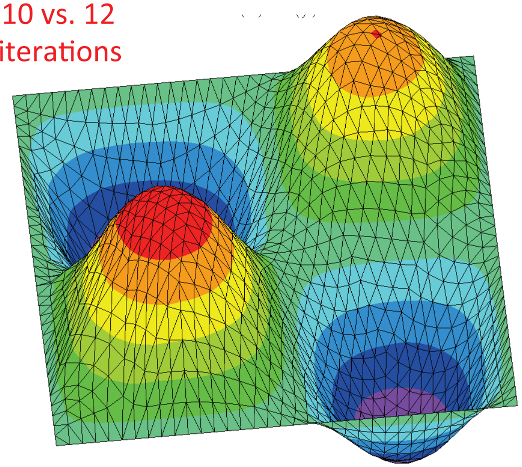


10 vs. 25 iterations

STOP DD when
$||\delta u_k||_{L^2(\Omega)} < \frac{1}{10}||u - u_h||_{L^2(\Omega)}$

YES

10 vs. 12 iterations

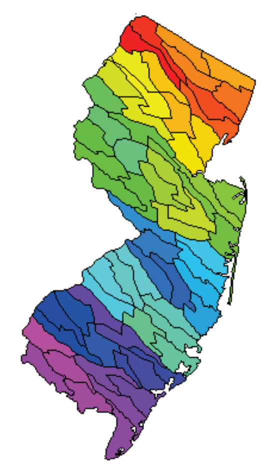


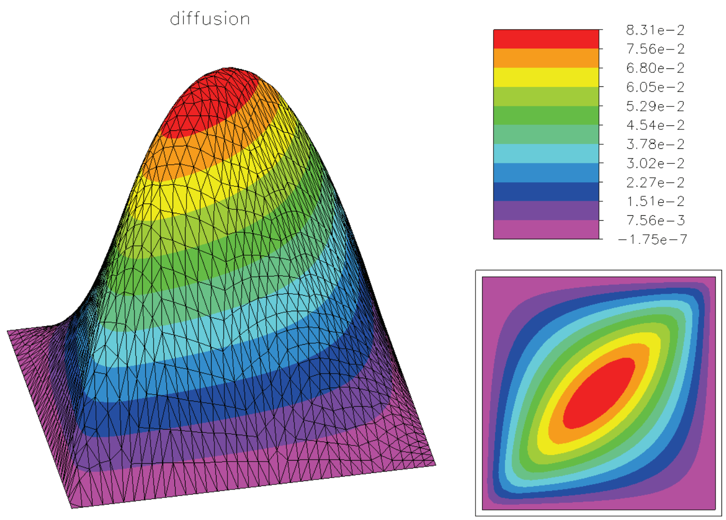
Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - [10^4 \ 0]^T \nabla u - f(x,y) = 0 \text{ on } \Omega = [0,2\pi] \times [0,2\pi] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega$$
$$f(x,y) = 2\sin(x)\sin(y) - 10^4\cos(x)\sin(y)$$

DD convergence of $||r_k||$ and $||\delta u_k||$ were
Weighted: 0.30 and 0.29
Unweighted: 0.48 and 0.41

10 vs. 24 iterations





Find $u \in H^2(\Omega)$ s.t.

$$-\nabla \cdot a\nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega$$
$$\text{where } a = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} \alpha & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}$$

YES

α	unweighted convergence $\|\|r_k\|\|/\|\|r_0\|\|$	weighted convergence $\|\|r_k\|\|/\|\|r_0\|\|$	unweighted convergence $\|\|\delta_k\|\|/\|\|u_0\|\|$	weighted convergence $\|\|\delta_k\|\|/\|\|u_0\|\|$	iteration reduction log/log
10^2	0.65	0.52	0.58	0.50	0.79
10^1	0.57	0.47	0.53	0.47	0.85
2	0.48	0.40	0.45	0.36	0.78
1.75	0.44	0.42	0.40	0.37	0.92
1	0.43	0.43	0.37	0.37	1.0

$$\frac{\lambda_1}{\lambda_2}(a) \geq 2$$

64: proc

$\|\|b\|\|h$	$\|\|b\|\|$	unweighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	weighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	unweighted convergence $\|\|\delta u_{k+1}\|\|/\|\|\delta u_k\|\|$	weighted convergence $\|\|\delta u_{k+1}\|\|/\|\|u_k\|\|$	iteration reduction log / log
347	10^6	0.50	0.29	0.44	0.28	0.64
34.7	10^5	0.50	0.28	0.44	0.28	0.64
3.47	10^4	0.45	0.24	0.39	0.24	0.66
1.10	$10^{3.5}$	0.29	0.21	0.25	0.19	0.83
0.347	10^3	0.26	0.27	0.16	0.17	1.0
0.0347	10^2	0.33	0.38	0.23	0.28	1.2

128:

$\|\|b\|\|h$	$\|\|b\|\|$	unweighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	weighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	unweighted convergence $\|\|\delta u_{k+1}\|\|/\|\|\delta u_k\|\|$	weighted convergence $\|\|\delta u_{k+1}\|\|/\|\|u_k\|\|$	iteration reduction log / log
2450	10^7	0.67	0.50	0.55	0.45	0.75
245	10^6	0.67	0.50	0.55	0.45	0.75
24.5	10^5	0.67	0.50	0.55	0.45	0.75
2.45	10^4	0.58	0.39	0.47	0.35	0.72
0.245	10^3	0.26	0.30	0.18	0.19	1.0
0.0245	10^2	0.39	0.34	0.20	0.24	1.1

256:

$\|\|b\|\|h$	$\|\|b\|\|$	unweighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	weighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	unweighted convergence $\|\|\delta u_{k+1}\|\|/\|\|\delta u_k\|\|$	weighted convergence $\|\|\delta u_{k+1}\|\|/\|\|u_k\|\|$	iteration reduction log / log
173	10^6	0.72	0.56	0.59	0.47	0.70
17.3	10^5	0.72	0.55	0.59	0.47	0.70
1.73	10^4	0.62	0.38	0.51	0.33	0.61
0.173	10^3	0.29	0.31	0.18	0.21	1.1
0.0173	10^2	0.36	0.59	0.26	0.43	1.3

512:

$\|\|b\|\|h$	$\|\|b\|\|$	unweighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	weighted convergence $\|\|r_{k+1}\|\|/\|\|r_k\|\|$	unweighted convergence $\|\|\delta u_{k+1}\|\|/\|\|\delta u_k\|\|$	weighted convergence $\|\|\delta u_{k+1}\|\|/\|\|u_k\|\|$	iteration reduction log / log
123	10^6	0.73	0.61	0.60	0.51	0.76
12.3	10^5	0.73	0.61	0.59	0.50	0.76
1.23	10^4	0.62	0.42	0.51	0.35	0.64
0.123	10^3	0.30	0.33	0.19	0.25	1.2
0.0123	10^2	0.38	0.66	0.28	0.48	1.7

Find $u \in H^2(\Omega)$ s.t

$$-\triangle u - [\beta\ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega_D$$
$$n \cdot \nabla u = 0 \text{ on } \partial\Omega_N$$

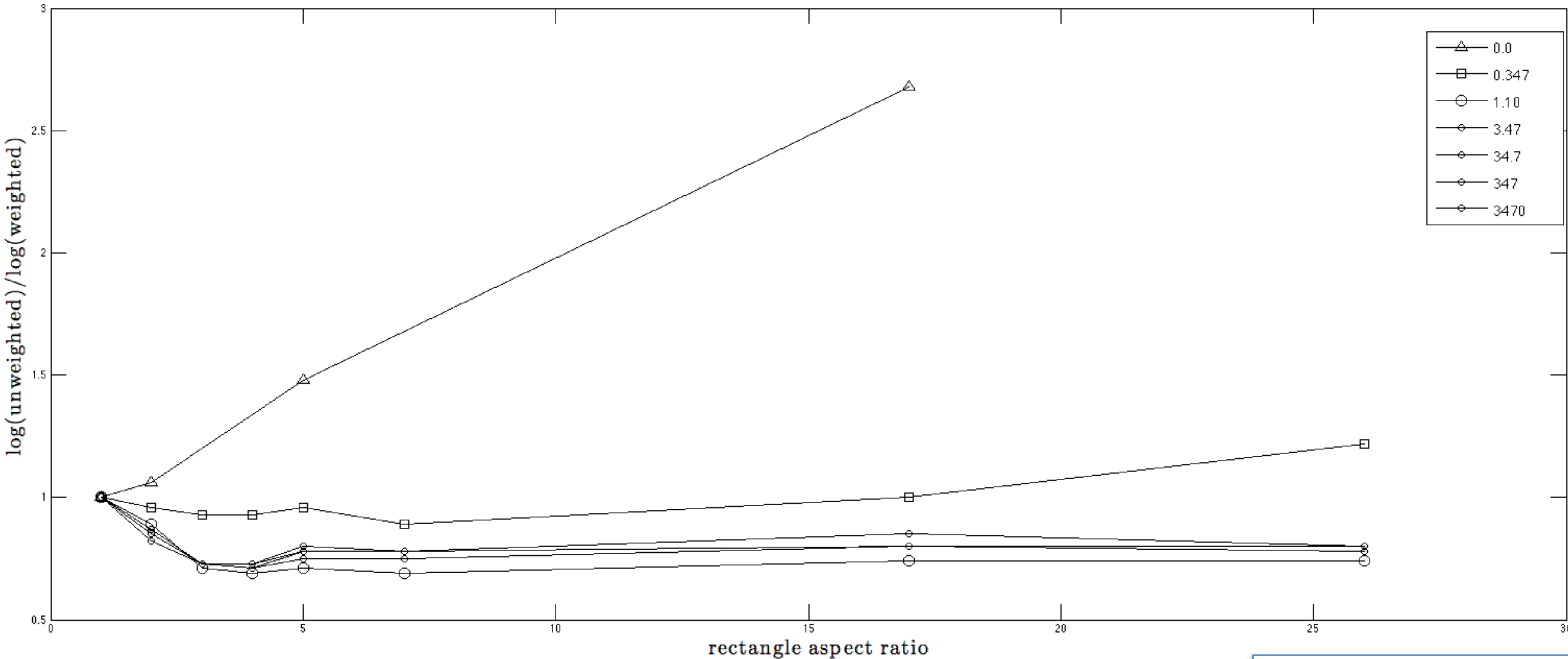
Vary $||b||$ and number of processors.

Keep number of unknowns per proc constant $= 2.0 \times 10^5$

$$\boxed{\frac{||b||h}{||a||} \geq 1}$$

What aspect ratio rectangles to use w/ convection?

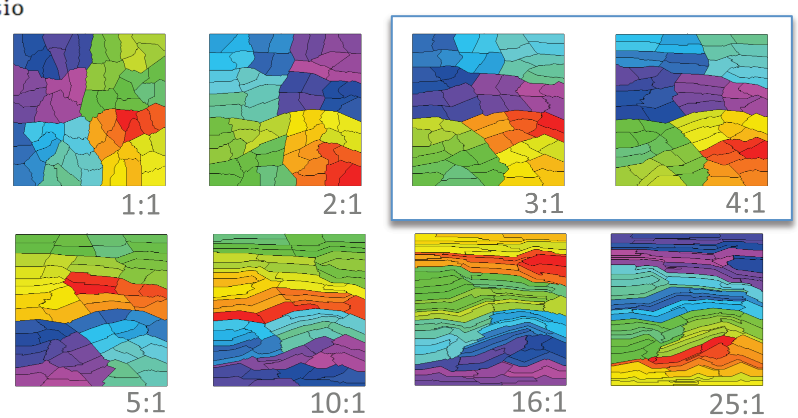
(What convection weighting parameter *s*?)



Find $u \in H^2(\Omega)$ s.t

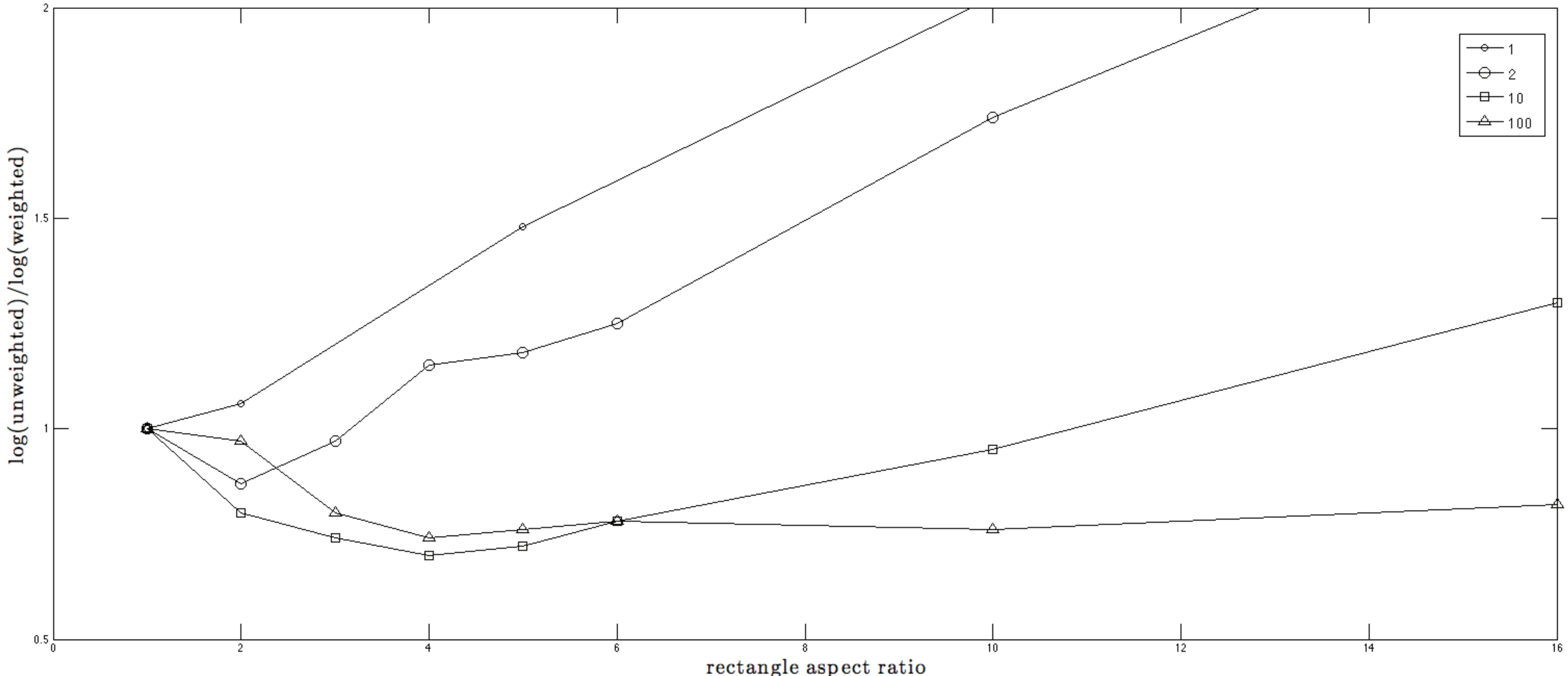
$$-\triangle u - [\beta\ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega$$

Vary $||b||h$ and rectangle aspect ratio.



What aspect ratio rectangles to use w/ diffusion?

(What convection weighting parameter *s*?)

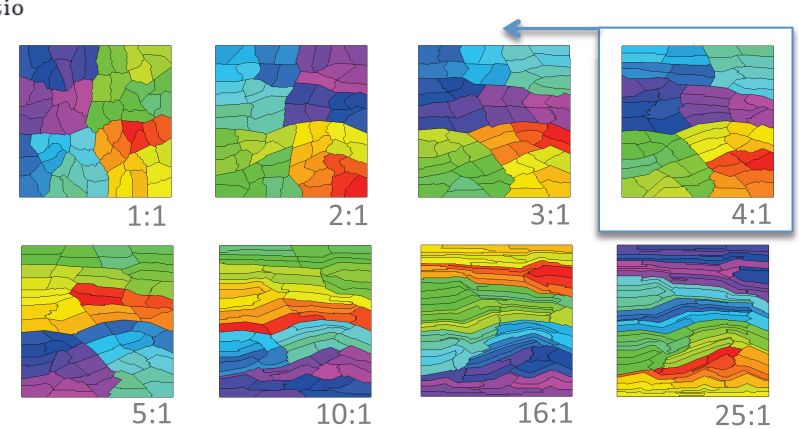


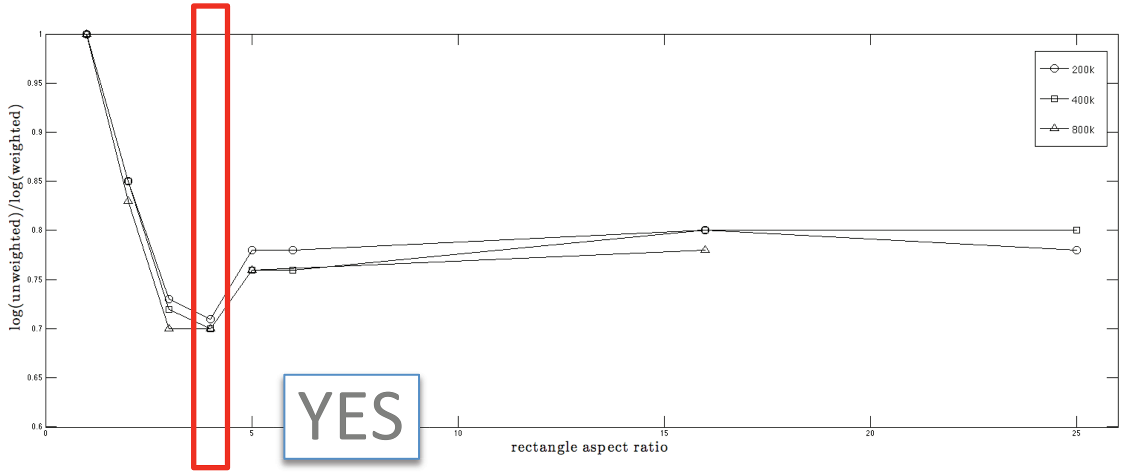
Find $u \in H^2(\Omega)$ s.t.

$$-\nabla \cdot \begin{bmatrix} \alpha & 0 \\ 0 & 1 \end{bmatrix} \nabla u + 1 = 0 \text{ on } \Omega = [0,1]^2$$

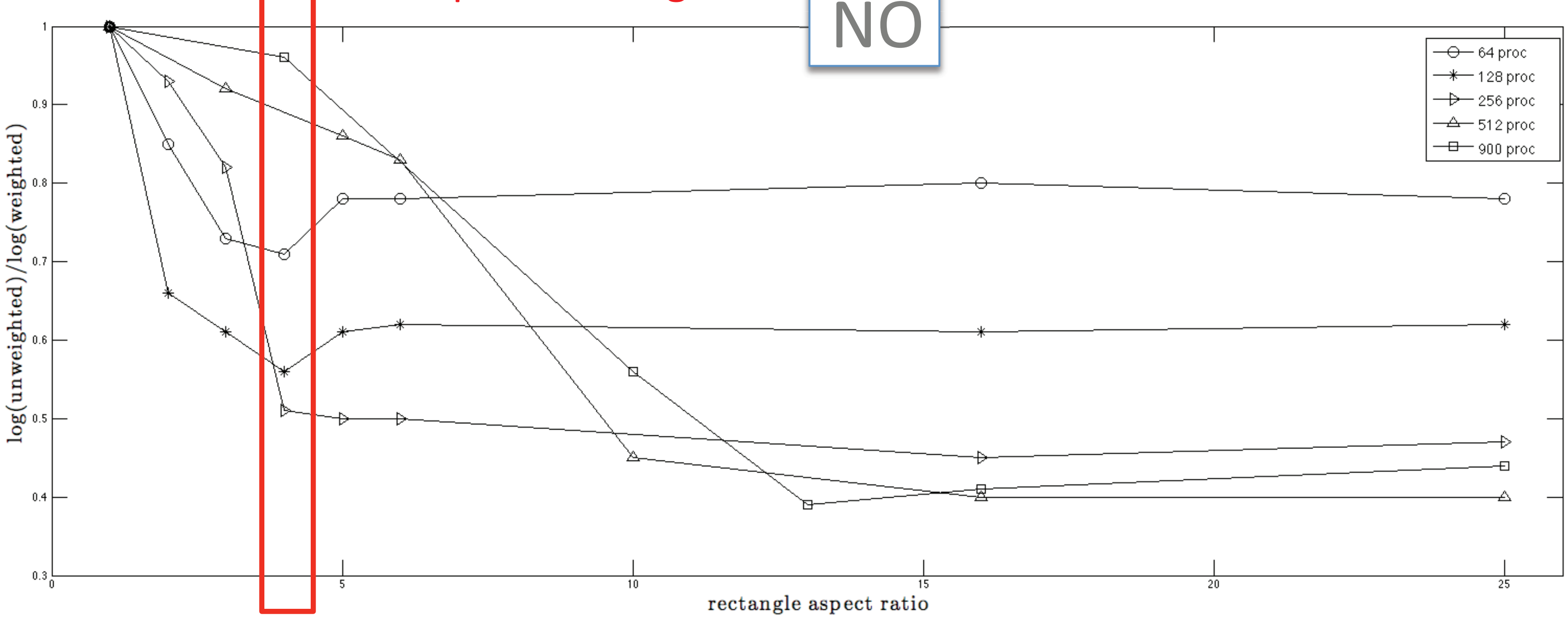
$$u = 0 \text{ on } \partial\Omega$$

Vary α and rectangle aspect ratio.





4:1 aspect rectangles ?



Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - \left[10^5 \;\; 0\right]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$

$$u = 0 \text{ on } \partial\Omega$$

Vary unknowns per processor and number of processors.

Keep $||b||h$ constant by adjusting β.

Is Edge Weighting independent of domain's aspect? Experiment H

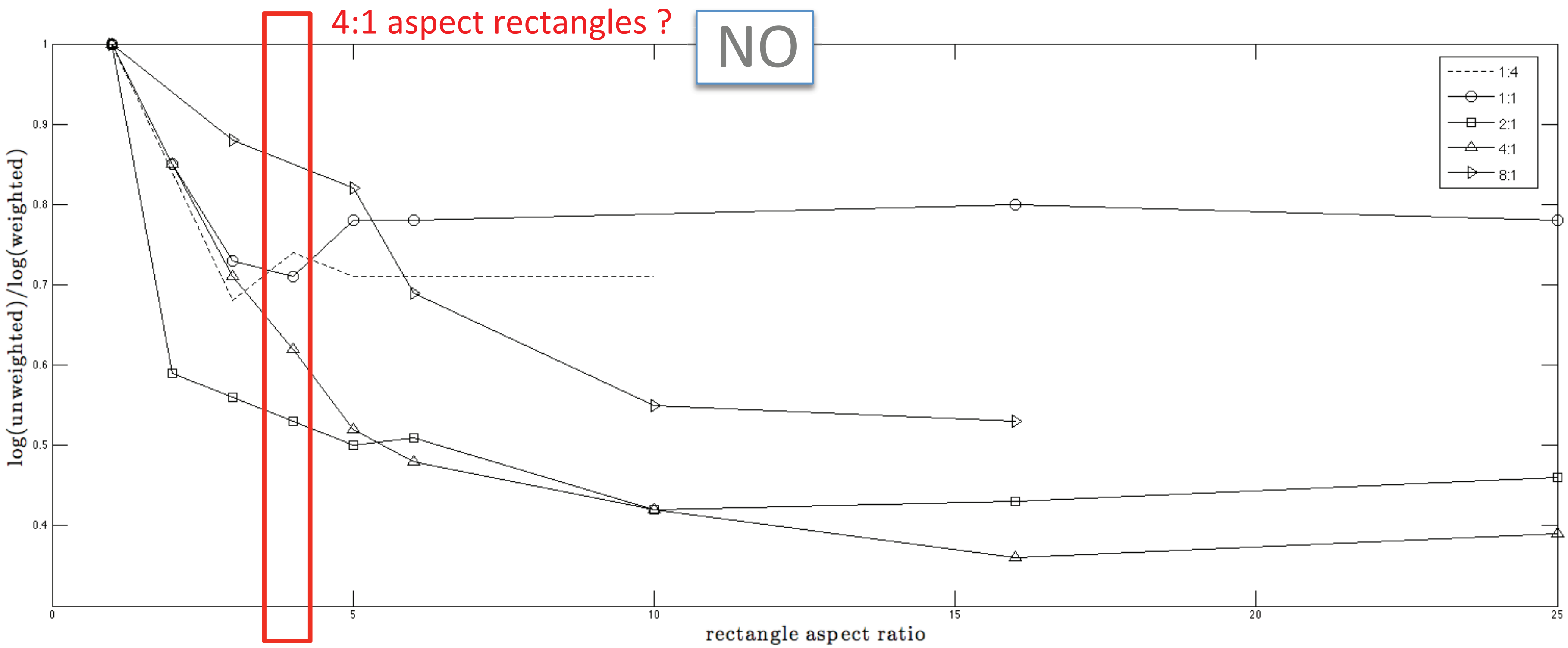
Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - [10^5\ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0, \kappa] \times [0, 1] \in \mathbb{R}^2$$

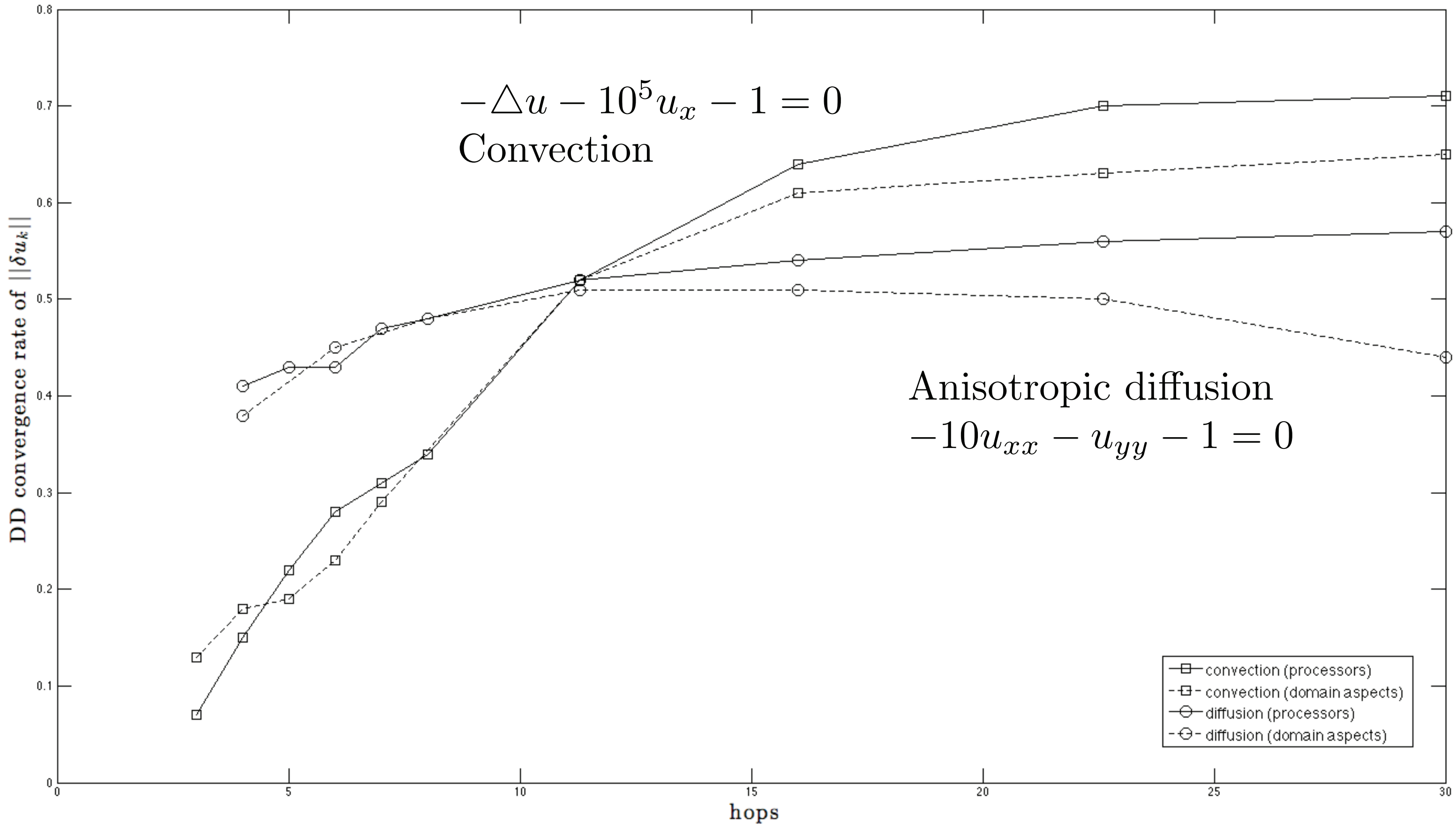
$$u = 0 \text{ on } \partial\Omega$$

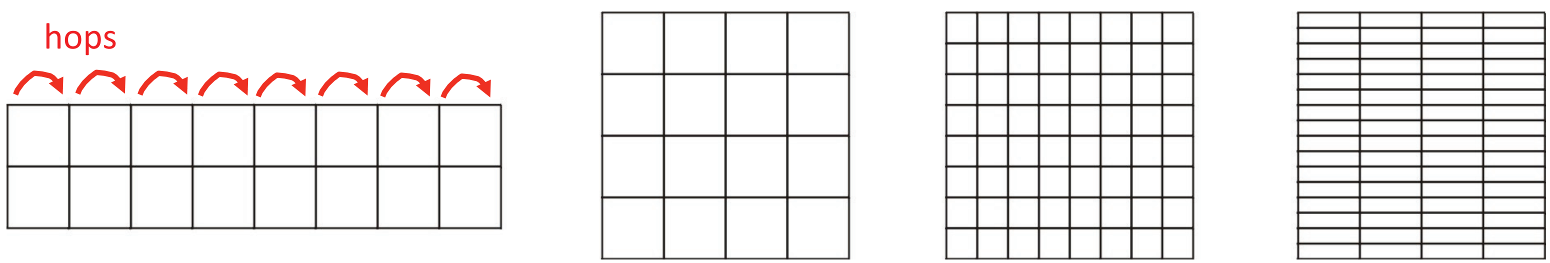
Vary domain's aspect ratio.

Keep $||b||h$ constant by adjusting β.



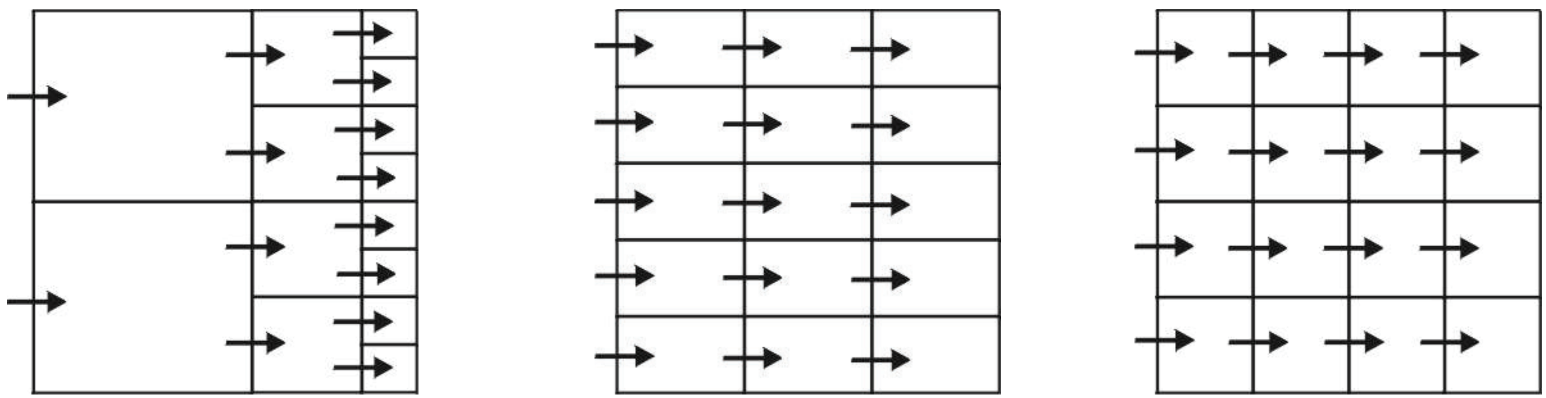
DD Convergence hop dependence





Hop reduction

3 | 3 | 4 hops

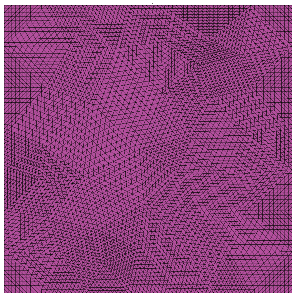


$$2\log_2 h - 1 \qquad \tfrac{3}{4}h \qquad h$$

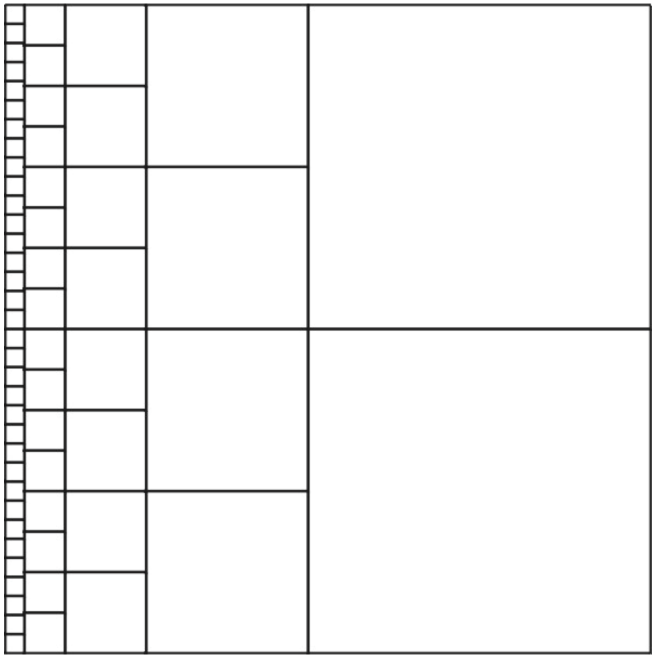
Triangle Weighting Scheme: Flow Weighting (aka Vertex Weighting)

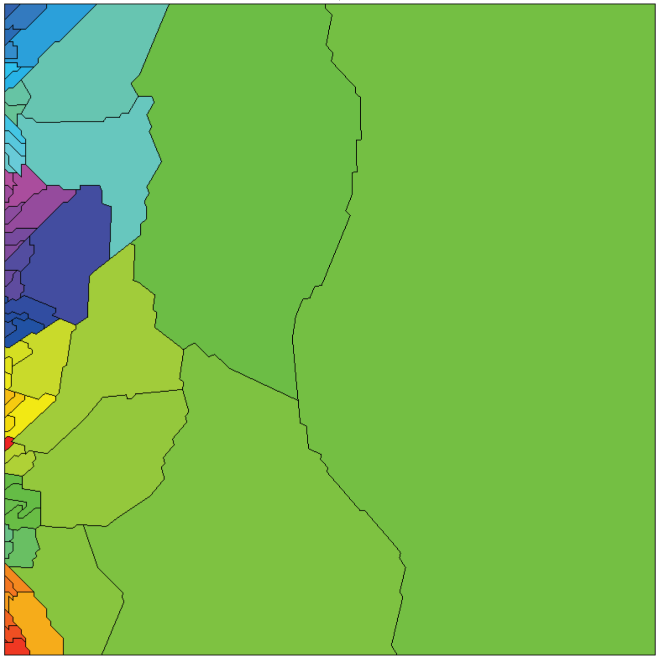
See pg 71-82 for more info

Define triangle weight $w(t_k) = z\left(p(t_k)\right)^{-s}$
where $p(t_k) = (x, y)$ center of triangle t_k
and flow function $z(x, y) : \Omega \to [0, 1] + \epsilon$



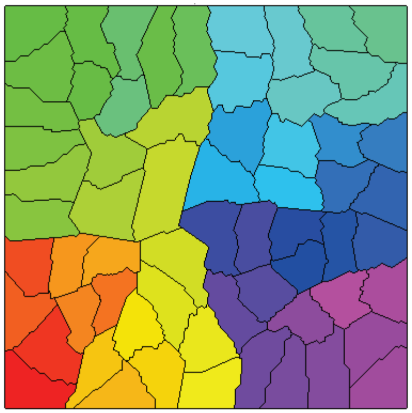
For example, $z(x, y) = x + 10^{-3}$
with $\Omega = [0, 1] \times [0, 1]$ and $s = 1.5$



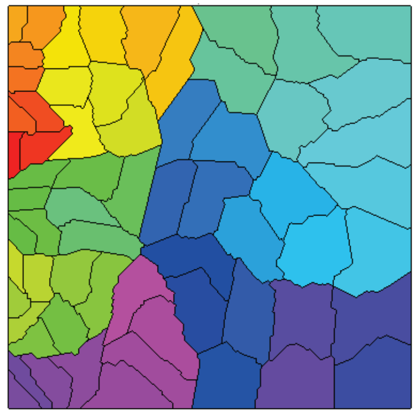


Flow Weighting Parameter *s*

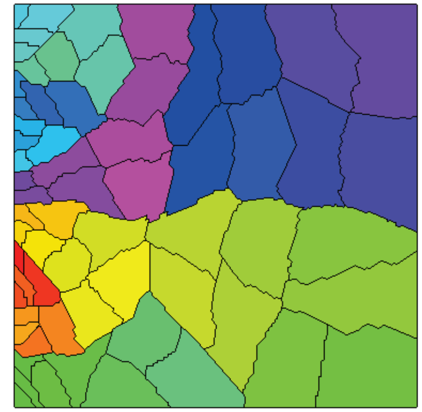
$$w(t_k) = z(p(t_k))^{-s}$$
$$z(x, y) = x + 10^{-3}$$
$$\Omega = [0, 1] \times [0, 1]$$



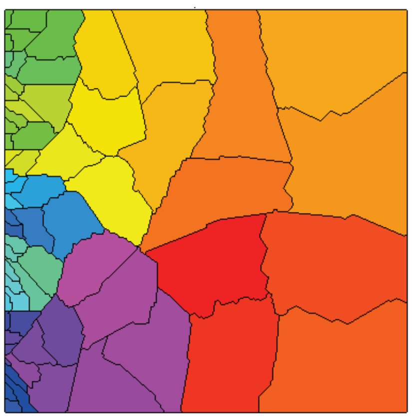
$s = 0$



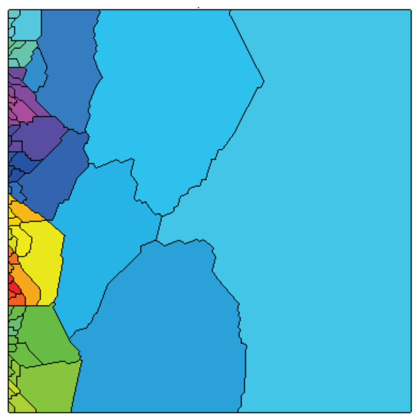
$s = 0.5$



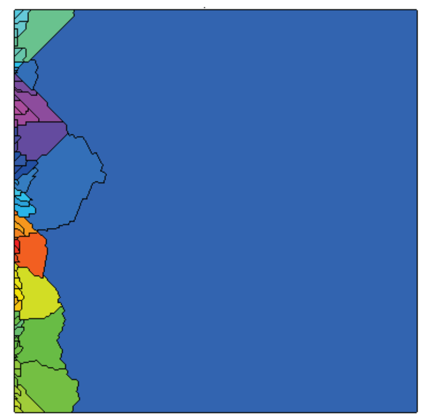
$s = 0.75$



$s = 1$



$s = 1.5$



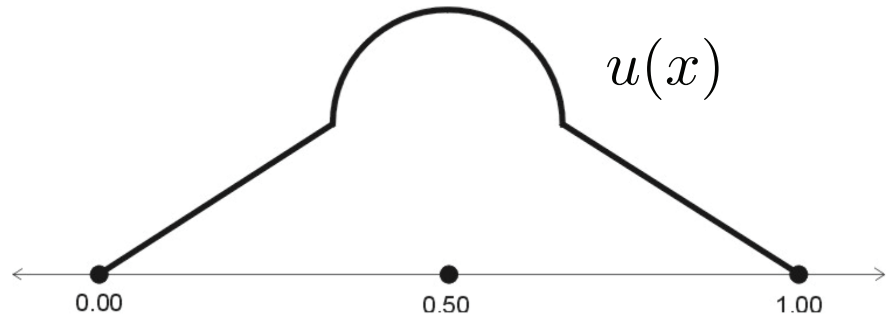
$s = 2$

A Posteriori Error Estimation

See pg 67-70
for more info

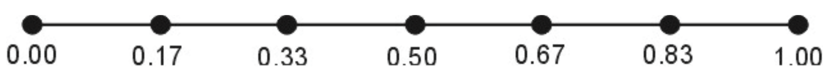
Find $u \in H^2(\Omega)$ s.t.

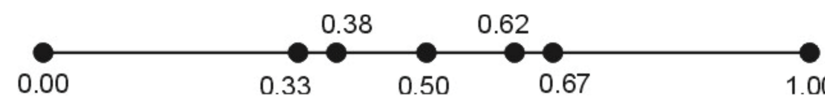
$$Lu = f \text{ on } [0,1] \in \mathbb{R}$$
$$u(0) = u(1) = 0$$

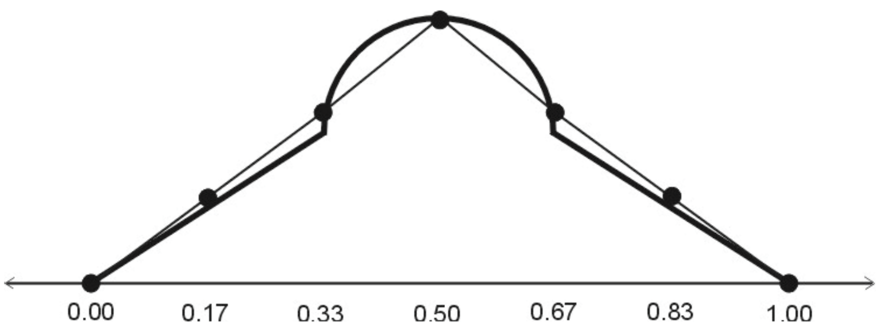


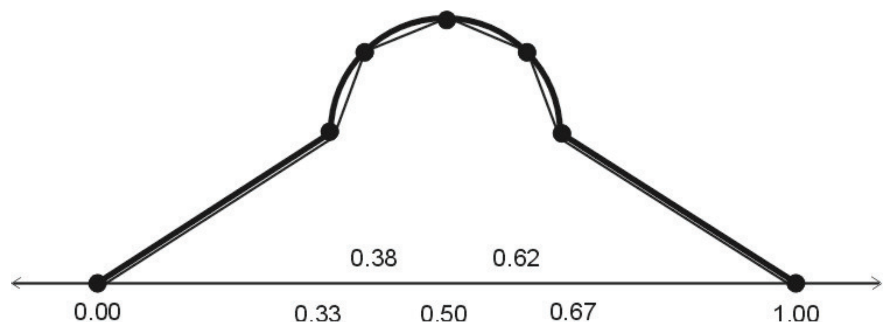
Find $u_h \in S_h \subset H^1(\Omega)$ s.t.

$$u_h = \sum_{k=1}^{5} U_k \psi_k$$









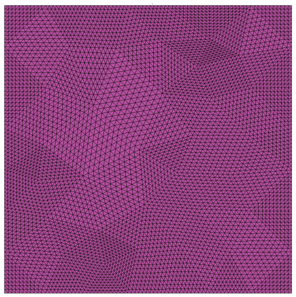
by Randy Bank and Mike Holst

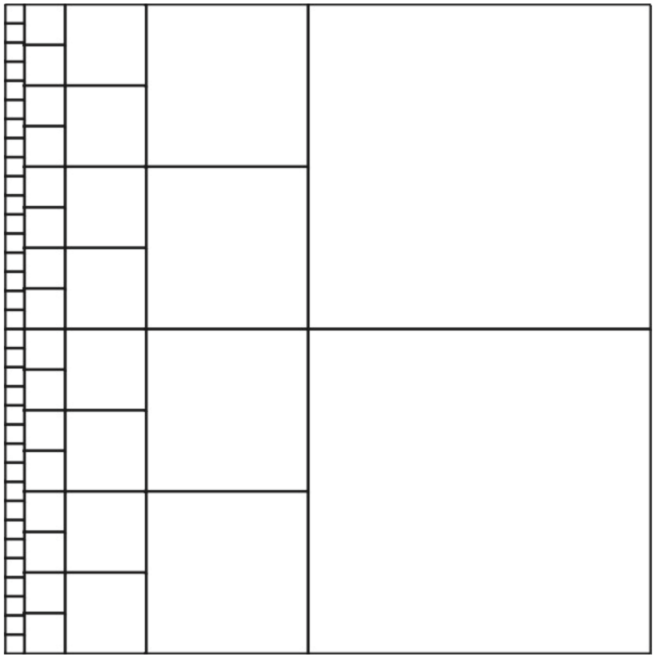
See pg 77-79 for more info

Define triangle weight $w(t_k) = ||\nabla(u - u_h)||_{L^2(t_k)}$

$$\approx ||\nabla(u_2 - u_1)||_{L^2(t_k)}$$

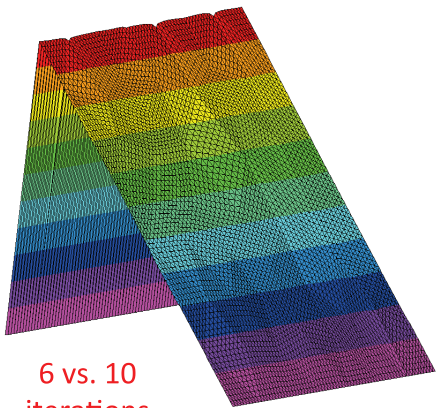
where u_k is the usual Lagrange interpolant





For example, $-\triangle u - [\beta \ 0]^T \nabla u - 1 = 0$ in Ω

$$u = 0 \text{ on } \partial\Omega$$



6 vs. 10 iterations

Find $u \in H^2(\Omega)$ s.t.

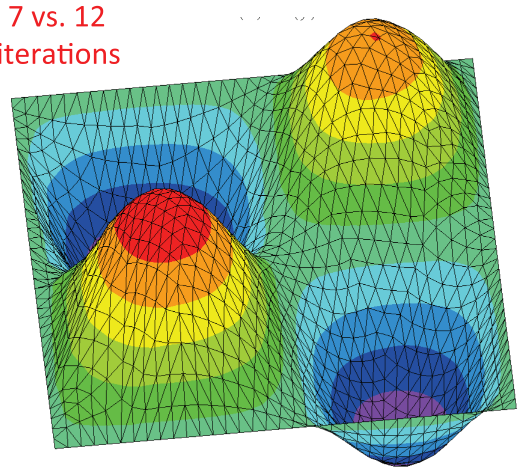
$$-\triangle u - [10^6\ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1] \times [0,1] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega_D = \{0,1\} \times [0,1]$$
$$n \cdot \nabla u = 0 \text{ on } \partial\Omega_N = [0,1] \times \{0,1\}$$

DD convergence of $||r_k||$ and $||\delta u_k||$ were
Weighted: 0.34 and 0.20
Unweighted: 0.50 and 0.44

Error $||u - u_h||_{L^2(\Omega)}$ reduction: 0.40

YES

STOP DD when $||\delta u_k||_{L^2(\Omega)} < \frac{1}{10}||u - u_h||_{L^2(\Omega)}$



7 vs. 12 iterations

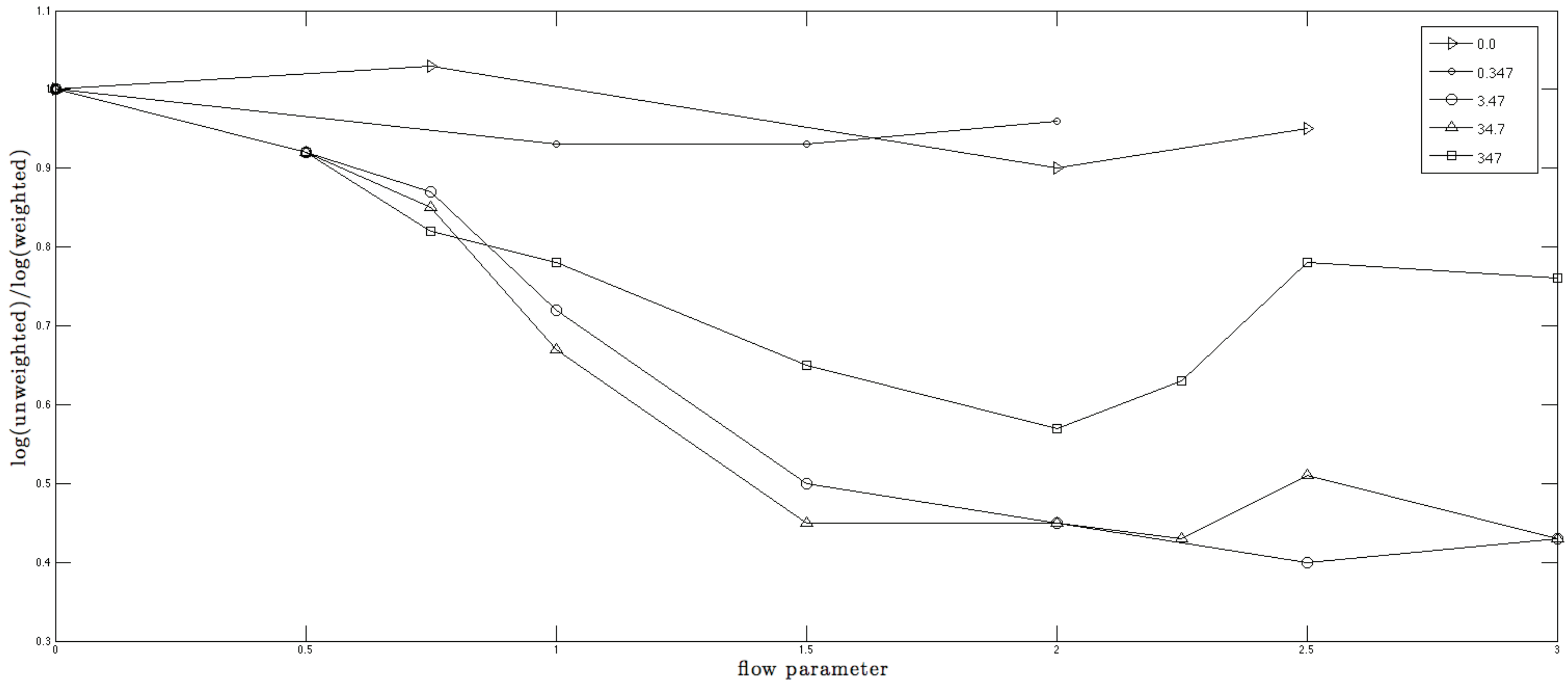
Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - [10^4\ 0]^T \nabla u - f(x,y) = 0 \text{ on } \Omega = [0,2\pi] \times [0,2\pi] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega$$
$$f(x,y) = 2\sin(x)\sin(y) - 10^4\cos(x)\sin(y)$$

DD convergence of $||r_k||$ and $||\delta u_k||$ were
Weighted: 0.20 and 0.14
Unweighted: 0.48 and 0.41

Error $||u - u_h||_{L^2(\Omega)}$ increase: 1.33

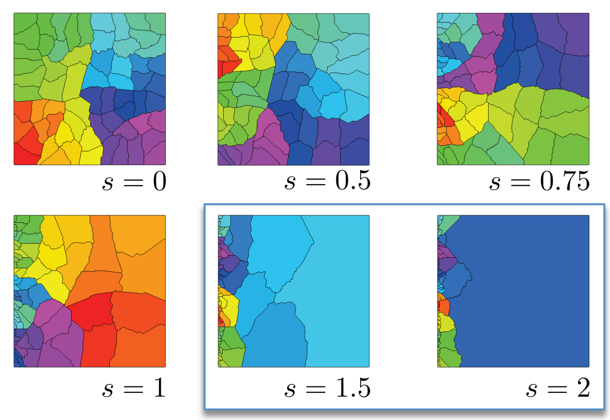
be careful



Find $u \in H^2(\Omega)$ s.t

$$-\triangle u - [\beta\ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega$$

Vary $||b||h$ and flow parameter.

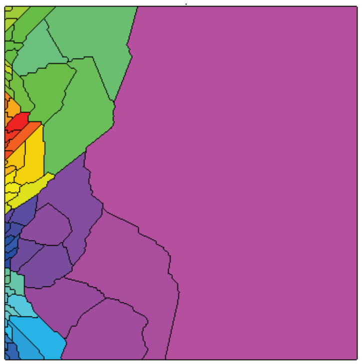


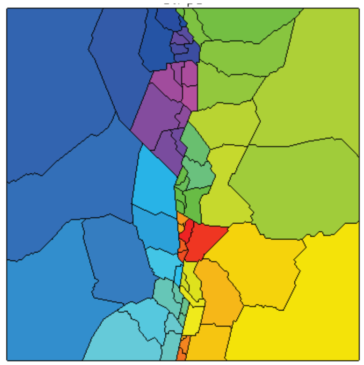
flow function	convergence $\|\|r_k\|\|/\|\|r_0\|\|$	convergence $\|\|\delta_k\|\|/\|\|u_0\|\|$	figure
$z(x,y) = x + \epsilon$	0.36	0.21	1
$z(x,y) = \|x - 0.5\| + \epsilon$	0.35	0.29	2
$z(x,y) = \|x - 1\| + \epsilon$	0.42	0.26	3
$z(x,y) = 1$	0.50	0.44	-
$z(x,y) = y + \epsilon$	0.70	0.51	4
$z(x,y) = \|y - 0.5\| + \epsilon$	0.68	0.53	5
$z(x,y) = \|y - 1\| + \epsilon$	0.70	0.51	6

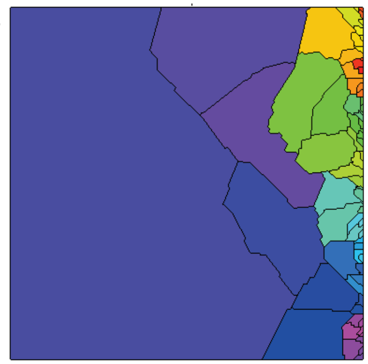
Find $u \in H^2(\Omega)$ s.t.

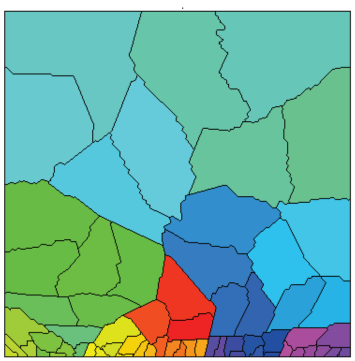
$$-\triangle u - [10^6 \ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1] \times [0,1] \in \mathbb{R}^2$$
$$u = 0 \text{ on } \partial\Omega_D = \{0,1\} \times [0,1]$$
$$n \cdot \nabla u = 0 \text{ on } \partial\Omega_N = [0,1] \times \{0,1\}$$

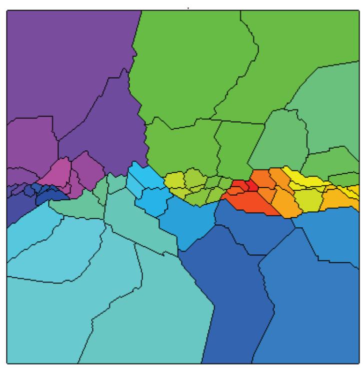
flow parameter $s = 1$

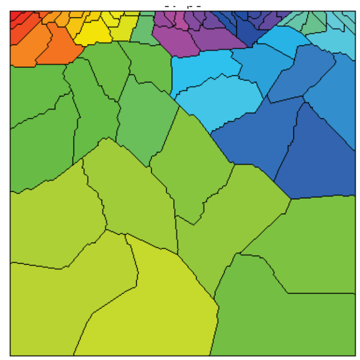
1


2


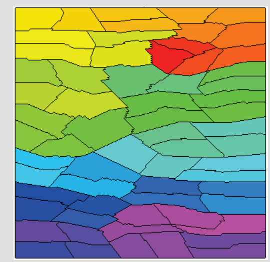
3


4


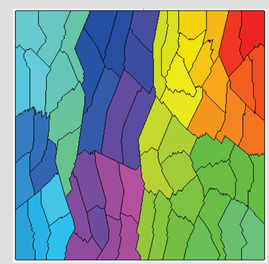
5


6


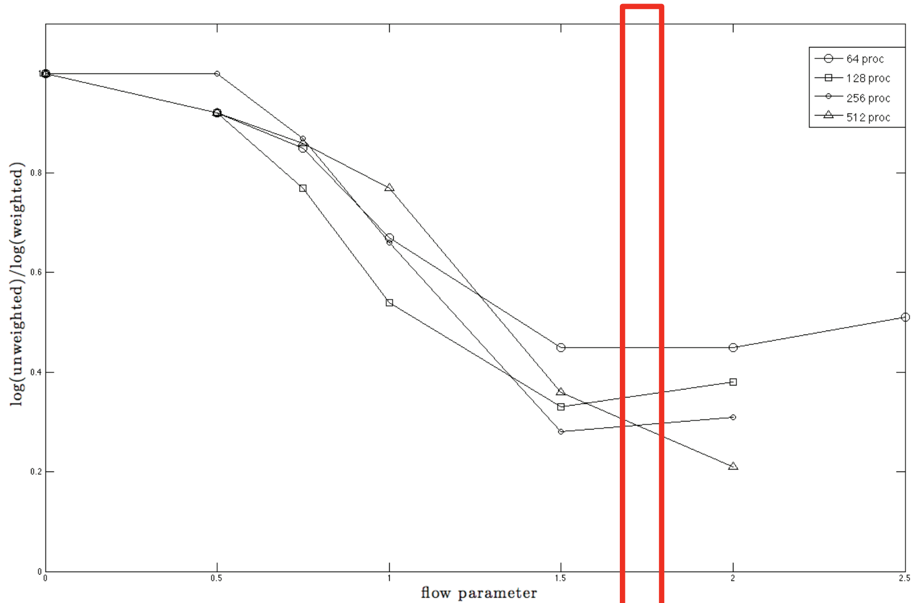
Experiment B



0.29 & 0.28



0.64 & 0.51



Find $u \in H^2(\Omega)$ s.t.

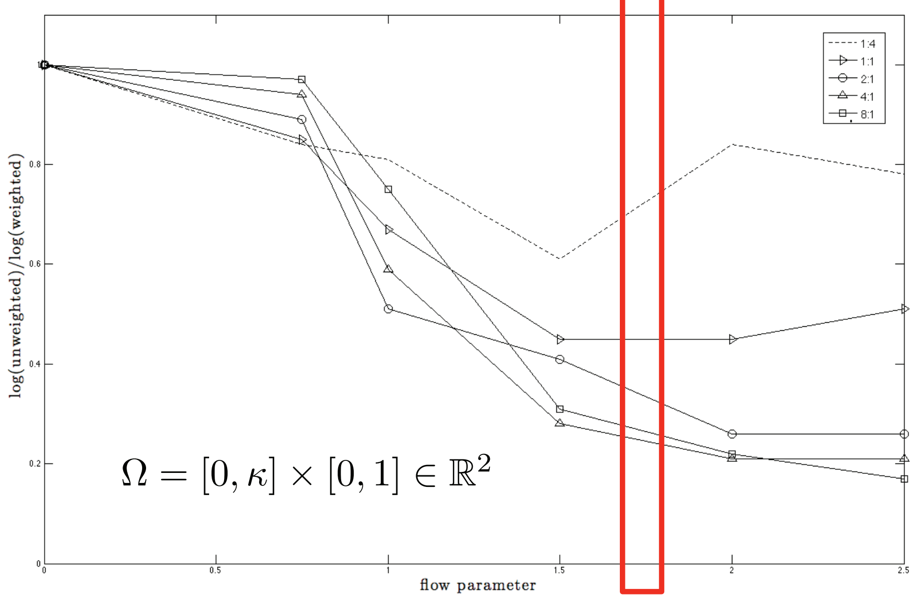
$$-\triangle u - \left[10^5 \ 0\right]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega$$

Vary unknowns per processor and number of processors.

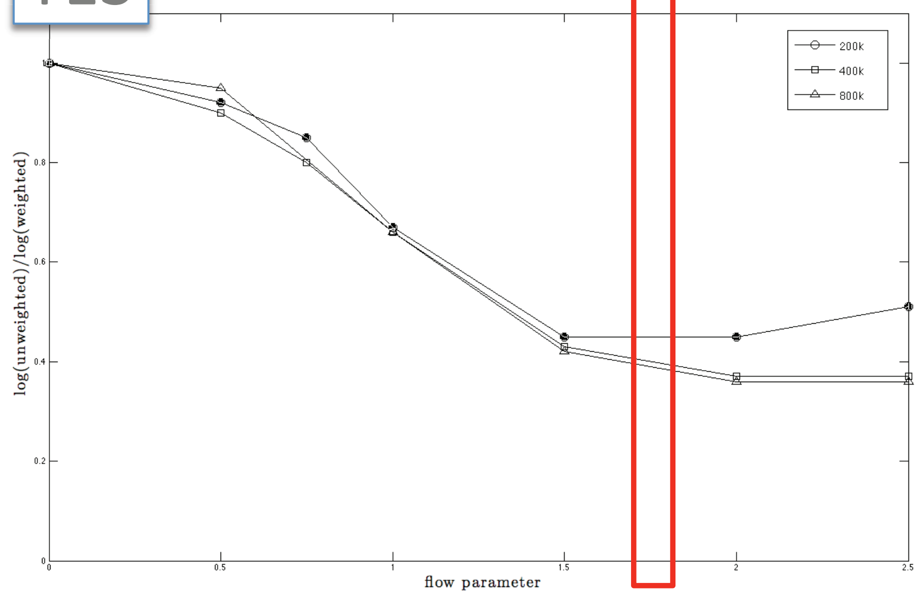
Keep $||b||h$ constant by adjusting β.

YES

1.75 flow parameter ?



$\Omega = [0, \kappa] \times [0, 1] \in \mathbb{R}^2$



DD convergent rate of $||\delta u_k||_{L^2(\Omega)}$

s_c	$s_f = 0.0$	0.5	0.75	1.0	1.5	2.0
0.0	0.34	0.31	0.28	0.20	0.09	0.09
1.0	0.28	0.25	0.22	0.19	0.12	0.13
1.56	0.23	0.21	0.16	0.16	0.13	0.11
1.83	0.22	0.21	0.17	0.15	0.15	0.12
2.0	0.25	0.21	0.18	0.17	0.15	0.13
2.23	0.25	0.21	0.18	0.15	0.18	0.15

Find $u \in H^2(\Omega)$ s.t.

$$-\triangle u - [10^5 \ 0]^T \nabla u - 1 = 0 \text{ on } \Omega = [0,1]^2$$
$$u = 0 \text{ on } \partial\Omega$$

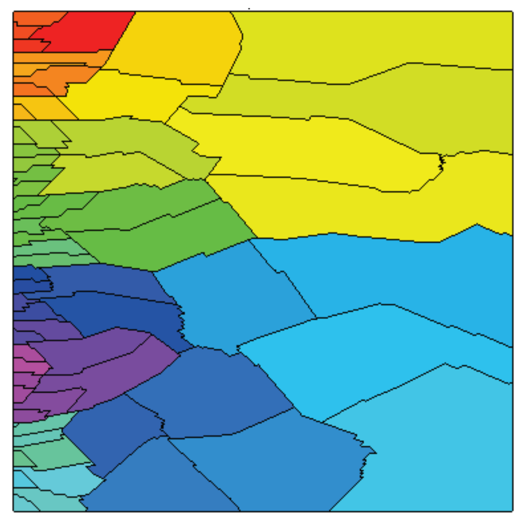
flow function $z(x,y) = x + 10^{-3}$

Vary flow parameter and convection weighting parameter.

NO

DD iterations reduction factor

s_c	$s_f = 0.0$	0.5	0.75	1.0	1.5	2.0
0.0	1.0	0.92	0.85	0.67	0.45	0.45
1.0	0.85	0.78	0.71	0.65	0.51	0.53
1.56	0.73	0.69	0.59	0.59	0.53	0.49
1.83	0.71	0.69	0.61	0.57	0.57	0.51
2.0	0.78	0.69	0.63	0.61	0.57	0.53
2.23	0.78	0.69	0.63	0.57	0.63	0.57



$s_c = 2$ and $s_f = 1$

Domain Decomposition converges faster when applying the Edge or Triangle Weighting schemes presented today. Specifically, convection dominated and anisotropic diffusion elliptic PDEs converge faster regardless of boundary conditions, forcing functions, reaction terms, number of processors, problem size, or domain aspect ratio. Expect a reduction in DD iterations between 0.25 and 0.75

See pg 166-169
for more info

- If Error Weighting can be applied, use it (or Flow Weighting).
- (Don't use Flow Weighting otherwise.)
- Don't combine Edge and Triangle Weighting.
- If Error Weighting cannot be applied, use Stiffness Matrix Weighting (or Convection or Gradient Weighting).
- For Edge Weighting, adjust rectangle aspect ratio if $rP > 400$ (where r is the domain aspect ratio in the direction of dependence and P is the number of processors).

Thank you everyone for coming ☺

Special thanks to my committee:
Professor Randolph E. Bank, Chair
Professor Scott B. Baden
Professor David J. Benson
Professor Michael Holst
Professor Melvin Leok