Multigrid Solution of Discrete Convection-Diffusion Equations

Alison Ramage
Dept of Mathematics
University of Strathclyde
Glasgow, Scotland

UNIVERSITY OF STRATHCLYDE

UNIVERSITY OF MARYLAND

Howard Elman
Dept of Computer Science
University of Maryland
College Park, MD, USA

Convection-Diffusion in 2D

$$\begin{aligned} -\epsilon \nabla^2 u(x,y) + \mathbf{w}.\nabla u(x,y) &= f(x,y) \quad \text{in} \quad \Omega \in \mathbb{R}^2 \\ u(x,y) &= g \quad \text{on} \quad \partial\Omega \end{aligned}$$

divergence-free convective velocity (‘wind’) $\mathbf{w}$

diffusion parameter $\epsilon << 1$

discretisation parameter h

mesh Péclet number $P_h = \frac{\|\mathbf{w}\| h}{2\epsilon}$

Boundary Layers and Oscillations

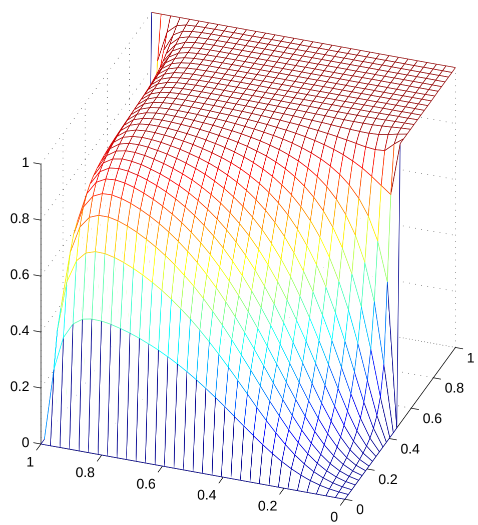
- Galerkin finite element method

$$\epsilon(\nabla u_h, \nabla v_h) + (\mathbf{w} \cdot \nabla u_h, v_h) = (f, v_h) \quad \forall v_h \in V_h$$

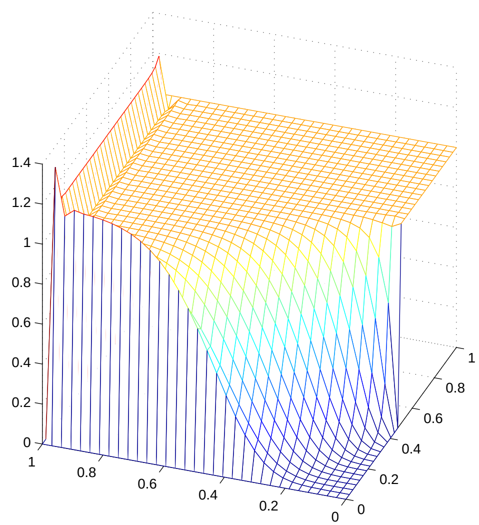
- solution features:

  exponential and **characteristic** boundary layers

- **oscillations** observed in discrete solutions for $P_h > 1$



$P_h = 0.5$



$P_h = 2$

Streamline Diffusion Method

streamline diffusion FEM, square bilinear elements

$$\begin{aligned}\epsilon(\nabla u_h, \nabla v_h) &+ (\mathbf{w}\cdot\nabla u_h, v_h) + \frac{\delta h}{||\mathbf{w}||}(\mathbf{w}\cdot\nabla u_h, \mathbf{w}\cdot\nabla v_h) \\ &= (f, v_h) + \frac{\delta h}{||\mathbf{w}||}(f, \mathbf{w}\cdot\nabla v_h) \quad \forall v_h \in V_h\end{aligned}$$

Streamline Diffusion Method

streamline diffusion FEM, square bilinear elements

$$
\begin{aligned}
\epsilon(\nabla u_h, \nabla v_h) &+ (\mathbf{w}\cdot\nabla u_h, v_h) + \frac{\delta h}{||\mathbf{w}||}(\mathbf{w}\cdot\nabla u_h, \mathbf{w}\cdot\nabla v_h) \\
&= (f, v_h) + \frac{\delta h}{||\mathbf{w}||}(f, \mathbf{w}\cdot\nabla v_h) \quad \forall v_h \in V_h
\end{aligned}
$$

- $P_h \le 1: \quad \delta = 0$ **Galerkin FEM**
- $P_h > 1: \quad \delta = \frac{1}{2} - \frac{\epsilon}{h}$ **Streamline Diffusion**

Multigrid Ideas

- fine grid (h), coarse grid ($2h$)

Multigrid Ideas

- fine grid (h), coarse grid ($2h$)
- decompose a grid function into components in two subspaces

approximate inverse operator
for components in subspace 1

smoothing iteration
rapidly reduces error components in subspace 2

Multigrid Ideas

- fine grid (h),   coarse grid ($2h$)

- decompose a grid function into components in two subspaces

  approximate inverse operator
  for components in subspace 1

  smoothing iteration
  rapidly reduces error components in subspace 2

- recursive process on nested grids

Multigrid Ideas

- fine grid (h),   coarse grid ($2h$)

- decompose a grid function into components in two subspaces

  approximate inverse operator
  for components in subspace 1

  smoothing iteration
  rapidly reduces error components in subspace 2

- recursive process on nested grids

- **optimal** in the sense of obtaining convergence rate independent of h

Issues for Convection-Diffusion

- **approximation**: choice of discretisation
  - oscillations on coarser grids?
  - grid transfer operators?

Issues for Convection-Diffusion

- **approximation**: choice of discretisation
  - oscillations on coarser grids?
  - grid transfer operators?
- **smoothing**: choice of relaxation method
  - direction of flow?
  - circular flows?

Issues for Convection-Diffusion

- **approximation**: choice of discretisation
  - oscillations on coarser grids?
  - grid transfer operators?
- **smoothing**: choice of relaxation method
  - direction of flow?
  - circular flows?
- multigrid can be implemented effectively for convection-diffusion problems

Convergence Analysis

- standard Poisson-type convergence analysis fails
- ideas for convection-diffusion less well-developed

Convergence Analysis

- standard Poisson-type convergence analysis fails
- ideas for convection-diffusion less well-developed
- various successful approaches
  - **perturbation arguments**
    Bank (1981), Bramble, Pasciak and Xu (1988), Mandel (1986), Wang (1993)
  - **matrix-based methods**
    Reusken (2002), Olshanskii and Reusken (2002)

Multigrid Method

- two-grid method: N_f (fine grid), N_c (coarse grid)

Multigrid Method

- two-grid method: N_f (fine grid), N_c (coarse grid)
- coefficient matrices: A_f (fine grid), A_c (coarse grid)

direct discretisation on coarse grid

Multigrid Method

- two-grid method: N_f (fine grid), N_c (coarse grid)
- coefficient matrices: A_f (fine grid), A_c (coarse grid)
- prolongation: bilinear interpolation P
- restriction: transpose of prolongation P^T

Multigrid Method

- two-grid method: N_f (fine grid), N_c (coarse grid)
- coefficient matrices: A_f (fine grid), A_c (coarse grid)
- prolongation: bilinear interpolation P
- restriction: transpose of prolongation P^T
- smoothing: line Gauss-Seidel S_A
- ν steps of pre-smoothing, no post-smoothing

Multigrid Convergence

- algebraic error $\mathbf{e}_k = \hat{\mathbf{u}} - \mathbf{u}_k$
- two-grid iteration matrix $M = (I - PA_c^{-1}P^T A_f)S_A^\nu$
- error equation $\mathbf{e}_k = M\mathbf{e}_{k-1} = M^k\mathbf{e}_0$
- convergence?

$$\|\mathbf{e}_k\| \leq \|M\|^k \|\mathbf{e}_0\|$$

convergence if $\|M\| < 1$

Two-Grid Convergence Analysis

AIM: find an upper bound for

$$\|M\|_2 = \|(I - PA_c^{-1}P^T A_f)S_A^{\nu}\|_2$$

Two-Grid Convergence Analysis

AIM: find an upper bound for

$$\|M\|_2 = \|(I - PA_c^{-1}P^T A_f)S_A^\nu\|_2$$

- Approach 1: write

$$M = (A_f^{-1} - PA_c^{-1}P^T)(A_f S_A^\nu) = M_A M_S$$

and bound $\|M_A\|_2$, $\|M_S\|_2$ separately

Two-Grid Convergence Analysis

AIM: find an upper bound for

$$\|M\|_2 = \|(I - PA_c^{-1}P^T A_f)S_A^\nu\|_2$$

- Approach 1: write

$$M = (A_f^{-1} - PA_c^{-1}P^T)(A_f S_A^\nu) = M_A M_S$$

  and bound $\|M_A\|_2$, $\|M_S\|_2$ separately

- Approach 2: bound $\|M\|_2$ directly

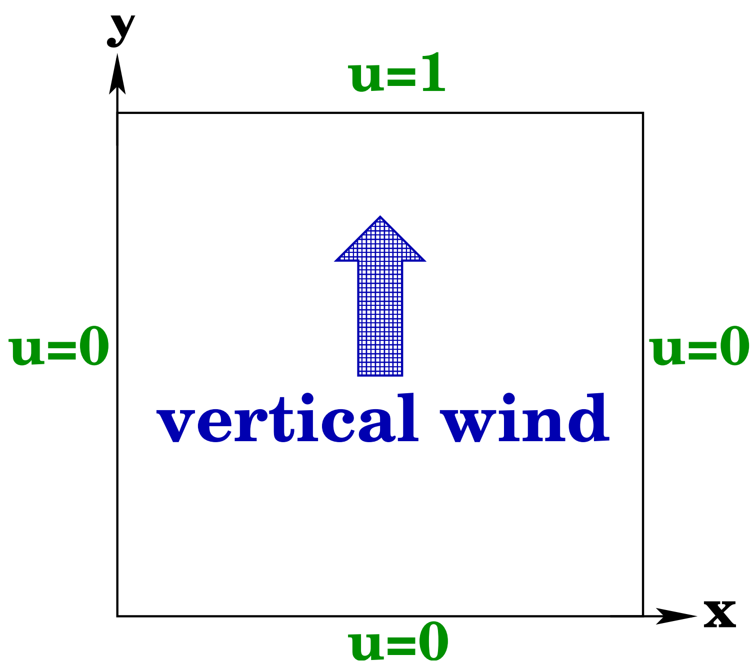
Model Problem

grid-aligned flow with vertical wind and $f = 0$

$$-\epsilon \nabla^2 u(x,y) + (0,1).\nabla u(x,y) = 0$$

Dirichlet boundary conditions

square bilinear elements



Computational Molecule

parameters $\quad h,\ \epsilon,\ \delta$

$$
\begin{array}{llcccc}
M_2: & -\frac{1}{12}[(2\delta-1)h+4\epsilon] & & -\frac{1}{3}[(2\delta-1)h+\epsilon] & & -\frac{1}{12}[(2\delta-1)h+4\epsilon] \\
 & & \nwarrow & \uparrow & \nearrow & \\
M_1: & \frac{1}{3}(\delta h-\epsilon) & \longleftarrow & \frac{4}{3}(\delta h+2\epsilon) & \longrightarrow & \frac{1}{3}(\delta h-\epsilon) \\
 & & \swarrow & \downarrow & \searrow & \\
M_3: & -\frac{1}{12}[(2\delta+1)h+4\epsilon] & & -\frac{1}{3}[(2\delta+1)h+\epsilon] & & -\frac{1}{12}[(2\delta+1)h+4\epsilon]
\end{array}
$$

symmetric stencil

Coefficient Matrix

$$A = \begin{bmatrix} M_1 & M_2 & & & \mathbf{0} \\ M_3 & M_1 & M_2 & & \\ & \ddots & \ddots & \ddots & \\ & & M_3 & M_1 & M_2 \\ \mathbf{0} & & & M_3 & M_1 \end{bmatrix}$$

Coefficient Matrix

$$A = \begin{bmatrix} M_1 & M_2 & & & \mathbf{0} \\ M_3 & M_1 & M_2 & & \\ & \ddots & \ddots & \ddots & \\ & & M_3 & M_1 & M_2 \\ \mathbf{0} & & & M_3 & M_1 \end{bmatrix}$$

eigenvectors and eigenvalues:

$$M_1 \mathrm{v}_j = \lambda_j \mathrm{v}_j, \quad \lambda_j = m_{1c} + 2m_{1r} \cos \frac{j\pi}{N_f}$$
$$M_2 \mathrm{v}_j = \sigma_j \mathrm{v}_j, \quad \sigma_j = m_{2c} + 2m_{2r} \cos \frac{j\pi}{N_f}$$
$$M_3 \mathrm{v}_j = \gamma_j \mathrm{v}_j, \quad \gamma_j = m_{3c} + 2m_{3r} \cos \frac{j\pi}{N_f}$$

$$\mathrm{v}_j = \sqrt{\frac{2}{N_f}} \left[\sin \frac{j\pi}{N_f}, \quad \sin \frac{2j\pi}{N_f}, \quad \ldots, \sin \frac{(N_f - 1)j\pi}{N_f} \right]^T$$

Transformation: Coefficient Matrix (1)

N_f^2 elements, n_f^2 unknowns $(n_f = N_f - 1)$

$$\hat{V}_f = \left[\mathrm{v}_1 \mathrm{v}_2 \ldots \mathrm{v}_{n_f} \right], \qquad V_f = \mathtt{diag}(\hat{V}_f, \ldots, \hat{V}_f)$$

$$M_1 \hat{V}_f = \hat{V}_f \Lambda, \qquad M_2 \hat{V}_f = \hat{V}_f \Sigma, \qquad M_3 \hat{V}_f = \hat{V}_f \Gamma$$

Transformation: Coefficient Matrix (1)

N_f^2 elements, n_f^2 unknowns $(n_f = N_f - 1)$

$$\hat{V}_f = \left[\mathrm{v}_1 \mathrm{v}_2 \ldots \mathrm{v}_{n_f}\right], \qquad V_f = \mathtt{diag}(\hat{V}_f, \ldots, \hat{V}_f)$$

$$M_1 \hat{V}_f = \hat{V}_f \Lambda, \qquad M_2 \hat{V}_f = \hat{V}_f \Sigma, \qquad M_3 \hat{V}_f = \hat{V}_f \Gamma$$

$$V_f^T A_f V_f = \hat{T}_f = \begin{bmatrix} \Lambda & \Sigma & & & \mathbf{0} \\ \Gamma & \Lambda & \Sigma & & \\ & \ddots & \ddots & \ddots & \\ & & \Gamma & \Lambda & \Sigma \\ \mathbf{0} & & & \Gamma & \Lambda \end{bmatrix}$$

Transformation: Coefficient Matrix (2)

permute into tridiagonal form:

$$\Pi_f^T \hat{T}_f \Pi_f = T_f = \begin{bmatrix} T_1 & & & & 0 \\ & T_2 & & & \\ & & \ddots & & \\ & & & T_{n_f-1} & \\ 0 & & & & T_{n_f} \end{bmatrix}$$

$$T_j = \texttt{tridiag}(\gamma_j, \lambda_j, \sigma_j)$$

$$A_f = Q_f T_f Q_f^T \qquad Q_f = V_f \Pi_f$$

Transformation: Coefficient Matrix (2)

permute into tridiagonal form:

$$\Pi_f^T \hat{T}_f \Pi_f = T_f = \begin{bmatrix} T_1 & & & & 0 \\ & T_2 & & & \\ & & \ddots & & \\ & & & T_{n_f-1} & \\ 0 & & & & T_{n_f} \end{bmatrix}$$

$$T_j = \mathtt{tridiag}(\gamma_j, \lambda_j, \sigma_j)$$

$$A_f = Q_f T_f Q_f^T \qquad Q_f = V_f \Pi_f$$

coarse grid: $A_c = Q_c T_c Q_c^T \qquad Q_c = V_c \Pi_c$

Transformation: Smoothing Matrix

block matrix splitting: $A_f = D_A - L_A - U_A$

Gauss-Seidel smoothing matrix:

$$S_A = (D_A - L_A)^{-1} U_A = I - (D_A - L_A)^{-1} A_f$$

Transformation: Smoothing Matrix

block matrix splitting: $A_f = D_A - L_A - U_A$

Gauss-Seidel smoothing matrix:

$$S_A = (D_A - L_A)^{-1} U_A = I - (D_A - L_A)^{-1} A_f$$

transformation:

$$S_A = Q_f S_T Q_f^T$$

where $S_T = I - (D_T - L_T)^{-1} T_f$ is block-diagonal

Transformation: Prolongation Matrix

2D prolongation matrix: $\quad P = L \otimes L$

$$L^T = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{2} & & & & & & \\ & & \frac{1}{2} & 1 & \frac{1}{2} & & & & \\ & & & & & \ddots & & & \\ & & & & & & \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}$$

Transformation: Prolongation Matrix

2D prolongation matrix: $P = L \otimes L$

$$L^T = \begin{bmatrix} \frac{1}{2} & 1 & \frac{1}{2} & & & & & & \\ & & \frac{1}{2} & 1 & \frac{1}{2} & & & & \\ & & & & & \ddots & & & \\ & & & & & & \frac{1}{2} & 1 & \frac{1}{2} \end{bmatrix}$$

transformation: $Q_f = (I_f \otimes \hat{V}_f)\Pi_f, \quad Q_c = (I_c \otimes \hat{V}_c)\Pi_c$

$$\bar{P} = Q_f^T P Q_c = \mathcal{A}^T \otimes L$$

$$\mathcal{A} = \begin{bmatrix} \alpha_1 & & & & & & & \alpha_{N_f-1} \\ & \alpha_2 & & & & & \alpha_{N_f-2} & \\ & & \ddots & & & \iddots & & \\ & & & \alpha_{n_c} & 0 & \alpha_{N_f-n_c} & & \end{bmatrix}$$

Transformation: Iteration Matrix (1)

$$\begin{aligned}
M &= (I - PA_c^{-1}P^T A_f)S_A^\nu \\
&= (I - PQ_cT_c^{-1}Q_c^T P^T Q_f T_f Q_f^T)S_A^\nu \\
&= Q_f(I - \bar{P}T_c^{-1}\bar{P}^T T_f)Q_f^T(Q_f S_T Q_f^T)^\nu \\
&= Q_f\left(I - \bar{P}T_c^{-1}\bar{P}^T T_f\right)S_T^\nu Q_f^T \\
\Rightarrow M &= Q_f \bar{M} Q_f^T
\end{aligned}$$

where $\bar{M} = \left(I - \bar{P}T_c^{-1}\bar{P}^T T_f\right)S_T^\nu$

Transformation: Iteration Matrix (1)

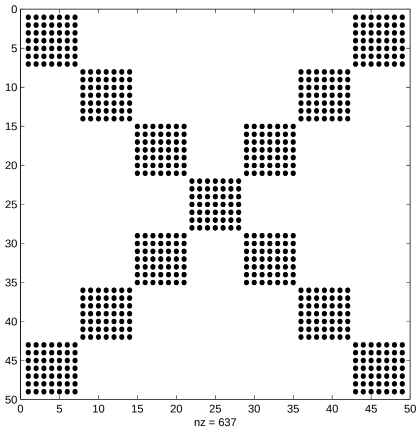
$$\begin{aligned} M &= (I - PA_c^{-1}P^T A_f)S_A^\nu \\ &= (I - PQ_cT_c^{-1}Q_c^T P^T Q_f T_f Q_f^T)S_A^\nu \\ &= Q_f(I - \bar{P}T_c^{-1}\bar{P}^T T_f)Q_f^T (Q_f S_T Q_f^T)^\nu \\ &= Q_f \left(I - \bar{P}T_c^{-1}\bar{P}^T T_f\right) S_T^\nu Q_f^T \\ \Rightarrow M &= Q_f \bar{M} Q_f^T \end{aligned}$$

where $\bar{M} = \left(I - \bar{P}T_c^{-1}\bar{P}^T T_f\right) S_T^\nu$

Q_f is orthogonal:

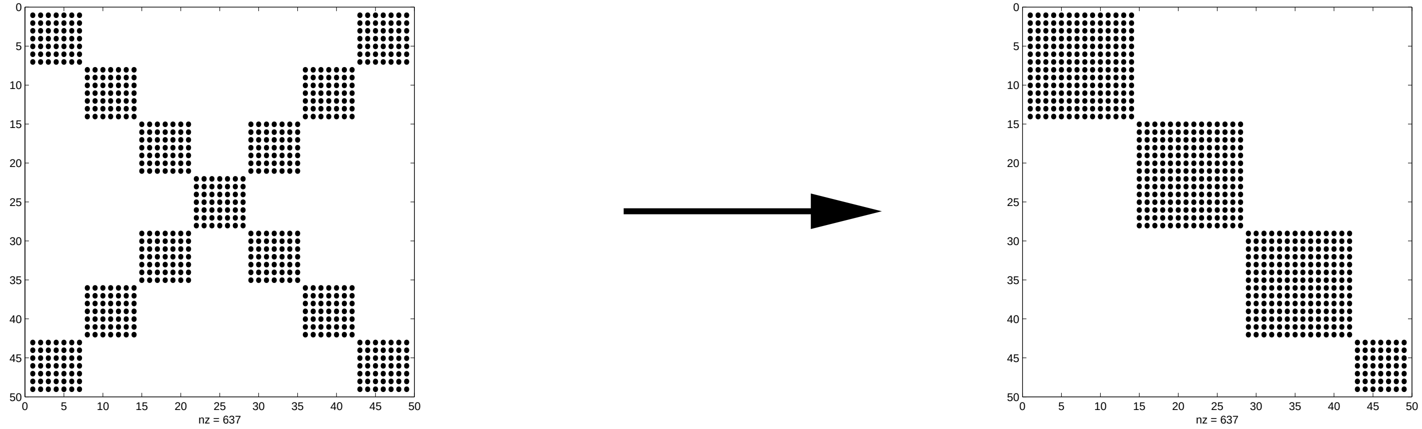
$$\|M\|_2 = \|\bar{M}\|_2$$

Transformation: Iteration Matrix (2)



$$\begin{bmatrix} \mathbf{B_1} & & & & & & \mathbf{C_1} \\ & \mathbf{B_2} & & & & \mathbf{C_2} & \\ & & \mathbf{B_3} & & \mathbf{C_3} & & \\ & & & \mathbf{B_4} & & & \\ & & \mathbf{C_5} & & \mathbf{B_5} & & \\ & \mathbf{C_6} & & & & \mathbf{B_6} & \\ \mathbf{C_7} & & & & & & \mathbf{B_7} \end{bmatrix}$$

Transformation: Iteration Matrix (2)



$$\begin{bmatrix} \mathbf{B_1} & & & & & & \mathbf{C_1} \\ & \mathbf{B_2} & & & & \mathbf{C_2} & \\ & & \mathbf{B_3} & & \mathbf{C_3} & & \\ & & & \mathbf{B_4} & & & \\ & & \mathbf{C_5} & & \mathbf{B_5} & & \\ & \mathbf{C_6} & & & & \mathbf{B_6} & \\ \mathbf{C_7} & & & & & & \mathbf{B_7} \end{bmatrix} \longrightarrow \begin{bmatrix} \mathbf{B_1} & \mathbf{C_1} & & & & & \\ \mathbf{C_7} & \mathbf{B_7} & & & & & \\ & & \mathbf{B_2} & \mathbf{C_2} & & & \\ & & \mathbf{C_6} & \mathbf{B_6} & & & \\ & & & & \mathbf{B_3} & \mathbf{C_3} & \\ & & & & \mathbf{C_5} & \mathbf{B_5} & \\ & & & & & & \mathbf{B_4} \end{bmatrix}$$

$$\|\bar{M}\|_2 = \max\left\{ \max_{j=1,\ldots,n_c} \left\| \begin{bmatrix} B_j & C_j \\ C_k & B_k \end{bmatrix} \right\|_2 , \|B_{N_c}\|_2 \right\},$$

$$k = N_f - j$$

The Story So Far...

- $n_f^2 \times n_f^2$ two-grid iteration matrix M
- Fourier transformation converts 2D problem to a set of n_f problems with 1D structure
- $\|M\|_2$ can be found from norms of N_c smaller problems

n_c of size $2n_f \times 2n_f$, 1 of size $n_f \times n_f$

The Story So Far...

- $n_f^2 \times n_f^2$ two-grid iteration matrix M
- Fourier transformation converts 2D problem to a set of n_f problems with 1D structure
- $\|M\|_2$ can be found from norms of N_c smaller problems

  n_c of size $2n_f \times 2n_f$, 1 of size $n_f \times n_f$

- **IDEA**: analyse semiperiodic version of the problem

  n_c of size $2N_f \times 2N_f$, 1 of size $N_f \times N_f$

- gain insight into Dirichlet problem behaviour?

Semiperiodic problem

- $\boldsymbol{B_j}$, $\boldsymbol{C_j}$ are replaced by periodic versions, e.g.

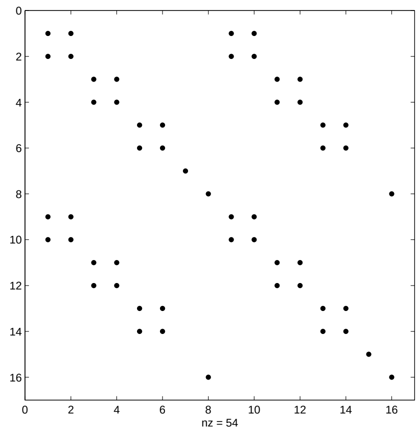
$$\boldsymbol{B_j^{per} = [I - \bar{P}_j^{per} (T_c^{per})_j^{-1} (\bar{P}_j^{per})^T (T_f^{per})_j]\ S_j^{per}}$$

Semiperiodic problem

- B_j, C_j are replaced by periodic versions, e.g.

$$B_j^{per} = [I - \bar{P}_j^{per} (T_c^{per})_j^{-1} (\bar{P}_j^{per})^T (T_f^{per})_j] \; S_j^{per}$$

- transform using coarse grid periodic eigenvectors
- B_j^{per}, C_j^{per} become block diagonal with $\mathbf{2} \times \mathbf{2}$ blocks

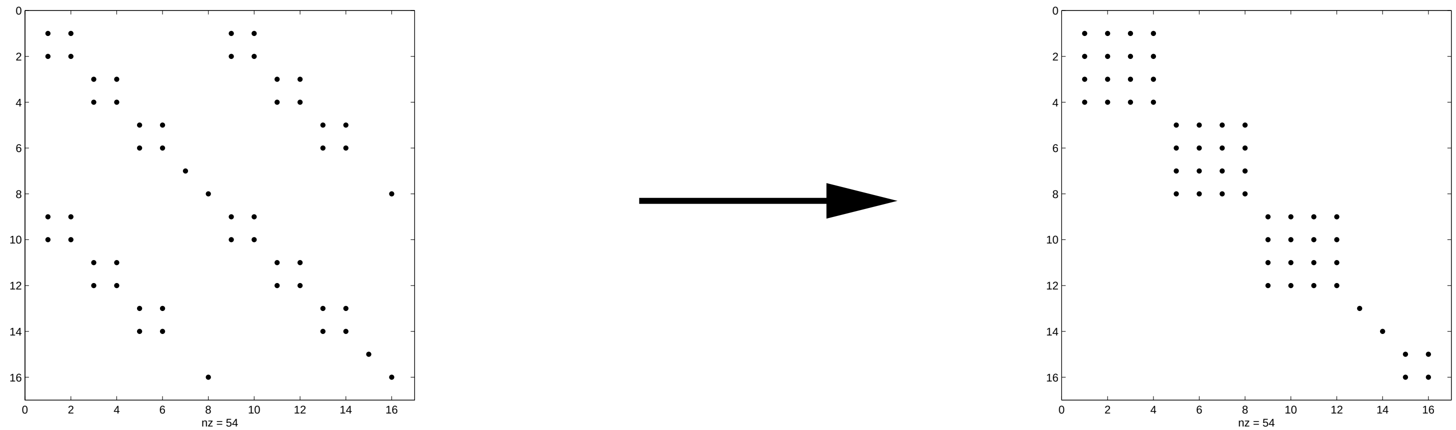


Semiperiodic problem

- B_j, C_j are replaced by periodic versions, e.g.

$$B_j^{per} = [I - \bar{P}_j^{per} (T_c^{per})_j^{-1} (\bar{P}_j^{per})^T (T_f^{per})_j] \; S_j^{per}$$

- transform using coarse grid periodic eigenvectors
- B_j^{per}, C_j^{per} become block diagonal with 2×2 blocks
- permute into block diagonal form



- 2-norm given by maximum 2-norm of the 4×4 blocks

Analytic result

- with semiperiodic approximation, when $P_h > 1$

$$\|M^{per}\|_2 = \frac{\sqrt{3 + \cos(2\pi h)}}{\sqrt{2}(5^\nu)}$$

- as h is small in practice,

$$\|M^{per}\|_2 \simeq \frac{\sqrt{2}}{5^\nu}$$

Analytic result

- with semiperiodic approximation, when $P_h > 1$

$$\|M^{per}\|_2 = \frac{\sqrt{3 + \cos(2\pi h)}}{\sqrt{2}(5^\nu)}$$

- as h is small in practice,

$$\|M^{per}\|_2 \simeq \frac{\sqrt{2}}{5^\nu}$$

- when $P_h < 1$, analysis is more detailed but good approximations to $\|M^{per}\|_2$ can be derived

Analytic result

- with semiperiodic approximation, when $P_h > 1$

$$\|M^{per}\|_2 = \frac{\sqrt{3 + \cos(2\pi h)}}{\sqrt{2}(5^\nu)}$$

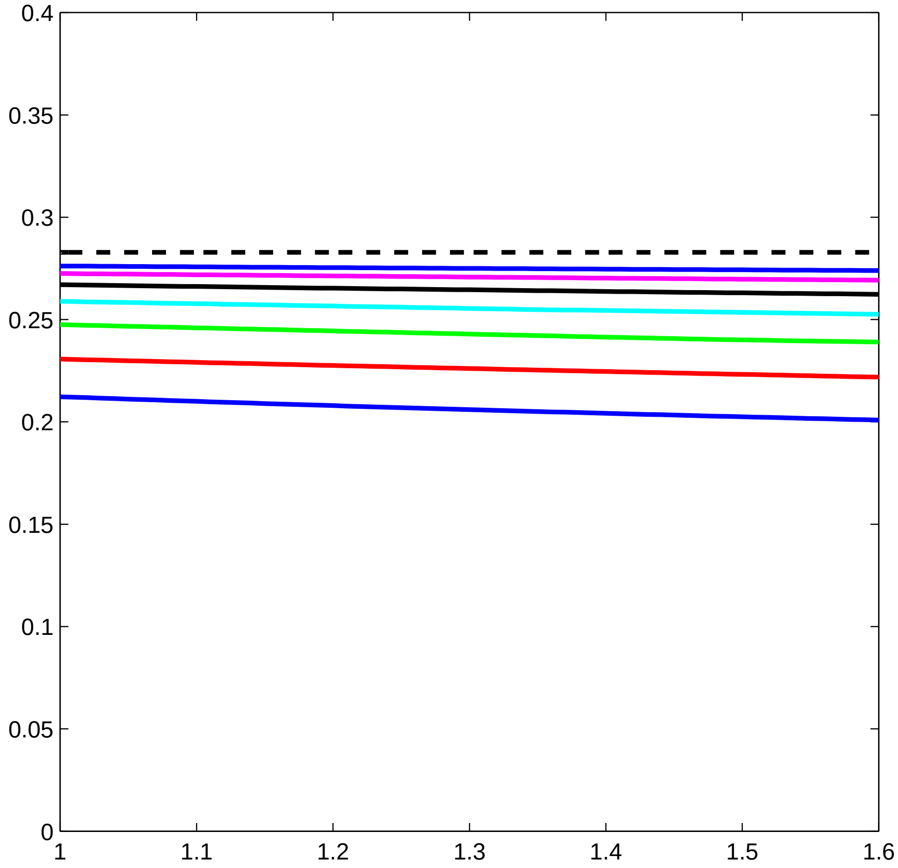
- as h is small in practice,

$$\|M^{per}\|_2 \simeq \frac{\sqrt{2}}{5^\nu}$$

- when $P_h < 1$, analysis is more detailed but good approximations to $\|M^{per}\|_2$ can be derived

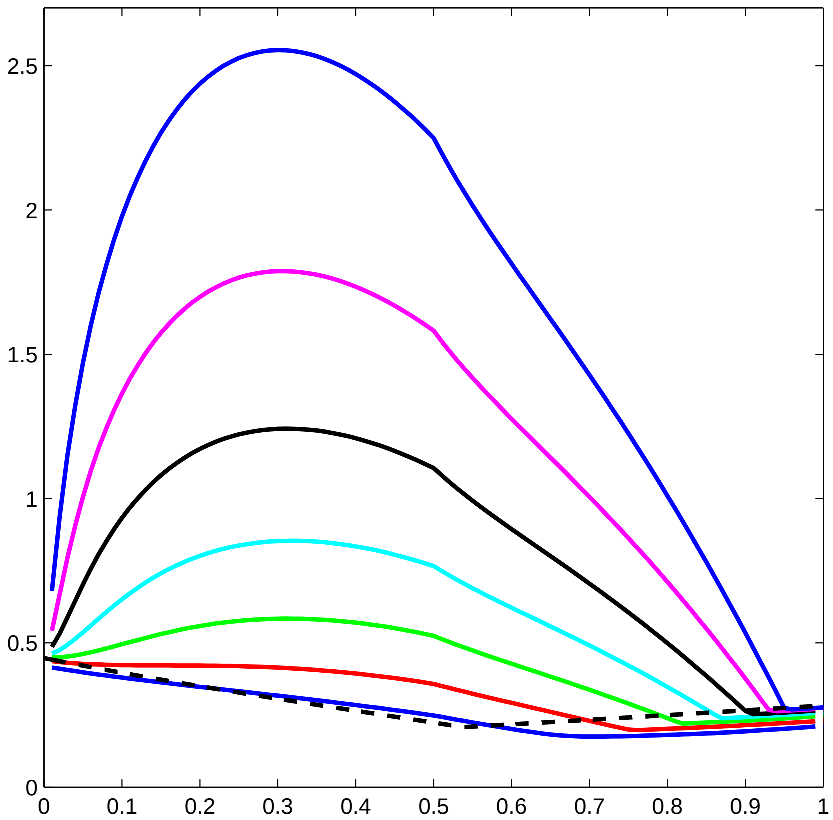
Question: Does this **semiperiodic** analysis correctly predict **Dirichlet** problem behaviour?

Model Problem Results (1)



- $\|M\|_2$ vs P_h
- $P_h \geq 1$ **only**
- semiperiodic: dashed line
- Dirichlet: solid lines
- h fixed for each line
- $h = \frac{1}{8}$ to $h = \frac{1}{512}$
- $\nu = 1$
- semiperiodic: $\frac{\sqrt{2}}{5} \simeq 0.28$
- Dirichlet $\longrightarrow \frac{\sqrt{2}}{5}$

Model Problem Results (2)



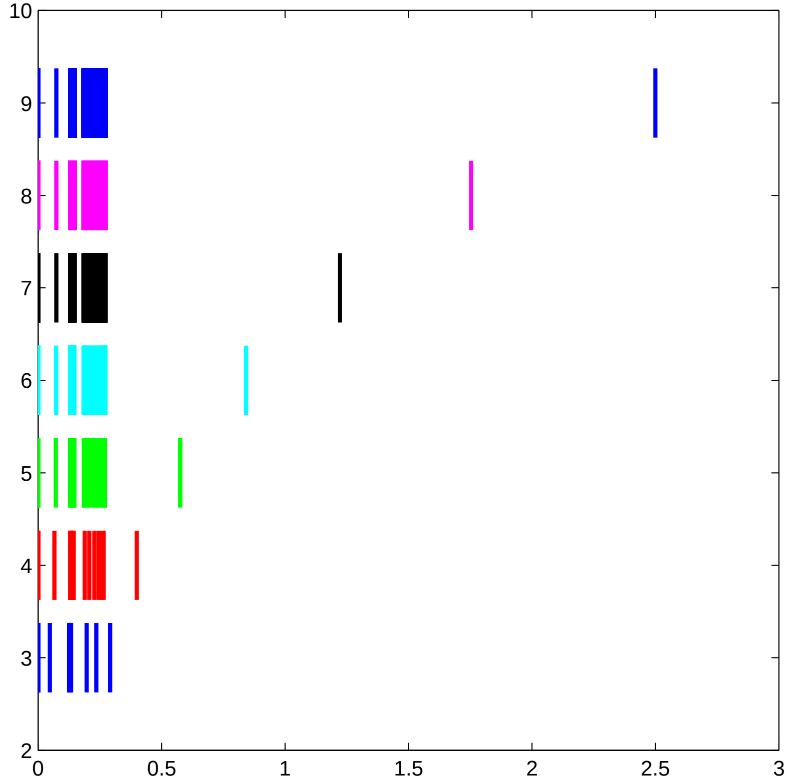
- $\|M\|_2$ vs P_h
- $P_h < 1$ **only**
- semiperiodic: dashed line
- Dirichlet: solid lines
- h fixed for each line
- $h = \frac{1}{8}$ to $h = \frac{1}{512}$
- $\nu = 1$
- not a good match
- MG may diverge!

Observations

- $\|M\|_2 = \sqrt{|\lambda_1(M^*M)|}$

Observations

- $\|M\|_2 = \sqrt{|\lambda_1(M^*M)|}$
- for $P_h < 1$, matrix blocks have one 'bad' eigenvalue



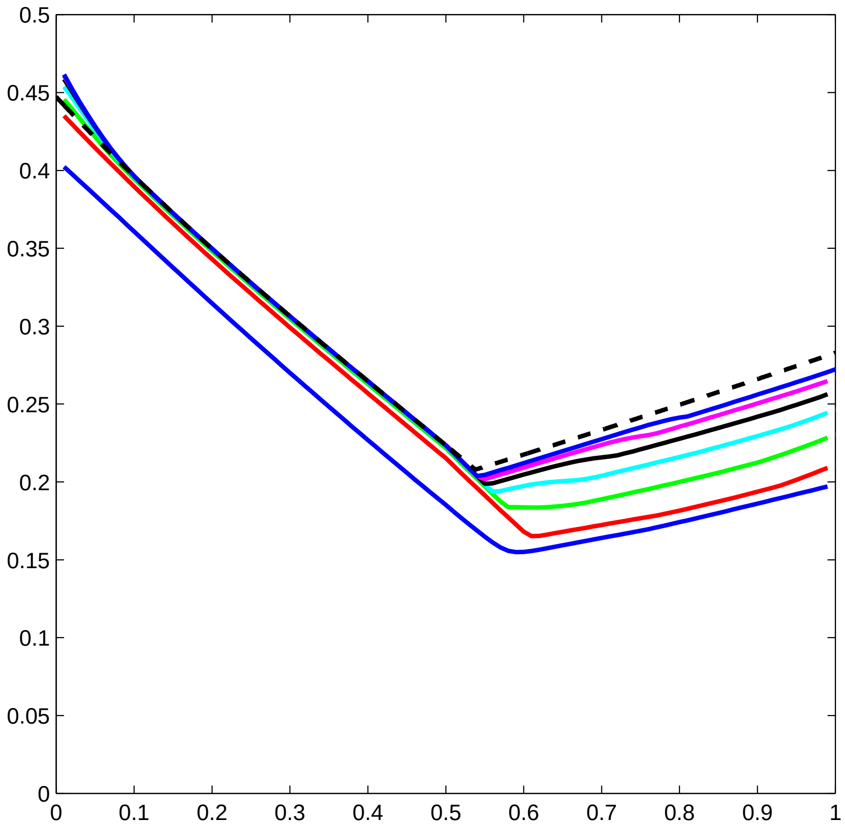
$\sqrt{|\lambda(\mathcal{M}_1^*\mathcal{M}_1)|}$ for fixed $P_h = 0.38$

Alternative Bound?

- artificially ‘remove’ this eigenvalue: use $\sqrt{|\boldsymbol{\lambda_2}(\boldsymbol{M^*M})|}$

Alternative Bound?

- artificially ‘remove’ this eigenvalue: use $\sqrt{|\boldsymbol{\lambda_2(M^*M)}|}$



- $P_h < 1$ **only**
- semiperiodic: $\|\boldsymbol{M^{per}}\|_{\mathbf{2}}$
- Dirichlet: $\sqrt{|\boldsymbol{\lambda_2(M^*M)}|}$

Outlying eigenvalue

- in practice, the effect of this outlying eigenvalue is **transient**
- the eigenvector corresponding to the outlying eigenvalue is large only on grid lines very close to the inflow boundary
- after a few MG iterations, it is **smooth** and so is easily eliminated by coarse grid correction

Outlying eigenvalue

- in practice, the effect of this outlying eigenvalue is **transient**
- the eigenvector corresponding to the outlying eigenvalue is large only on grid lines very close to the inflow boundary
- after a few MG iterations, it is **smooth** and so is easily eliminated by coarse grid correction
- these effects do not have an impact on **practical** MG performance

MG Iteration Counts

- MG-like convergence for any value of P_h

	ϵ										
h	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{8}$	$\frac{1}{16}$	$\frac{1}{32}$	$\frac{1}{64}$	$\frac{1}{128}$	$\frac{1}{256}$	$\frac{1}{512}$	$\frac{1}{1024}$	$\frac{1}{2048}$
$\frac{1}{4}$	5	5	5	5	5	4	4	3	2	2	2
$\frac{1}{8}$	7	7	6	6	5	5	4	4	3	2	2
$\frac{1}{16}$	7	7	7	6	5	5	5	4	4	3	2
$\frac{1}{32}$	7	7	7	7	6	5	5	4	4	3	3
$\frac{1}{64}$	7	7	7	7	6	5	5	4	4	4	3
$\frac{1}{128}$	7	6	6	6	6	6	5	4	4	4	3

$P_h < 1$ $\quad\quad$ $P_h \geq 1$

Further Remarks

- Separate approximation and smoothing matrices:
  - semiperiodic analysis for smoothing matrix norm is representative of Dirichlet problem behaviour for all values of P_h,
  - semiperiodic analysis for approximation matrix norm is representative of Dirichlet problem behaviour for $P_h \geq 1$: for $P_h < 1$, 'bad' eigenvalues again cause trouble.

Further Remarks

- Separate approximation and smoothing matrices:
  - semiperiodic analysis for smoothing matrix norm is representative of Dirichlet problem behaviour for all values of P_h,
  - semiperiodic analysis for approximation matrix norm is representative of Dirichlet problem behaviour for $P_h \geq 1$: for $P_h < 1$, 'bad' eigenvalues again cause trouble.

- Replacing the Dirichlet condition by a Neumann condition on the outflow boundary leads to similar computational results.

Conclusions

- Linear algebra can be used to give a useful insight into convergence of two-grid iteration.
- We have obtained bounds on the multigrid convergence factor for a problem with semiperiodic boundary conditions.
- Boundary effects associated with a Dirichlet condition on the inflow boundary appear to be transient.
- Semiperiodic analysis gives an accurate description of MG behaviour for the full Dirichlet problem.