Fast Solvers for Models of Fluid Flow with Spectral Elements

P. Aaron Lott[1]

University of Maryland, College Park

August 26, 2008
Ph.D. Final Oral Exam

[1]Applied Mathematics & Scientific Computation, palott@ipst.umd.edu

- Motivation & Scientific Context
- Spectral Element Discretization
- Solvers for Convection-Diffusion and Navier-Stokes Equations
- Summary & Future Directions

General Motivation: Develop efficient computational tools to help

- Predict flow behavior
- Understand flow instabilities

General Motivation: Develop efficient computational tools to help

- Predict flow behavior
- Understand flow instabilities

Challenges

- inertial and viscous forces occur on disparate scales
- discrete systems are non-symmetric & poorly conditioned

Conventional Methods use Operator Splitting

- Based on Fast Poisson Solvers LU, SOR, FDM, Cyclic Reduction, Multigrid
- Limited by CFL condition $\Delta t \leq C \frac{\Delta x}{U}$

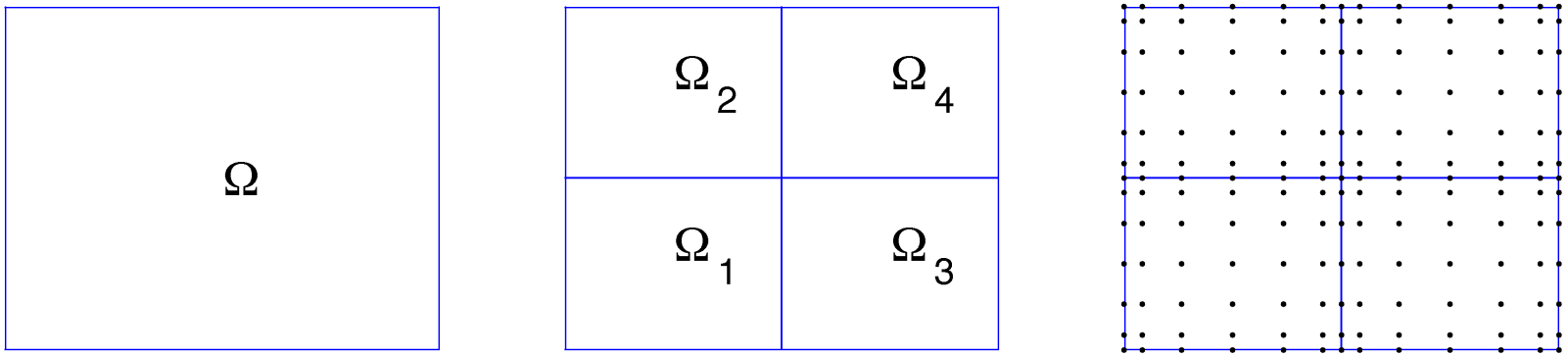
Modern Methods: Newton-Krylov-Schwarz

- Nonlinear solver (Picard/Newton) used to solve nonlinear system
- Krylov subspace method used to solve subsidiary linear system
- Domain Decomposition scheme used to precondition linear system
- Require Fast Convection-Diffusion Solvers

Plan:

- use spectral element method for an accurate discretization
- develop fast solvers that take advantage of the structure of the discrete system

Spectral Element Discretization



Variables on each element are expressed via a nodal basis

$$u_N^e(x,y) = \sum_{i=0}^{N}\sum_{j=0}^{N} u_{ij}\pi_i^N(x)\pi_j^N(y) \tag{1}$$

Variables on element interfaces are coupled using an averaging operation

$$\Sigma' = \underbrace{Q}_{scatter} \underbrace{W_L}_{weight} \underbrace{Q^T}_{sum} \tag{2}$$

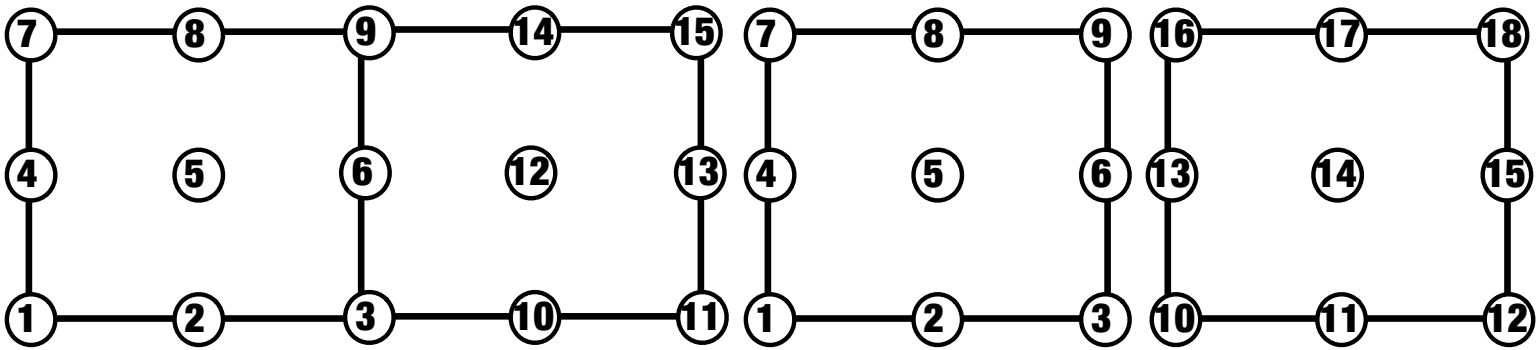


Figure: (Left) Global and (Right) Local ordering of the degrees of freedom.

Convection-Diffusion

$$-\epsilon\nabla^2 u + \vec{w}\cdot\nabla u = f \tag{3}$$

Navier-Stokes

$$\begin{aligned} -\nu\nabla^2\vec{u} + \vec{u}\cdot\nabla\vec{u} + \nabla p &= f \\ -\nabla\cdot\vec{u} &= 0 \end{aligned} \tag{4}$$

Break problem into 3 cases

- Convection-Diffusion systems with constant wind field on each element $\vec{w} = (w_x, w_y)$
- General Convection-Diffusion systems $\vec{w} = (w_x(x, y), w_y(x, y))$
- Linearized Navier-Stokes systems convection field $\vec{u}$ obtained from nonlinear iteration

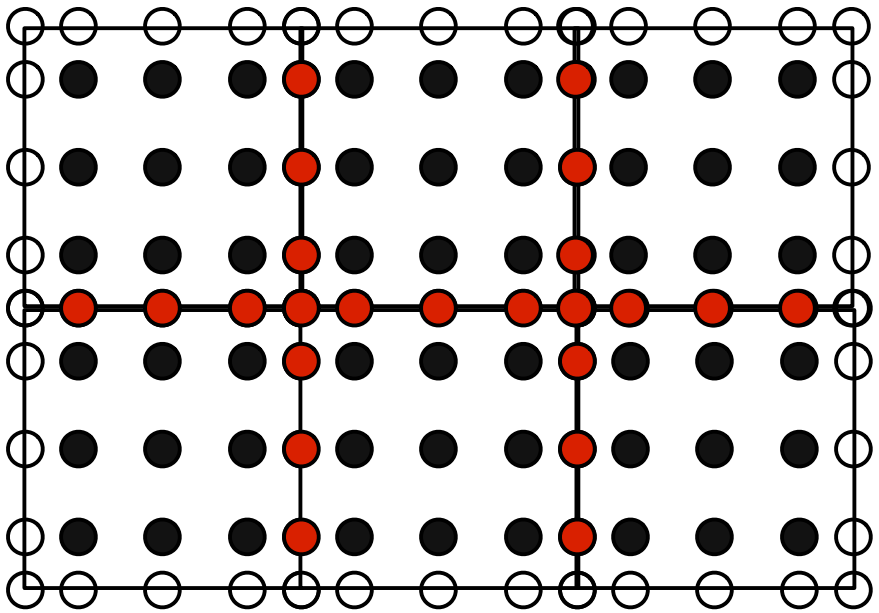
Start with the Discrete weak form of the convection-diffusion equation

$$F(w)u = b \tag{5}$$

$$\begin{aligned} F^e(w^e) &= F_x^e + F_y^e \\ F_x^e &= \underbrace{\epsilon(\hat{M} \otimes \frac{h_y}{h_x}\hat{A})}_{\text{Diffusion in x}} + \underbrace{W_x^e(\hat{M} \otimes \frac{h_y}{2}\hat{C})}_{\text{Convection in x}} \\ F_y^e &= \underbrace{\epsilon(\frac{h_x}{h_y}\hat{A} \otimes \hat{M})}_{\text{Diffusion in y}} + \underbrace{W_y^e(\frac{h_x}{2}\hat{C} \otimes \hat{M})}_{\text{Convection in y}}. \end{aligned} \tag{6}$$

Key Idea: for constant winds

$$F^e(w_x, w_y) = \hat{M} \otimes \hat{F}_x + \hat{F}_y \otimes \hat{M} =: \bar{F}^e \tag{7}$$



- use an iterative solver on **elemental interfaces**
- diagonalize 1D operators in **element interiors**

Formally order nodes by interior and boundary nodes

$$\begin{bmatrix} \bar{F}^1_{II} & 0 & \dots & 0 & \bar{F}^1_{I\Gamma} \\ 0 & \bar{F}^2_{II} & 0 & \dots & \bar{F}^2_{I\Gamma} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{F}^E_{II} & \bar{F}^E_{I\Gamma} \\ \bar{F}^1_{\Gamma I} & \bar{F}^2_{\Gamma I} & \dots & \bar{F}^E_{I\Gamma} & \bar{F}_{\Gamma\Gamma} \end{bmatrix} \begin{pmatrix} u_{I^1} \\ u_{I^2} \\ \vdots \\ u_{I^E} \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} \hat{b}_{I^1} \\ \hat{b}_{I^2} \\ \vdots \\ \hat{b}_{I^E} \\ \hat{b}_\Gamma \end{pmatrix}.$$

LU decomposition of the system matrix gives us

$$\begin{bmatrix} I & 0 & \dots & 0 & 0 \\ 0 & I & 0 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & I & 0 \\ \bar{F}^1_{\Gamma I}\,\bar{F}^{1\,-1}_{II} & \bar{F}^2_{\Gamma I}\,\bar{F}^{2\,-1}_{II} & \dots & \bar{F}^E_{\Gamma I}\,\bar{F}^{E\,-1}_{II} & I \end{bmatrix} \begin{bmatrix} \bar{F}^1_{II} & 0 & \dots & 0 & \bar{F}^1_{I\Gamma} \\ 0 & \bar{F}^2_{II} & 0 & \dots & \bar{F}^2_{I\Gamma} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{F}^E_{II} & \bar{F}^E_{I\Gamma} \\ 0 & 0 & \dots & 0 & \bar{F}_S \end{bmatrix}$$

$$\begin{bmatrix} \bar{F}_{II}^1 & 0 & \dots & 0 & \bar{F}_{I\Gamma}^1 \\ 0 & \bar{F}_{II}^2 & 0 & \dots & \bar{F}_{I\Gamma}^2 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \bar{F}_{II}^E & \bar{F}_{I\Gamma}^E \\ 0 & 0 & \dots & 0 & \bar{F}_S \end{bmatrix} \begin{pmatrix} u_{I1} \\ u_{I2} \\ \vdots \\ u_{IE} \\ u_\Gamma \end{pmatrix} = \begin{pmatrix} \hat{b}_{I1} \\ \hat{b}_{I2} \\ \vdots \\ \hat{b}_{IE} \\ g_\Gamma \end{pmatrix}$$

$\bar{F}_S = \sum_{e=1}^{E} (\bar{F}_{\Gamma\Gamma}^e - \bar{F}_{\Gamma I}^e \, \bar{F}_{II}^{e\,-1} \, \bar{F}_{I\Gamma}^e)$ represents the Schur complement of the system.

Almost a direct solver via a 3 step procedure:

- Compute g_Γ (Lower Triangular)
- Solve $\bar{F}_S u_\Gamma = g_\Gamma$ via preconditioned GMRES
- Back solve for elemental interiors u_I^e, using $F_{II}^{e\,-1}$ (Upper Triangular)

$$F^e(w_x, w_y) = \hat{M} \otimes \hat{F}_x + \hat{F}_y \otimes \hat{M} =: \bar{F}^e \tag{8}$$

Write $\bar{F}^e = \tilde{M}\tilde{F}^e\tilde{M}$ and diagonalizae $\tilde{F}^e$.

$$\begin{aligned} \tilde{F}^e &= \tilde{M}^{-1/2}\bar{F}^e\tilde{M}^{-1/2} \\ &= (\hat{M}^{-1/2} \otimes \hat{M}^{-1/2})(\hat{M} \otimes \hat{F}_x + \hat{F}_y \otimes \hat{M})(\hat{M}^{-1/2} \otimes \hat{M}^{-1/2}) \\ &= (I \otimes \hat{M}^{-1/2}\hat{F}_x\hat{M}^{-1/2}) + (\hat{M}^{-1/2}\hat{F}_y\hat{M}^{-1/2} \otimes I) \\ &= (I \otimes B) + (A \otimes I). \end{aligned} \tag{9}$$

$$\bar{F}^{e\,-1} = \tilde{M}^{-1}(V_y \otimes V_x)(\Lambda_y \otimes I + I \otimes \Lambda_x)^{-1}(V_y^{-1} \otimes V_x^{-1})\tilde{M}^{-1}$$

- All based on Tensor Products of 1D operators
- Extends to 3D problems

The interface problem can be solved via GMRES, and Matrix-Vector products can be applied element-wise

$$\bar{F}_S u_\Gamma = g_\Gamma$$

$$\underbrace{\sum_{e=1}^{E} (\bar{F}^e_{\Gamma\Gamma} - \bar{F}^e_{\Gamma I}\, {\bar{F}^e_{II}}^{-1}\, \bar{F}^e_{I\Gamma})}_{\bar{F}_S} \underbrace{u_\Gamma}_{u_\Gamma} = \underbrace{\sum_{e=1}^{E} (\hat{b}_{\Gamma^e} - \bar{F}^e_{\Gamma I}\, {\bar{F}^e_{II}}^{-1}\, \hat{b}_{I^e})}_{g_\Gamma}.$$

We use a preconditioner of the form

$$\sum_{e=1}^{E} D^{(e)} R_e^T \left(\bar{F}^e_S\right)^{-1} R_e D^{(e)}, \tag{10}$$

which can be applied as

$${\bar{F}^{(e)}_S}^{-1} v = \begin{pmatrix} 0 & I \end{pmatrix} {\bar{F}^{(e)}}^{-1} \begin{pmatrix} 0 \\ I \end{pmatrix} v. \tag{11}$$

Grid Aligned Flow with Analytical Solution

$$u(x,y) = x\left(\frac{1-e^{(y-1)/\epsilon}}{1-e^{-2/\epsilon}}\right). \tag{12}$$

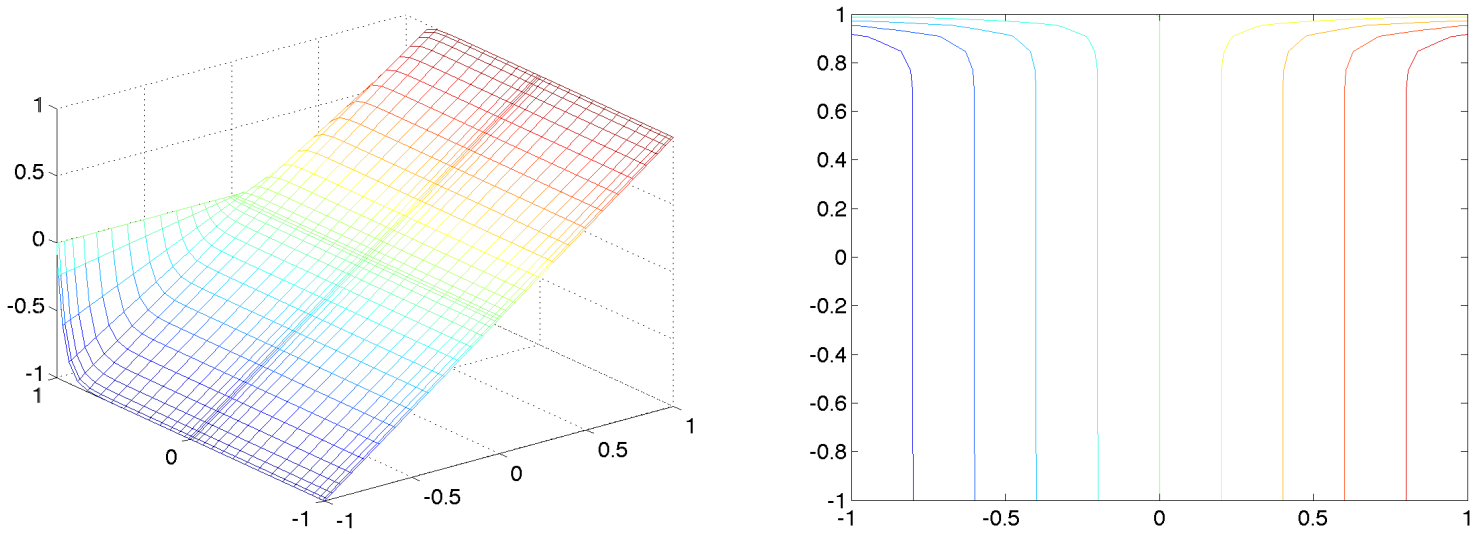


Figure: Computed solution and contours of steady convection diffusion flow with constant wind $\vec{w} = (0,1)$ and moderate convection $Pe = 40$.

N	$\|u - u_N\|_2$	None	N-N	R-R	$\frac{1}{h}(1 + log(N))^2$
4	5.535×10^{-2}	3	3	3	5.6
8	2.505×10^{-3}	7	7	7	9.4
16	2.423×10^{-7}	15	11	14	14.2
32	7.931×10^{-13}	30	16	18	19.9

E	$\|u - u_N\|_2$	$13h^3$	None	N-N	R-R	$\frac{1}{h}(1 + log(N))^2$
16	8.594×10^{-2}	2.031×10^{-1}	13	13	12	11.5
64	2.593×10^{-2}	2.523×10^{-2}	49	47	25	22.9
256	3.558×10^{-3}	3.174×10^{-3}	108	88	45	45.9
1024	3.610×10^{-4}	3.967×10^{-4}	312	180	85	91.7

Table: Pe=40, polynomial degree is varied with a fixed 2×2 element grid (top) and the number of quadratic (N=2) elements are varied (bottom).

Pe	None	N-N	R-R
125	85	92	35
250	79	98	30
500	81	119	25
1000	103	164	27
2000	160	> 200	38
5000	> 200	> 200	74

Table: Comparison of iteration counts for example 1 with increasingly convection-dominated flows. N=8, E=256 using 16×16 element grid.

Oblique flow with internal and outflow boundary layers

$w = (-sin(\pi/6), cos(\pi/6))$

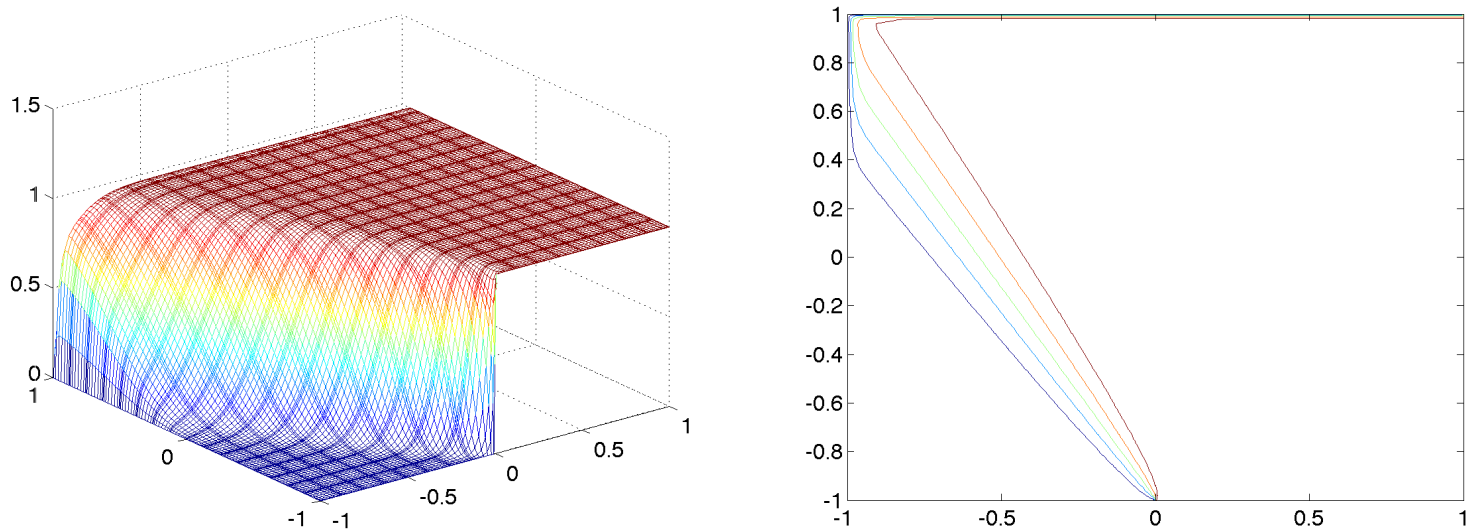


Figure: Velocity (left) and contours (right) of a convection-dominated steady convection-diffusion flow, $Pe = 250$, corresponding to example 2.

N	None	N-N	R-R
4	13	13	13
8	25	25	18
16	36	28	20
32	50	29	21

E	None	N-N	R-R
16	29	33	21
64	40	63	26
256	69	117	46
1024	132	> 200	87

Table: Pe=40, fixed 2×2 element grid (top), quadratic (N=2) (bottom).

Pe	None	N-N	R-R
125	93	104	38
250	75	98	32
500	64	115	27
1000	69	150	30
2000	99	> 200	41
5000	> 200	> 200	94

Table: N=8, E=256 using 16×16 element grid.

Recap for constant wind problems

- GMRES with RR converges in roughly $\frac{C}{h}(1 + log(N))^2$, where C grows with Pe
- algorithm has mild dependence on Pe (C ranged from $\sim .5$ to ~ 1.25)

Next: apply this technology to non-constant wind problems.

Double Glazing problem - recirculating wind

$$\vec{w} = (y(1-x^2), -x(1-y^2))$$

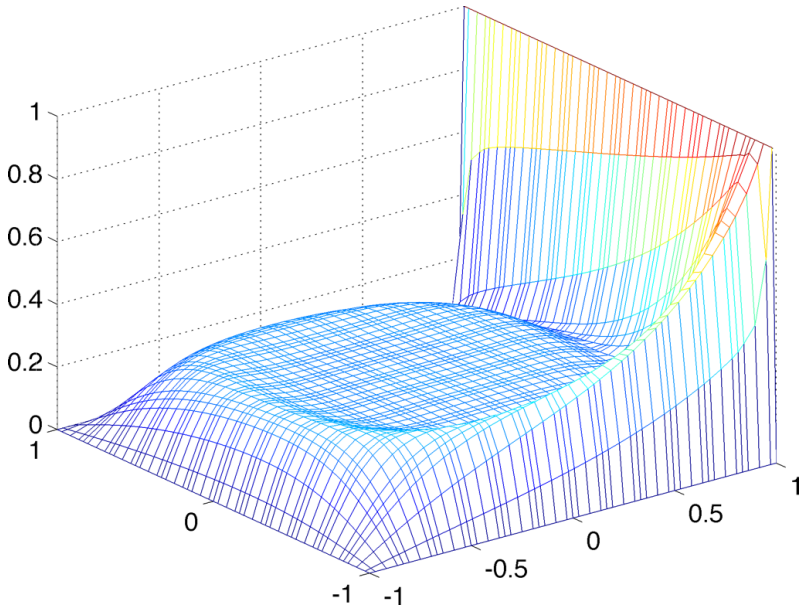


Figure: Solution corresponding to Peclet number= 400 N=4, E=12x12

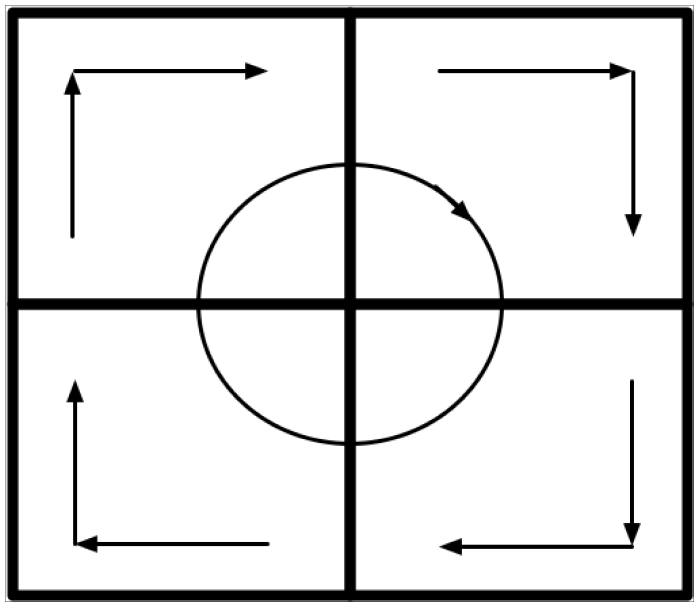


Figure: Depiction of a constant wind approximation

Non-constant wind solution algorithm

When the wind is non-constant, we use $\bar{F}$ as Preconditioner to accelerate convergence of GMRES.

$$F(\vec{w})\bar{F}(\bar{w})^{-1}\bar{F}(\bar{w})u = Mf$$

- FGMRES (outer iteration F)
- Domain Decomposition Preconditioner $\bar{F}$
  - Interior Subdomain Solver - FDM
  - Interface Solver - GMRES (inner iteration $\bar{F}_S$)
    - Inexact Solve
    - Robin-Robin Preconditioner

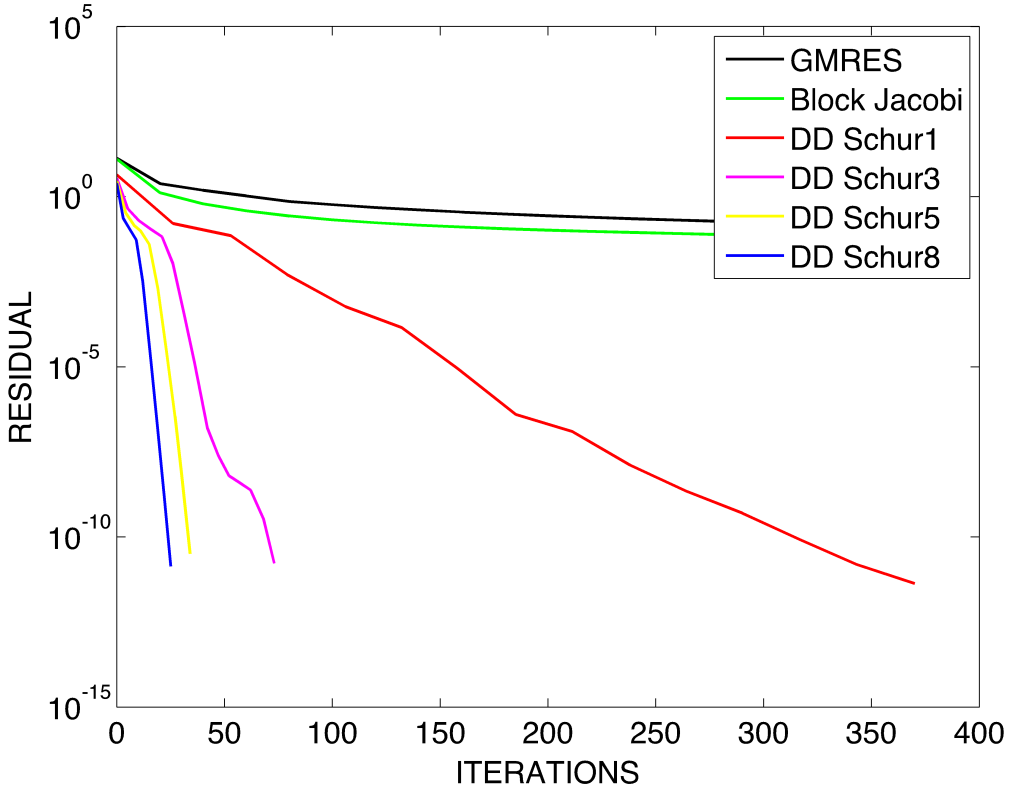


Figure: Convergence comparison of outer GMRES iterations needed depending on number of interface solve steps taken. Peclet number= 400 P=4, E=12x12

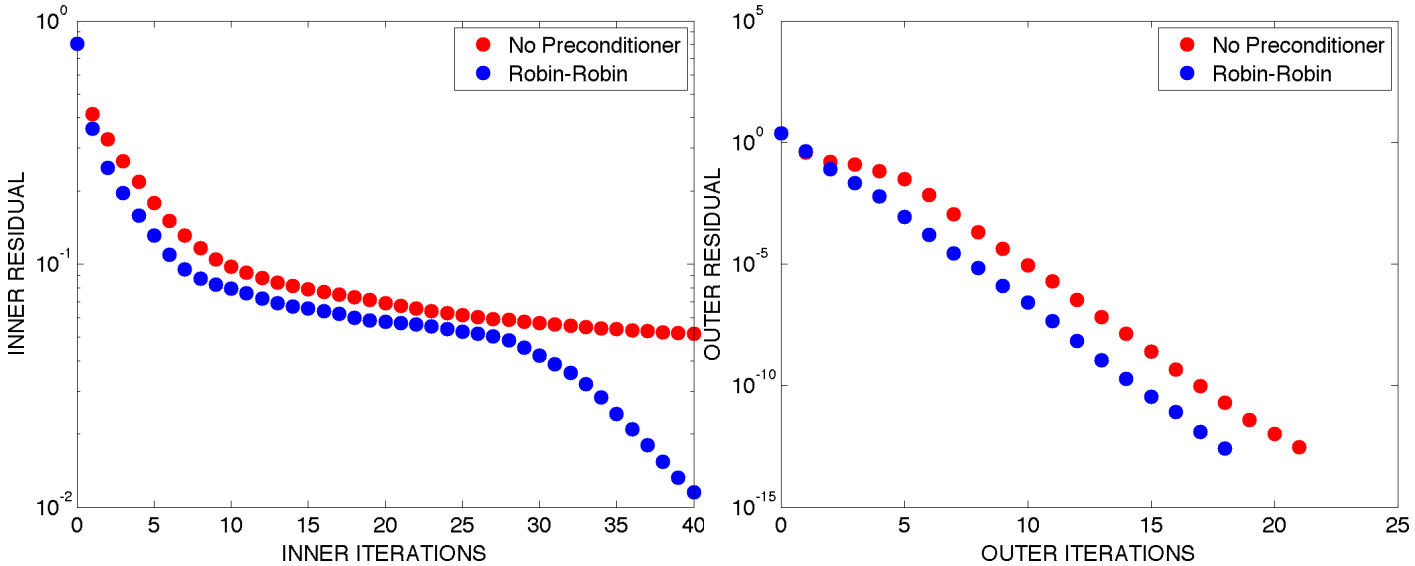


Figure: Comparison of residuals for interface iterations obtained by GMRES without preconditioning and with Robin-Robin preconditioning (left). Affect on FGMRES residuals with inexact $\bar{F}^{-1}$ using no interface preconditioner and Robin Robin (right).

N	Number of FGMRES Outer Iterations	Number of Inner Iterations
4	40	5
8	51	5
16	44	13
32	48	20

E	Number of FGMRES Outer Iterations	Number of Inner Iterations
16	40	5
64	25	12
256	17	19
1024	28	20

Table: $Pe = 400$ fixed 4×4 element grid (top), $N = 4$(bottom).

Pe	Number of FGMRES Outer Iterations	Number of Inner Iterations
125	14	20
250	16	16
500	19	18
1000	24	18
2000	34	16
5000	67	13

Table: Peclet number is increased on a fixed grid $N = 8$ and $E = 256$.

Recap for convection-diffusion problems

- for constant wind problems use Domain Decomposition Solver
- for non-constant wind problems use (inexact) Domain Decomposition as Preconditioner for FGMRES

Next: apply this technology to steady Navier-Stokes problems.

Steady Navier-Stokes equations

$$-\nu\nabla^2\vec{u} + \vec{u}\cdot\nabla\vec{u} + \nabla p = f \tag{13}$$
$$-\nabla\cdot\vec{u} = 0$$

Discrete weak form:

$$\begin{bmatrix} F(u) & -D^T \\ -D & 0 \end{bmatrix} \begin{pmatrix} u \\ p \end{pmatrix} = \begin{pmatrix} b \\ 0 \end{pmatrix}. \tag{14}$$

Steady Navier-Stokes Solution Algorithm Overview

- Nonlinear iteration (Picard)
- Linear iteration (Block FGMRES)
- Block LSC Preconditioner

$$P = \begin{bmatrix} \bar{F} & -D^T \\ 0 & -P_S \end{bmatrix}. \tag{15}$$

  - Domain Decomposition and FDM for $\bar{F}$ Block
  - Conjugate Gradient Method for P_S block

Nonlinear system solved via Picard Iteration $x_{k+1} = x_k + \delta x_k$ with updates δx_k obtained by solving

$$\begin{bmatrix} F(u_k) & -D^T \\ -D & 0 \end{bmatrix} \underbrace{\begin{pmatrix} \delta u_k \\ \delta p_k \end{pmatrix}}_{\delta x_k} = \begin{pmatrix} f - (F(u_k) + D^T p_k) \\ D^T u_k \end{pmatrix}.$$

Our block preconditioner is based on the upper block of LU factorization

$$\begin{bmatrix} F & -D^T \\ -D & 0 \end{bmatrix} = \begin{bmatrix} I & 0 \\ -DF^{-1} & I \end{bmatrix} \begin{bmatrix} F & -D^T \\ 0 & -S \end{bmatrix}. \tag{16}$$

Least Squares Commutator:
if $\epsilon_h = (M^{-1}F)(M^{-1}D^T) - (M^{-1}D^T)(M_p^{-1}F_p)$ is small, then

$$DF^{-1}D^T \approx DM^{-1}D^T F_p^{-1} M_p \tag{17}$$

F_p is constructed to make ϵ_h small by minimizing an L_2 norm of ϵ_h via least-squares i.e.

$$min \| [M^{-1}FM^{-1}D^T]_j - [M^{-1}D^T M_p^{-1}[F_p]_j] \|_M \tag{18}$$

$$P_s = (DM^{-1}D^T)(DM^{-1}FM^{-1}D^T)^{-1}(DM^{-1}D^T) \tag{19}$$

Lid-Driven Cavity

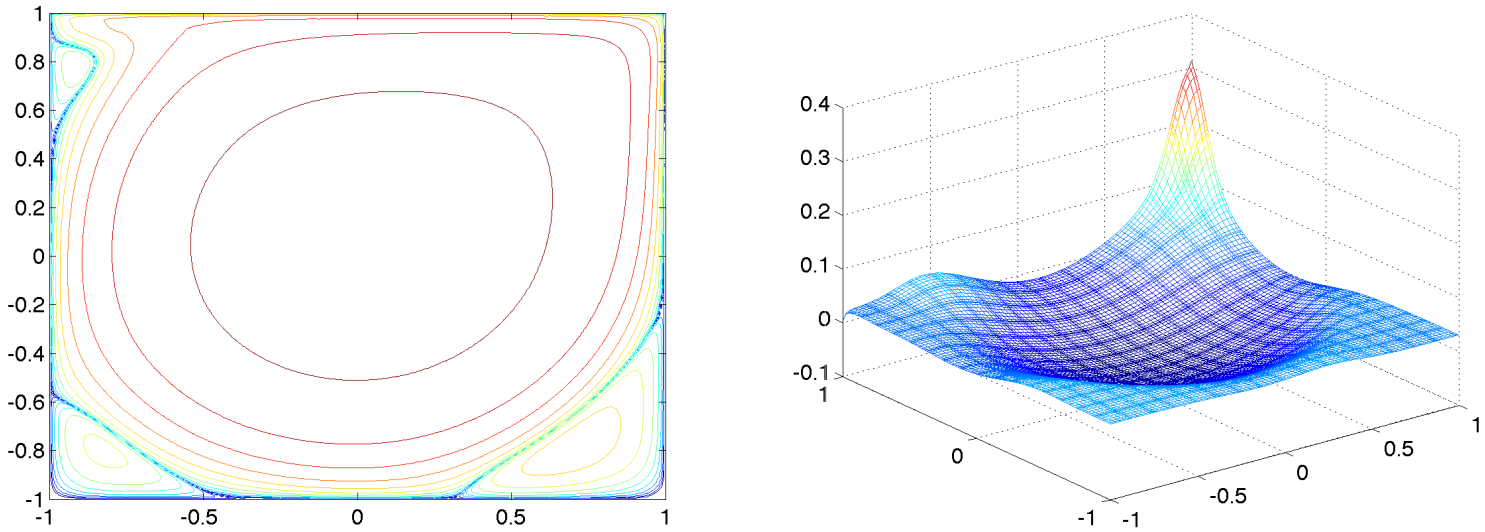


Figure: Streamline plot (left) and pressure plot (right) of Lid-Driven Cavity with $Re = 2000$.

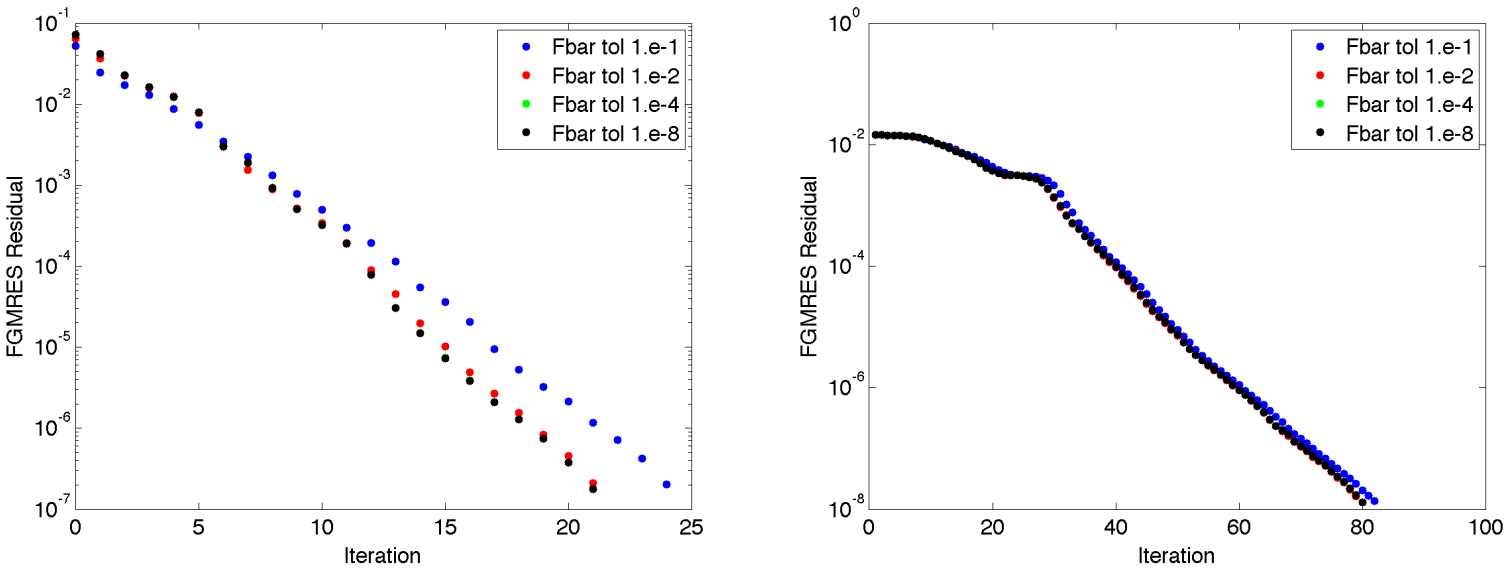


Figure: Comparison of linear residuals for $Re = 100$ (left) and $Re = 1000$ (right) using FGMRES based on solving $\bar{F}_S^{-1}$ to a tolerance of 10^{-1} (blue), 10^{-2} (red), 10^{-4} (beneath black) and 10^{-8} (black).

N	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
2	16	20	14	8
4	9	32	16	60
8	8	38	24	139
16	8	54	33	200

E	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
16	16	20	14	8
64	12	26	32	22
256	9	37	94	45
1024	8	55	200	87

Table: $Re = 100$. N is varied as $E = 16$ on a fixed 4×4 element grid (top). E is varied as $N = 4$ is fixed (bottom).

Re	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
10	5	30	186	200
100	7	42	141	200
500	9	57	130	200
1000	11	96	119	200
2000	15	194	145	200
5000	20	240	107	200

Table: Re is increased on a fixed grid $N = 8$ and $E = 256$.

Kovasznay Flow - Analytical Solution

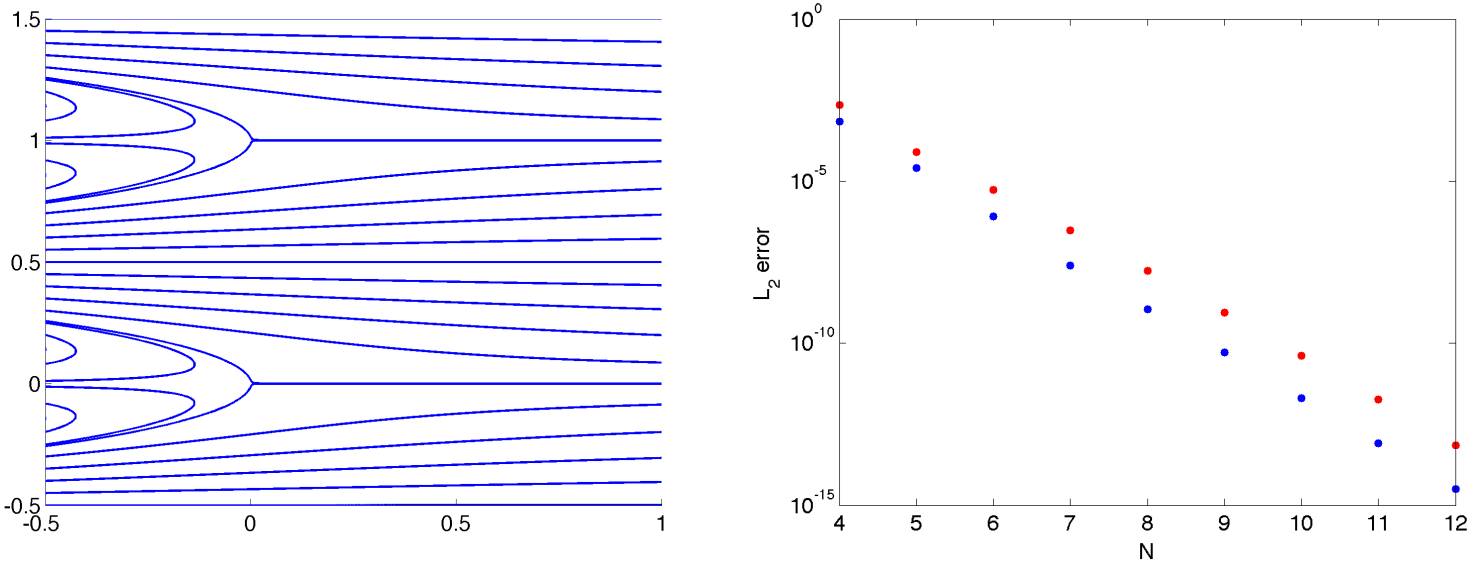


Figure: $Re = 40$ Streamline plot of Kovasznay Flow (left). Spectral convergence on fixed 4×6 element grid (right).

N	$\frac{\|u_x - u_x^N\|_2}{\|u_x\|_2}$	$\frac{\|u_y - u_y^N\|_2}{\|u_y\|_2}$	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
4	6.8×10^{-4}	2.2×10^{-3}	12	42	4	27
5	2.6×10^{-5}	8.0×10^{-5}	12	39	4	35
6	3.6×10^{-6}	2.0×10^{-5}	15	35	3	22
7	2.5×10^{-8}	3.1×10^{-7}	18	32	3	25
8	1.1×10^{-9}	1.8×10^{-8}	23	27	4	28
9	5.1×10^{-11}	8.8×10^{-10}	27	26	5	45
10	2.0×10^{-12}	4.2×10^{-11}	29	24	6	74
11	8.1×10^{-14}	1.8×10^{-12}	30	24	5	58
12	3.1×10^{-15}	7.0×10^{-14}	36	24	10	72

Table: Re=40, on a fixed 4×6 element grid.

Flow over a step

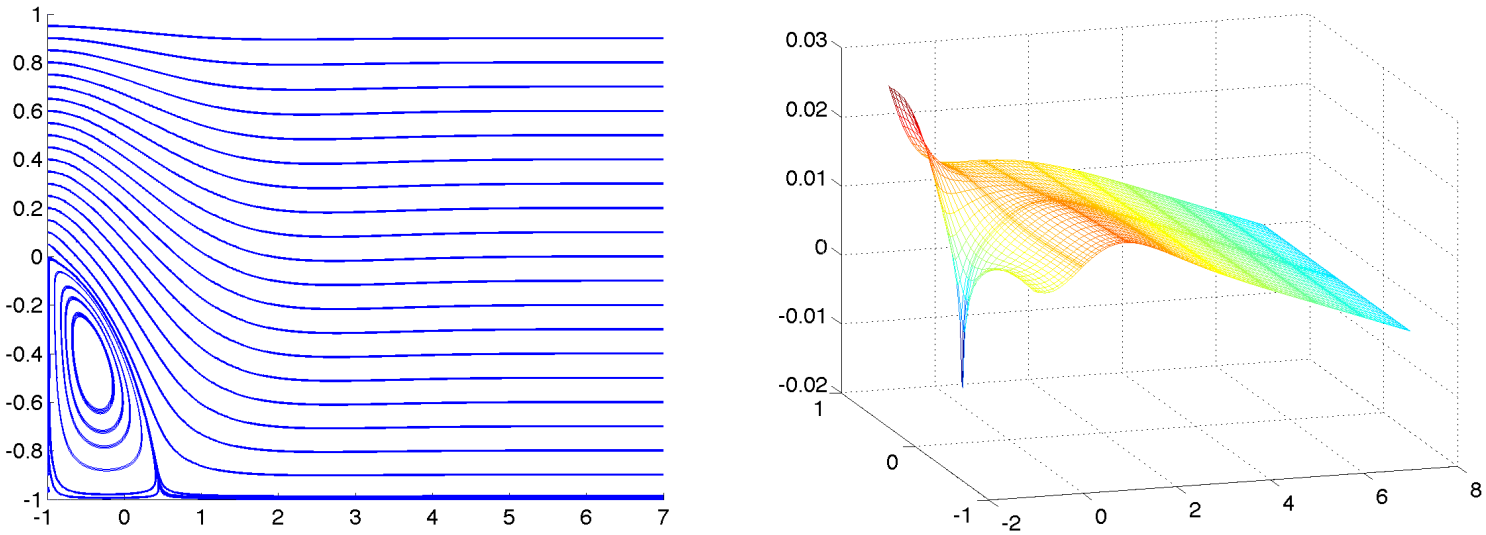


Figure: Streamline plot (left) and pressure plot (right) of flow over a step with $Re = 200$.

Re	N	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
10	4	4	14	22	66
100	4	6	22	16	27
200	4	7	33	14	20
10	8	4	22	34	59
100	8	5	31	25	39
200	8	7	31	23	39
10	16	4	42	42	106
100	16	5	57	38	200
200	16	6	69	34	72

Table: Increasing Reynolds number and N using a fixed 8×2 element grid.

Re	E	Picard steps	FGMRES steps	$\bar{F}_S^{-1}$ steps	P_S^{-1} steps
10	16	4	14	22	66
100	16	6	22	16	27
200	16	7	33	14	20
10	64	4	16	56	47
100	64	5	22	34	24
200	64	7	28	32	28
10	256	4	28	109	124
100	256	5	31	94	48
200	256	6	34	86	38

Table: Increasing Reynolds number and E with $N = 4$ fixed.

Developed Solvers for 3 models of steady fluid flow

- Convection-Diffusion with constant wind
- Convection-Diffusion with non-constant wind
- Steady Navier-Stokes

Solvers are robust

- grid aligned, oblique, rotating, enclosed flows, outflows
- slight dependence on mesh size and Peclet/Reynolds number

Primary Contributions

- extended use of FDM to steady convection-diffusion problems
- extended Least-Squares commutator to Matrix-Free SEM framework

Future Directions

- Use in stability analysis & Implicit time integration methods for flow studies
- Extend to 3D. FDM very competitive against LU $O(n^4)$ vs. $O(n^6)$
- Extend to Parallel Architectures using second level of Σ'

Thank You.

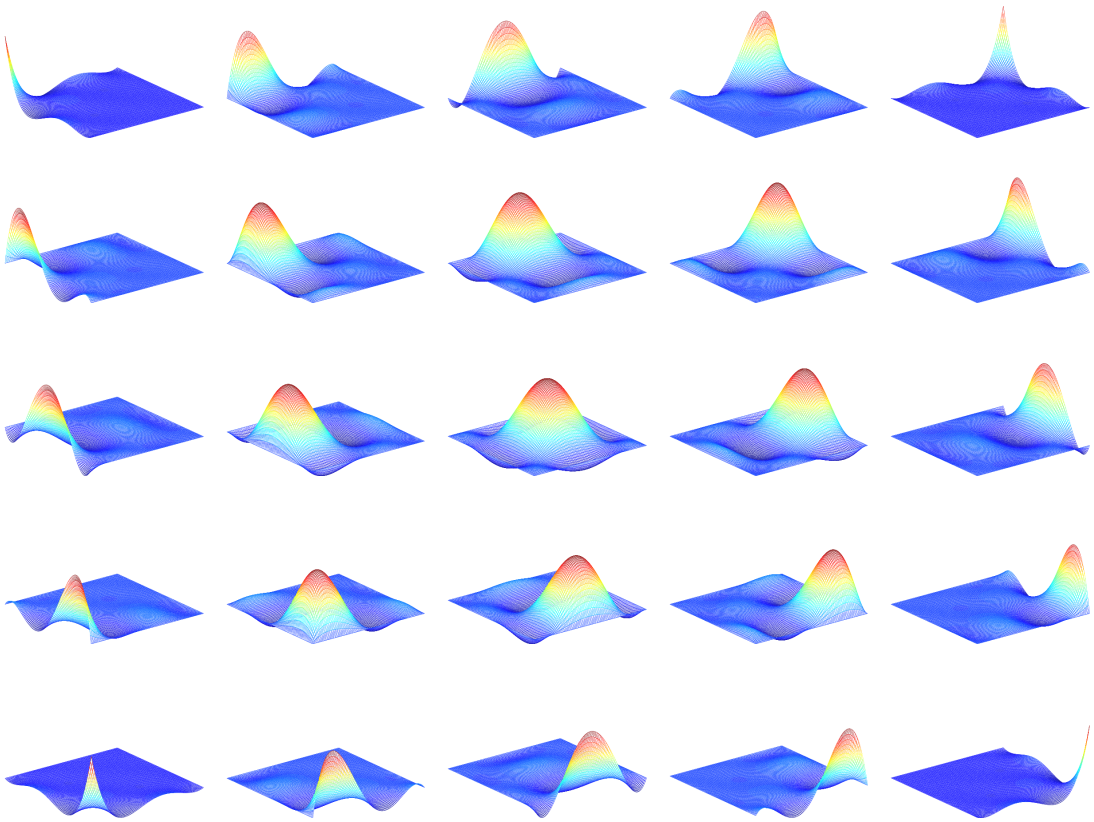


Figure: 4th Order 2D Lagrangian nodal basis functions $\pi_i \otimes \pi_j$ based on the Gauss-Labotto-Legendre points.

Convection-Diffusion Operator

$$\epsilon \int_{\Omega_e} \nabla u \cdot \nabla v + \int_{\Omega^e} (\vec{w} \cdot \nabla u) v \tag{20}$$

$$F^e = [F^e_{ij}], \quad F^e_{ij} = \int_{\Omega_e} \nabla \pi_i \cdot \nabla \pi_j + \int_{\Omega_e} (\vec{w} \cdot \nabla \pi_j) \pi_i \tag{21}$$

Divergence Operator

$$\int_{\Omega^e} q(\nabla \cdot \vec{u}) = \frac{h_y}{2} \int_{-1}^{1} \pi_{N-2,k} \frac{\partial \pi_{N,i}}{\partial x} + \frac{h_x}{2} \int_{-1}^{1} \pi_{N-2,k} \frac{\partial \pi_{N,i}}{\partial y}. \quad (22)$$

Applying Gauss-Legendre quadrature yields the discrete form

$$D_{ij}^e = \sum_{d=1}^{2} \sum_{i,j=1}^{N} \pi_{N-2,i}(\eta_i) \pi_{N-2,j}(\eta_j) \frac{\partial \pi_N}{\partial x_d}(\eta_i, \eta_j) \sigma_i \sigma_j, \quad (23)$$

$$a_e(u,v) = \int_{\Omega_e} (\epsilon \nabla u \cdot \nabla v + (\vec{w} \cdot \nabla u)v), \tag{24}$$

which can be derived from the element-based Neumann problem

$$-\epsilon \nabla^2 u + (\vec{w} \cdot \nabla)u = f \qquad \text{in } \Omega_e, \tag{25}$$

$$-\epsilon \frac{\partial u}{\partial n} = 0 \text{ on } \Gamma_e. \tag{26}$$

$$a_e(u,v) = \int_{\Omega_e} (\epsilon \nabla u \cdot \nabla v + (\vec{w} \cdot \nabla u)v) - \int_{\Gamma_i} \vec{w} \cdot \vec{n} uv, \qquad (27)$$

which corresponds to the element-based Robin problem

$$-\epsilon \nabla^2 u + (\vec{w} \cdot \nabla)u = f \qquad \text{in } \Omega_e, \qquad (28)$$

$$-\epsilon \frac{\partial u}{\partial n} + \vec{w} \cdot \vec{n} u = 0 \text{ on } \Gamma_e. \qquad (29)$$

If the function $u \in H_0^s(\Omega) \times H^{s-1}(\Omega)$ having smoothness s, then $\|u - u_N\| \leq Ch^{\min(N,s)} N^{-s} \|u\|_{H_0^s(\Omega)}$.