Multigrid Preconditioner for Solving GePUP Equations

Liu jiyu

October 23, 2022

1 GePuP Equations

In a bounded domain $\Omega \in \mathbb{R}^D$, we numerically solve the incompressible Navier-Stokes equations with no-slip condition:

$$\frac{\partial \mathbf{w}}{\partial t} = \mathbf{g} - \mathbf{u} \cdot \nabla \mathbf{u} - \nabla q + \nu \Delta \mathbf{w} \text{ in } \Omega, \tag{1a}$$

$$\mathbf{w} = 0 \text{ on } \partial\Omega, \tag{1b}$$

$$\mathbf{u} = \mathscr{P}\mathbf{w} \text{ in } \Omega, \tag{1c}$$

$$\mathbf{u} \cdot \mathbf{n} = 0 \text{ on } \partial\Omega, \tag{1d}$$

$$\Delta q = \nabla \cdot (\mathbf{g} - \mathbf{u} \cdot \nabla \mathbf{u}) \text{ in } \Omega, \tag{1e}$$

$$\mathbf{n} \cdot \nabla q = \mathbf{n} \cdot (\mathbf{g} + \nu \Delta \mathbf{u} - \nu \nabla \nabla \cdot \mathbf{u}) \text{ on } \partial\Omega. \tag{1f}$$

2 Spatial Discretization

Definition 2.1. R is called a *square domain* if it can be written as $R = [\mathbf{x}_O, \mathbf{x}_E]$, where $\mathbf{x}_O, \mathbf{x}_E \in \mathbb{R}^D$.

Definition 2.2. Ω is called a *regular domain* if it can be written as an union of finite disjoint square domains, i.e.

$$\Omega = \bigcup_{k=1}^{N} R_k, \tag{2}$$

where $R_k,\ k = 1, \ldots, N$ are square domains.

Denote the computational domain as Ω and let R be a regular domain containing Ω, we discretize R into a collection of square control volumes. Denote a control colume by a multi-index $\mathbf{i} \in \mathbb{Z}^D$, then the region of cell $\mathbf{i}$ can be represented by

$$C_{\mathbf{i}} = [\mathbf{x}_O + \mathbf{i}h,\ \mathbf{x}_O + (\mathbf{i} + \mathbb{1}h], \tag{3}$$

and the region of the higher face of cell $\mathbf{i}$ in dimension d by

$$F_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} := \left[\mathbf{x}_\mathbf{O} + \left(\mathbf{i} + \mathbf{e}^d\right) h, \mathbf{x}_O + (\mathbf{i} + \mathbb{1})\, h\right], \tag{4}$$

where $\mathbf{x}_O \in \mathbb{R}^D$ is some fixed origin of the coordinates, h is the uniform grid size, $\mathbb{1} \in \mathbb{Z}^D$ is the multi-index with all its components equal to one, and $\mathbf{e}^d \in \mathbb{Z}^D$ is a multi-index with 1 as its dth component and 0 otherwise.

For an irregular domain Ω, we embed it into a discrete square domain and denote the corresponding cutting control volume and control face as

$$\mathcal{C}_{\mathbf{i}} := C_{\mathbf{i}} \cap \Omega, \quad \mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} := F_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \cap \Omega. \tag{5}$$

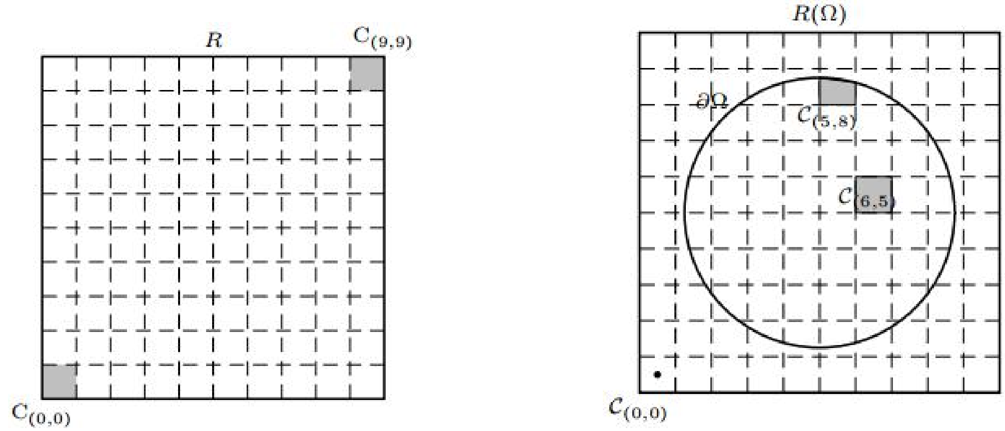
We call $\mathcal{C}_{\mathbf{i}}$ an interior control volume if $\mathcal{C}_{\mathbf{i}} = C_{\mathbf{i}}$, an exterior control volume if $\mathcal{C}_{\mathbf{i}} = \emptyset$, a boundary control volume if $\mathcal{C}_{\mathbf{i}} \neq C_{\mathbf{i}}, \emptyset$. For a boundary control volume, we denote its boundary as

$$\mathcal{S}_{\mathbf{i}} := C_{\mathbf{i}} \cap \partial\Omega. \tag{6}$$

There are the same classifications for control faces. Figure 1 shows the spatial discretization of two different computational domains:

Definition 2.3. Denote the average of a scalar function $\varphi : \mathbb{R}^D \to \mathbb{R}$ on a control volume $\mathcal{C}_{\mathbf{i}}$ as

$$\langle \varphi \rangle_{\mathbf{i}} := \frac{1}{\|\mathcal{C}_{\mathbf{i}}\|} \int_{\mathcal{C}_{\mathbf{i}}} \varphi \, \mathrm{d}\mathbf{x}. \tag{7}$$



(a) Sptial discretization of regular domain R

(b) Spatial discretization $R(\Omega)$ of irregular domain (disc Ω) embedded in the regular domain R, in which $\mathcal{C}_{(0,0)}, \mathcal{C}_{(6,5)}, \mathcal{C}_{(5,8)}$ is respectively an exterior, interior and boundary control volume.

Figure 1: Sptial discretization of computational domain

Definition 2.4. Denote the average of a scalar function $\varphi : \mathbb{R}^D \to \mathbb{R}$ on a control face $\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ as

$$\langle \varphi \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} := \frac{1}{\|\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}\|} \int_{\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}} \varphi \, \mathrm{d}\mathbf{x}. \tag{8}$$

Definition 2.5. Denote the average of a scalar function $\varphi : \mathbb{R}^D \to \mathbb{R}$ on the boundary $\mathcal{S}_\mathbf{i}$ of a control volume as

$$[\varphi]_\mathbf{i} := \frac{1}{\|\mathcal{S}_\mathbf{i}\|} \int_{\mathcal{S}_\mathbf{i}} \varphi \, \mathrm{d}\mathbf{x}. \tag{9}$$

There are three cases when discretizing operators:

1. For control volumes or faces in Ω which are far away from the boundary, we use standard finite volume stencils.
2. For control volumes or faces near the regular boundary, we fill the ghost cells with boundary conditions and then use standard finite volume formulas.
3. For control volumes or faces near the irregular boundary, we generate a poised lattice by poised lattice generating algorithm, then fit a complete multivariate polynomial with the poised lattice to obtain the discretized equation.

2.1 Standard finite difference formulas

In the regular domain, the discrete gradient, the discrete divergence and the dicrete Laplacian are

$$\begin{aligned}
\mathbf{G}_d \langle \phi \rangle_\mathbf{i} &:= \frac{1}{12h} \left(-\langle \phi \rangle_{\mathbf{i}+2\mathbf{e}^d} + 8 \langle \phi \rangle_{\mathbf{i}+\mathbf{e}^d} - 8 \langle \phi \rangle_{\mathbf{i}-\mathbf{e}^d} + \langle \phi \rangle_{\mathbf{i}-2\mathbf{e}^d} \right) \qquad (10) \\
\mathbf{D} \langle \mathbf{u} \rangle_\mathbf{i} &:= \frac{1}{12h} \sum_d \left(-\langle u_d \rangle_{\mathbf{i}+2\mathbf{e}^d} + 8 \langle u_d \rangle_{\mathbf{i}+\mathbf{e}^d} - 8 \langle u_d \rangle_{\mathbf{i}-\mathbf{e}^d} + \langle u_d \rangle_{\mathbf{i}-2\mathbf{e}^d} \right) \qquad (11) \\
\mathbf{L} \langle \phi \rangle_\mathbf{i} &:= \frac{1}{12h^2} \sum_d \left(-\langle \phi \rangle_{\mathbf{i}+2\mathbf{e}^d} + 16 \langle \phi \rangle_{\mathbf{i}+\mathbf{e}^d} - 30 \langle \phi \rangle_\mathbf{i} + 16 \langle \phi \rangle_{\mathbf{i}-\mathbf{e}^d} - \langle \phi \rangle_{\mathbf{i}-2\mathbf{e}^d} \right) \qquad (12) \\
& \qquad (13)
\end{aligned}$$

The discrete divergence also acts on tensor averages

$$\mathbf{D} \langle \mathbf{uu} \rangle_\mathbf{i} := \frac{1}{h} \sum_d \left(\mathbf{F} \langle u_d, \mathbf{u} \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} - \mathbf{F} \langle u_d, \mathbf{u} \rangle_{\mathbf{i}-\frac{1}{2}\mathbf{e}^d} \right), \tag{14}$$

where the average of the product of two scalars over a cell face is

$$\mathbf{F} \langle \phi, \psi \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} := \langle \phi \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \langle \psi \rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + \frac{h^2}{12} \sum_{d' \neq d} \left(\mathbf{G}_{d'}^{\perp} \phi \right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} \left(\mathbf{G}_{d'}^{\perp} \psi \right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}, \tag{15}$$

and $\mathbf{G}_{d'}^{\perp}$ is the discrete gradient in the transverse directions

$$\left(\mathbf{G}_{d'}^{\perp}\phi\right)_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} := \frac{1}{2h}\left(\langle\phi\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d+\mathbf{e}^{d'}} - \langle\phi\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d-\mathbf{e}^{d'}}\right) \tag{16}$$

$\mathbf{G}\langle\phi\rangle$, $\mathbf{D}\langle\mathbf{u}\rangle$, $\mathbf{L}\langle\mathbf{u}\rangle$ and $\mathbf{D}\langle\mathbf{uu}\rangle$ approximate the cell average of a gradient, velocity divergence, Laplacian, and convection, respectively. These discrete operators are fomally fourth-order accurate in approximating their continuous counterparts.[3][4]

2.2 Ghost Cell

Following [3], two layers of ghost cells are used to enforce boundary conditions. For nonperiodic boundaries, the values of ghost cells are obtained by extrapolating those of the interior cells, with the boundary conditions incorporated in the extrapolation formulas. For example, homogeneous Dirichlet boundary conditions for a scalar ψ are fulfilled by filling the ghost cells with the following fifth-order formulas:

$$\begin{aligned}\langle\psi\rangle_{\mathbf{i}+\mathbf{e}^d} &= \frac{1}{12}\left(-77\langle\psi\rangle_{\mathbf{i}} + 43\langle\psi\rangle_{\mathbf{i}-\mathbf{e}^d} - 17\langle\psi\rangle_{\mathbf{i}-2\mathbf{e}^d} + 3\langle\psi\rangle_{\mathbf{i}-3\mathbf{e}^d}\right) + O(h^5),\\ \langle\psi\rangle_{\mathbf{i}+2\mathbf{e}^d} &= \frac{1}{12}\left(-505\langle\psi\rangle_{\mathbf{i}} + 335\langle\psi\rangle_{\mathbf{i}-\mathbf{e}^d} - 145\langle\psi\rangle_{\mathbf{i}-2\mathbf{e}^d} + 27\langle\psi\rangle_{\mathbf{i}-3\mathbf{e}^d}\right) + O(h^5),\end{aligned} \tag{17}$$

Neumann boundary conditions are fulfilled by

$$\begin{aligned}\langle\psi\rangle_{\mathbf{i}+\mathbf{e}^d} &= \frac{1}{10}\left(5\langle\psi\rangle_{\mathbf{i}} + 9\langle\psi\rangle_{\mathbf{i}-\mathbf{e}^d} - 5\langle\psi\rangle_{\mathbf{i}-2\mathbf{e}^d} + \langle\psi\rangle_{\mathbf{i}-3\mathbf{e}^d}\right) + \frac{6}{5}h\left\langle\frac{\partial\psi}{\partial n}\right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^5),\\ \langle\psi\rangle_{\mathbf{i}+2\mathbf{e}^d} &= \frac{1}{10}\left(-75\langle\psi\rangle_{\mathbf{i}} + 145\langle\psi\rangle_{\mathbf{i}-\mathbf{e}^d} - 75\langle\psi\rangle_{\mathbf{i}-2\mathbf{e}^d} + 15\langle\psi\rangle_{\mathbf{i}-3\mathbf{e}^d}\right) + 6h\left\langle\frac{\partial\psi}{\partial n}\right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d} + O(h^5),\end{aligned} \tag{18}$$

where $\left\langle\frac{\partial\psi}{\partial n}\right\rangle_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$ is the Neumann condition for ψ.

2.3 PLG

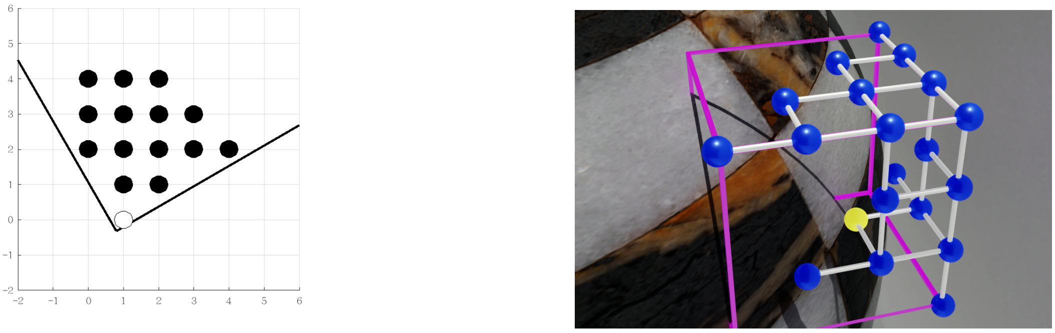
For the control volumes and faces near the irregular boundary, where we can not use standard difference formulas, our idea is:

1. Clarify the control volume $\mathcal{C}_{\mathbf{i}}$ (or control face $\mathcal{F}_{\mathbf{i}+\frac{1}{2}\mathbf{e}^d}$) where differential operator $\mathcal{L}$ is to be discretized, the order of spatial discretization p, and the average of differential operator on control volumes $\left\{\langle\varphi\rangle_{\mathbf{j}} : \mathbf{j}\in\mathbb{Z}^D\right\}$ (or on control faces $\left\{\langle\varphi\rangle_{\mathbf{j}} : \mathbf{j}\in\mathbb{Z}^D + \frac{1}{2}\mathbf{e}^d\right\}$).

2. As shown in Figure 2, we calculate finite volume stencil of $\langle\mathcal{L}\varphi\rangle_{\mathbf{i}}$ with poised lattice generating algorithm, denote it as
$$\mathcal{X}(\mathbf{i}) = \{\mathcal{C}_{\mathbf{j}_1},\cdots,\mathcal{C}_{\mathbf{j}_N}\}\cup\{\mathcal{S}_{\mathbf{j}_{N+1}},\cdots,\mathcal{S}_{\mathbf{j}_{N+N'}}\} \tag{19}$$
where $N=\dim\Pi_n^D$, $n=p+q-1$ is the order of stencil, q is the order of differential operator, N' is the number of boundary terms.

3. Fit a complte multivariate polynomial with the finite volume stencil $\mathcal{X}(\mathbf{i})$:
$$p(\mathbf{x}) = \sum_{j=1}^{N}\alpha_j\phi_j(\mathbf{x})\in\Pi_n^D, \tag{20}$$
where $\{\phi_j\}_{j=1}^N$ is a basis of Π_n^D, coefficient $\alpha=[\alpha_1,\cdots,\alpha_n]^T$ is the solution of weighted least squares problem
$$\min_{\alpha}\sum_{k=1}^{N}\omega_k\left|\langle p\rangle_{\mathbf{j}_k}-\varphi_k\right|^2 + \sum_{k=N+1}^{N=N'}\omega_k\left|[\mathcal{N}p]_{\mathbf{j}_k}-\varphi_k\right|^2, \tag{21}$$
where weight ω_k depends on the relative position of $\mathcal{C}_{\mathbf{j}_k}$ and $\mathcal{C}_{\mathbf{i}}$.

4. Finally discretize the differential operator with fitted polynomial (20):
$$\langle\mathcal{L}\varphi\rangle_{\mathbf{i}} = \langle\mathcal{L}p\rangle_{\mathbf{i}} + O(h^p) = \sum_{k=1}^{N}\beta_k\langle\varphi\rangle_{\mathbf{j}_k} + \sum_{k=N+1}^{N+N'}\beta_k[\mathcal{N}\varphi]_{\mathbf{j}_k} + O(h^p). \tag{22}$$



(a) A poised lattice in Π_4^2 at the hollow dot

(b) A poised lattice in Π_3^3 at the hollow dot

Figure 2: For a given starting point, we choose a set of nodes to fit a high-order, multi-variate, complete polynomial, which facilitates the discretization of spatial operators and the enforcement of boundary conditions.

Now we obtain the fourth-order (spatial) semi-discrete formulas of GePuP (1) for incompressible Navier-Stokes equations.

$$\begin{aligned}
\tfrac{\mathrm{d}\langle \mathbf{w}\rangle}{\mathrm{d}t} &= \langle \mathbf{g}\rangle - \mathbf{D}\langle \mathbf{u}\mathbf{u}\rangle - \mathbf{G}\langle q\rangle + \nu\mathbf{L}\langle \mathbf{w}\rangle \ \text{ in } \Omega, && \text{(23a)}\\
[\mathbf{w}] &= 0 \text{ on } \partial\Omega, && \text{(23b)}\\
\langle \mathbf{u}\rangle &= \mathbb{P}\langle \mathbf{w}\rangle \ \text{ in } \Omega, && \text{(23c)}\\
[\mathbf{u}\cdot\mathbf{n}] &= 0 \text{ on } \partial\Omega, && \text{(23d)}\\
\mathbf{L}\langle q\rangle &= \mathbf{D}(\langle \mathbf{g}\rangle - \mathbf{D}\langle \mathbf{u}\mathbf{u}\rangle) \text{ in } \Omega, && \text{(23e)}\\
[\mathbf{n}\cdot\nabla q] &= [\mathbf{n}\cdot\mathbf{g}] + \nu\,[(\mathbf{n}\cdot\mathbf{L})\mathbf{u}] - \nu\,[(\mathbf{n}\cdot\mathbf{G})\mathbf{D}\mathbf{u}] \ \text{ on } \partial\Omega. && \text{(23f)}
\end{aligned}$$

with the initial conditions

$$\begin{aligned}
\langle \mathbf{u}\rangle (t_0) &= \langle \mathbf{w}\rangle (t_0), && \text{(24a)}\\
[\mathbf{w}] &= \mathbf{0}. && \text{(24b)}
\end{aligned}$$

3 Time Integration

We apply ERK-ESDIRK to semi-discrete formulae (23), in which the RHS of (23a) is divided into an explicit part and an implicit part:

$$\begin{aligned}
\mathbf{X}^{[E]} &= \langle \mathbf{g}\rangle - \mathbf{D}\langle \mathbf{u}\mathbf{u}\rangle - \mathbf{G}\langle q\rangle, && \text{(25a)}\\
\mathbf{X}^{[I]} &= \nu\mathbf{L}\langle \mathbf{w}\rangle. && \text{(25b)}
\end{aligned}$$

Now we obtain GePuP-IMEX algorithm:

$$\langle \mathbf{w}\rangle^{(1)} = \langle \mathbf{w}\rangle^{n}, \tag{26a}$$

$$\left\{\begin{aligned}
&\text{for } s = 2, 3, \cdots, n_s,\\
&(\mathbf{I} - \Delta t\gamma\nu\mathbf{L})\langle \mathbf{w}\rangle^{(s)} = \langle \mathbf{w}\rangle^{n} + \Delta t\sum_{j=1}^{s-1} a_{s,j}^{[E]}\mathbf{X}^{[E]}\left(\langle \mathbf{u}\rangle^{(j)}, t^{(j)}\right) + \Delta t\nu\sum_{j=1}^{s-1} a_{s,j}^{[I]}\mathbf{L}\langle \mathbf{w}\rangle^{(j)},\\
&\langle \mathbf{u}\rangle^{(s)} = \mathbb{P}\langle \mathbf{w}\rangle^{(s)},
\end{aligned}\right. \tag{26b}$$

$$\left\{\begin{aligned}
&\langle \mathbf{w}\rangle^{*} = \langle \mathbf{w}\rangle^{(n_s)} + \Delta t\sum_{j=1}^{n_s}\left(b_j - a_{n_s,j}^{[E]}\right)\mathbf{X}^{[E]}\left(\langle \mathbf{u}\rangle^{(j)}, t^{(j)}\right),\\
&\langle \mathbf{u}\rangle^{n+1} = \mathbb{P}\langle \mathbf{w}\rangle^{*},\\
&\langle \mathbf{w}\rangle^{n+1} = \langle \mathbf{u}\rangle^{n+1}.
\end{aligned}\right. \tag{26c}$$

There are three linear systems of equations need to be solved at each intermediate stage:

1. A Poisson equation with Neumann boundary condition for extracting $\langle q\rangle$ from $\langle \mathbf{u}\rangle$.
2. A Helmholtz equation with no-slip boundary condition for evaluating $\langle \mathbf{u}\rangle$.
3. A Poisson equation with Neumann boundary condition for projecting $\langle \mathbf{w}\rangle^{*}$.

It can be seen that the essence of our work is solving a Poisson equation $L\phi = b$ on an irregular domain, the spatial discretization is the same as that in Section 2.

4 The multigrid preconditioned Krylov methods

We concentrate on the system of discrete equations

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{27}$$

Matrix $\mathbf{A}$ has right preconditioning as follows:

$$\mathbf{A}\mathbf{K}^{-1}(\mathbf{K}\mathbf{x}) = \mathbf{b}. \tag{28}$$

The preconditioned Krylov subspace methods are used for solving (28), such as BiCGSTAB and GMRES(m).

Algorithm 4.1. The GMRES(m) algorithm with a right multigrid preconditioner appears as follows:

```
Algorithm 1: GMRES($m, A, b, x, \epsilon$)
    Input: $m \in \mathbb{Z}^+$, $A \in \mathbb{R}^{N \times N}$, $b \in \mathbb{R}^n$, $x \in \mathbb{R}^n$, $\epsilon \in \mathbb{R}^+$. .
    Output: The solution which overwrites $x$.
 1  Choose $x^{(0)} = x$, dimension $m$. matrix $\mathbf{H} = \mathbf{0}$ with dim: $(m+1) \times m$;
 2  $r^{(0)} = b - Ax^{(0)}$, $\beta = \|r\|_2^2$, $f_1 = r^{(0)}/\beta$;
 3  for $j = 1, \cdots, m$ do
 4      $u_j = K^{-1} f_j$;
 5      $w = Au_j$;
 6      for $i = 1, \cdots, j$ do
 7          $h_{i,j} = (w, f_i)$;
 8          $w = w - h_{i,j} f_j$;
 9          $h_{j+1,j} = \|w\|_2$;
10          $f_{j+1} = w/h_{j+1,j}$;
11      end
12  end
13  Define $\mathbf{F}_m := [f_1, \cdots, f_m]$;
14  $x^{(m)} := x^{(0)} + K^{-1}\mathbf{F}_m y_m$ with $y_m = \min_y \|\beta e_1 - \mathbf{H}y\|_2$;
15  Compute $r^{(m)} = b - Ax^{(m)}$;
16  if $|r^{(m)}\|_2 < \epsilon \|b\|_2$ then
17      Stop.
18  end
19  else
20      restrart with $x^{(0)} \leftarrow x^{(m)}$;
21  end
```

In line 4, $K^{-1} f_j$ is the preconditioning step, which is one iteration of a multigrid cycle.

The following theorem is an estimation of the convergence of GMRES(m) in cases where most of the eigenvalues (the last $n - l$ eigenvalues in the theorem below) of the preconditioned matrix $\tilde{A}$ are close to $1 \in \mathbb{C}$.

Theorem 4.2. Let $\tilde{A}$ be an $n \times n$ nonsingular matrix with eigenvalues $\{\lambda_i \in \mathbb{C} \mid 1 \leq i \leq n\}$, $\overline{V}_l$ be the subspace spanned by the vectors $\left\{v \mid \prod_{k>l}^{n}(\lambda_k I - \tilde{A})v = 0\right\}$, K_l be the Krylov subspave $K(l, r^{(0)}, \tilde{A}) = \operatorname{span}\left\{r^{(0)}, \tilde{A}r^{(0)}, \cdots, \tilde{A}^{l-1}r^{(0)}\right\}$, P_k be a set of kth-order complex polynomials $p_k(\lambda)$ that satisfy $p_k(0) = 1$, and $r^{(0)} = b - \tilde{A}x^{(0)}$ be the initial residual. Define radius Γ_i as

$$\Gamma_i := \max\left\{\|(\lambda_i I - \tilde{A})v\|_2 \mid v \in \overline{V}_{i-1} \cap K_i, \|v\|_2 = 1\right\}. \tag{29}$$

Then a vector $\overline{r}^{(l)}$ defined by

$$\overline{r}^{(l)} := \left(\prod_{i=1}^{l} \frac{1}{\lambda_i}(\lambda_i I - \tilde{A})\right) r^{(0)} \tag{30}$$

is included in $\overline{V}_l \cap K_{l+1}$.

Furthermore,

$$\|\overline{r}^{(l)}\|_2 \leq \left(\prod_{i=1}^{l} \frac{\Gamma_i}{|\lambda_i|}\right) \|r^{(0)}\|_2. \tag{31}$$

Assuming $l < k \leq m$, then the norm of the residual of kth GMRES(m) iteration can be estimated as follows:

$$\begin{aligned} \|r^{(k)}\|_2 &\leq \min\left\{\|p_{k-l}(\tilde{A})r^{(l)}\|_2 \mid p_{k-l}(\lambda) \in P_{k-l}\right\} && (32) \\ &\leq \|(I - \tilde{A})^{k-l}\overline{r}^{(l)}\|_2. && (33) \end{aligned}$$

According to Theorem 4.2, if some eigenvalues $\lambda_i (i \leq l)$ are close to zero, the norm of $\overline{r}^{(l)}$ becomes large. This suggests that we may need a relatively large k to reduce the residual by a certain order of magnitude even if l is not large.

Inequality (32) shows that $\|r^{(k)}\|_2$ is not larger than the norm of the $(k-l)$th residual of GMRES(m) with initial residual $\overline{r}^{(l)}$ that is included in the subspace corresponding to the eigenvalues close to one $\{\lambda_i \mid i > l\}$.

Backtrack to the multigrid preconditioner, an iteration of a multigrid cycle is equivalent to a Richardson iteration on the preconditioned matrix[2]. With K being the iteration matrix, multigrid can be written as follows:

$$Kx^{(k+1)} + (A-K)x^{(k)} = b. \tag{34}$$

If we use multigrid solver, the formulation is equivalent to

$$x^{(k+1)} = x^{(k)} + K^{-1}(b - Ax^{(k)}) = x^{(k)} + K^{-1}r^{(k)};\ r^{(k+1)} = (I - AK^{-1})r^{(k)}. \tag{35}$$

The spectral radius of $I - AK^{-1}$ determines the convergence of multigrid solver. With this spectrum we can also investigate the convergence of preconditioned GMRES method. From (33), the asymptotic convergence of GMRES(m) is at least faster than Richardson with an initial residual.

In [2], the author compared multigrid preconditioned GMRES(m) and multigrid solver in several singularly perturbed problems. In [1], the author compared multigrid preconditioned GMRES(m) and multigrid solver for implicit immersed boundary equations. Here is an example:

Example 4.3. Here is a rotated anisotropic diffusion problem:

$$-(\cos^2\beta + \epsilon\sin^2\beta)\frac{\partial^2\phi}{\partial x^2} - 2(\epsilon - 1)\cos\beta\sin\beta\frac{\partial^2\phi}{\partial x\partial y} - (\epsilon\cos^2\beta + \sin^2\beta)\frac{\partial^2\phi}{\partial y^2} = 1 \text{ on } \Omega = (0,1)\times(0,1). \tag{36}$$

Here $\epsilon = 10^{-5}$, $\beta = 135^o$ and boundary conditions are prescribed :

$$\begin{aligned} \frac{\partial\phi}{\partial n} &= 0 \text{ on } \{x = 0,\ 0 \leq y \leq 1\}, \{0 \leq x \leq 1,\ y = 0\}, \\ \phi &= 0 \text{ on } \{x = 1,\ 0 \leq y \leq 1\}, \{0 \leq x \leq 1,\ y = 1\}. \end{aligned} \tag{37}$$

For this test problem, the eigenvalue spectrum of the Richardson iteration matrix $I - AK^{-1}$ is investigated on a 33×33 grid, as presented in figure 3.

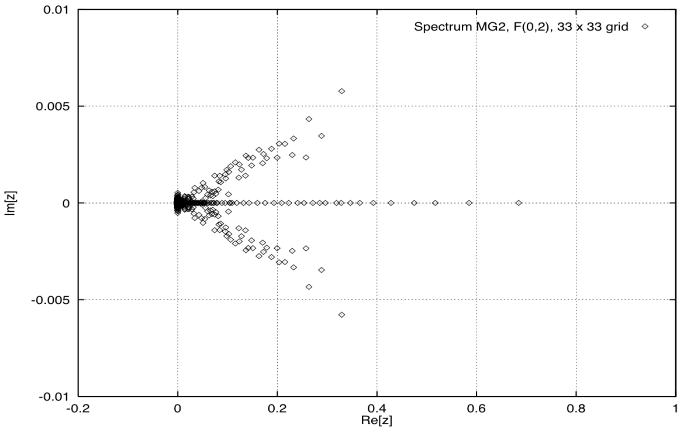


Figure 3: The eigenvalue spectra for the rotated anisotropic diffusion problem,$\epsilon = 10^{-5}$, $\beta = 135^o$ on a 33×33 grid.

It can be seen that the spectral radius for coarse grid problems is already larger than 0.6. Therefore, the multigrid convergence slows down more dramatically than the convergence of the preconditioned Krylov methods. What's more, eigenvalues of the preconditioned matrix (AK^{-1}) are clusted around 1, which is advantageous for the Krylov methods. The convergence of multigrid as solution methods and as preconditioners for GMRES(20) is presented for 33×33 grid in Figure 4.

MG1, MG2, MG3 are multigrid methods with different prolongation and restriction operators. It can be seen that different prolongation and restriction operators will affect the convergence rate. However, no matter which operators are used, MG preconditioned GMRES(m) method performs better than MG solver in this problem.

The conclusion is that the behaviors of the multigrid methods are much more robust when they are used as preconditioners, problems that could not be solved with the multigrid methods as solvers could be solved with the preconditioned Krylov methods. The efficiency of the multigrid solver alone is not impressive, but it is a very effective preconditioner for Krylov methods. We also can improve the preconditioned Krylov methods by using a more sophisticated smoother in multigrid preconditioner, just like Figure 4.

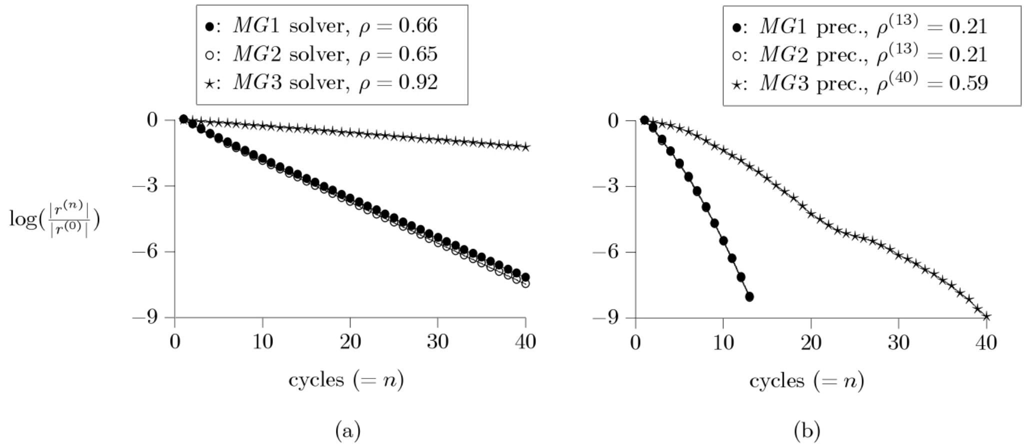


Figure 4: The convergence of the MG solvers (a), GMRES(20) with MG preconditioners (b) for the rotated anisotropic diffusion equation on a 33 × 33 grid.

5 Next step

The follow-up work is mainly divided into two steps:

1. There is not much mathematical theory on multigrid preconditioned Krylov subspace methods, most papers do numerical experience to compare multigrid solver and multigrid preconditioner. So firstly we could investigate the eigenvalue spectrum of the Richardson iteration matrix $I - AK^{-1}$ in (35) of three linear systems during solving GePuP equations. If many eigenvalues are clustered around zero and only a limited number of eigenvalues are far from zero, eigenvalues of the preconditioned matrix AK^{-1} will be clustered around one, which is advantageous for the Krylov methods by Theorem 4.2.
2. Then implement multigrid preconditioned GMRES(m) method, apply it to solve GePuP equations, and compare it with the original program in terms of convergence and efficiency.

References

[1] Robert D. Guy and Bobby Philip. A multigrid method for a model of the implicit immersed boundary equations. *Communications in Computational Physics*, 12(2):378–400, 2012. Publisher: Cambridge University Press.

[2] Cornelis W. Oosterlee and Takumi Washio. An evaluation of parallel multigrid as a solver and a preconditioner for singularly perturbed problems. *SIAM Journal on Scientific Computing*, 19(1):87–110, 1998. Publisher: SIAM.

[3] Qinghai Zhang. A fourth-order approximate projection method for the incompressible Navier–Stokes equations on locally-refined periodic domains. *Applied Numerical Mathematics*, 77:16–30, 2014. Publisher: Elsevier.

[4] Qinghai Zhang, Hans Johansen, and Phillip Colella. A fourth-order accurate finite-volume method with structured adaptive mesh refinement for solving the advection-diffusion equation. *SIAM Journal on Scientific Computing*, 34(2):B179–B201, 2012. Publisher: SIAM.