Section 08. Algebraic Multigrid Methods

From GMG To AMG

Q: Anisotropic coefficients? Jump coefficients? Heterogenous coefficients? Non-symmetric problems?

- Prolongation and restriction for GMG usually depend only on the topological structure of the grids without using the grid coordinates.
- In GMG, the topologies of the graph for the stiffness matrices on different levels are explicitly clear from the geometric refinement procedure.
- The key to an efficient GMG algorithm is to construct effective and cheap smoothers for the problem at hand.
- For GMG, smoothness of error is defined geometrically. In more general settings, a geometrically smooth error can be non-smooth.
- AMG algorithms employ simple smoothers and attempt to construct a suitable operator-dependent interpolation using the algebraic information of A.
- AMG algorithms focus on how to pick coarse space and constructing interpolation to approximate the error components that cannot be effectively reduced by smoothing.

Toward Good AMG Performance

1. Some simple graph theory results and algorithms
    - Maximum independent set
    - Ordering and Tarjan's algorithm
2. Designing algebraic multigrid methods
    - General setup methods for AMG
    - Two-level convergence theory
    - Algebraic smoothness
    - Prolongation and restriction operators
3. Some examples of AMG methods
    - Classical AMG
    - Classical AMG with aggressive coarsening
    - Smoothed/unsmoothed aggregation AMG
4. Convergence results of AMG methods

Maximum Independent Set

- Set of vertices $\subseteq \{1, \ldots, N\}$. An edge in E is an unordered pair $(j,\ k)$ with $j,\ k \in \mathsf{V}$.
- An undirected graph $\mathsf{G} := (\mathsf{V}, \mathsf{E})$, where V is a set of vertices and E is a set of edges.
- A graph $\mathsf{G}_0 = (\mathsf{V}_0, \mathsf{E}_0)$ is called a subgraph of $\mathsf{G} = (\mathsf{V}, \mathsf{E})$, if $\mathsf{V}_0 \subset \mathsf{V}$ and $\mathsf{E}_0 \subset \mathsf{E}$.
- If $(j, k) \in \mathsf{E}$ is an edge in an undirected graph $\mathsf{G} = (\mathsf{V},\ \mathsf{E})$, vertices j and k are said to be adjacent.
- The set of neighboring vertices $\mathsf{N}_i \subseteq \mathsf{V}$ of i is the set of all vertices that are adjacent to i.
- An independent set of G is a set of vertices where no two vertices are adjacent.
- A maximal independent set (mIS) is an independent set such that adding any other vertex would introduce an adjacent pair. A graph can have many mIS's of varying sizes.
- The largest mIS, or several equally large mIS's, is called a maximum independent set or MIS.
- MIS problem: Find an independent set in the graph $\mathsf{G} = (\mathsf{V},\ \mathsf{E})$ of maximum cardinality.

However, finding MIS is NP-hard and it is natural to ask for approximation algorithms.

Path and Adjacency Graph

- A path from a vertex i to another vertex j is a sequence of edges

$$p(i,j) := \big\{(i,\, k_1),\, (k_1, k_2),\, \ldots,\, (k_{\ell-2}, k_{\ell-1}),\, (k_{\ell-1}, j)\big\} \subseteq \mathsf{E};$$

  the number of edges ℓ is called the length of this path.

- A vertex j is connected to another vertex i if there is a path from j to i.
- The distance between j and i is defined as the length of the shortest path between them.
- The distance between j and i is equal to 1 if they are adjacent, and is ∞ if they are not connected.
- An undirected graph G is connected if any pair of vertices are connected by a path.
- The adjacency graph $\mathsf{G}(A)$ of matrix $A \in \mathbb{R}^{N\times N}$ is a graph $\mathsf{G} = (\mathsf{V},\, \mathsf{E})$ with

$$\mathsf{V} := \{1, 2, \ldots, N\} \quad \text{and} \quad \mathsf{E} := \big\{(i,\, j) \,:\, a_{i,j} \neq 0\big\}.$$

Sparse matrices do not provide geometric information for the underlying graph except the combinatorial/topological properties of $\mathsf{G}(A)$ or its subgraphs.

Coefficient Matrix and Graph

Different discretizations on different meshes could lead to same coefficient matrix A and $\mathsf{G}(A)$.

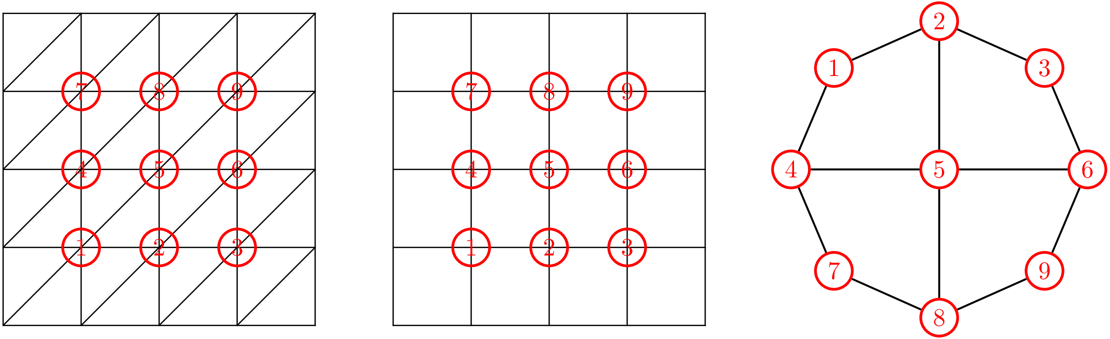


Figure: Finite element grid (left), finite difference grid (middle), and graph of stiffness matrices (right).

M-Matrix And Its Properties

Definition (M-matrix)

A matrix A is an M-matrix if it is irreducible (i.e., the graph $\mathsf{G}(A)$ is connected) and

$$a_{i,i} > 0, \quad a_{i,j} \leq 0 \;\; (i \neq j), \quad \text{and} \quad a_{j,j} \geq \sum_{i \neq j} \left| a_{i,j} \right|, \quad a_{j,j} > \sum_{i \neq j} \left| a_{i,j} \right| \;\; \text{for at least one } j.$$

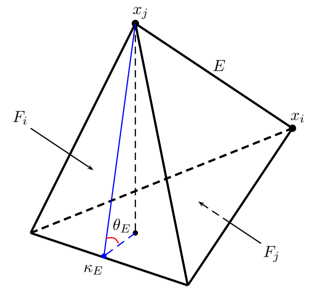
- Stiffness matrices of FE discretizations might not be M-matrices in general.
- Whether a stiffness matrix is an M-matrix depends on the underlying mesh $\mathcal{M}$.
- We can apply simple filtering schemes to construct an approximate M-matrix A_M from A.

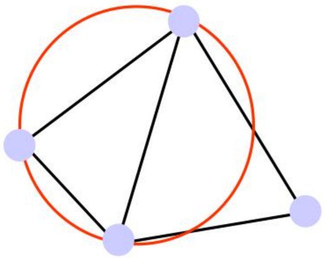
Remark (Delaunay triangulation and M-matrix):

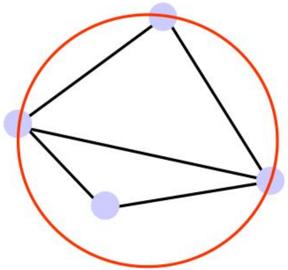
The stiffness matrix for the Poisson's equation is an M-matrix if the triangulation is Delaunay. This condition is nearly sharp — The opposite direction is true with a few possible exceptions near the boundary.

FE Matrix And M-Matrix

Let $\omega_E(\tau) := \frac{1}{d(d-1)} \big|\kappa_E(\tau)\big| \cot \theta_E(\tau)$.







Proposition (Condition for M-matrix)

The stiffness matrix for the Poisson's equation is an M-matrix, if and only if

$$\sum_{\tau \supset E} \omega_E(\tau) \geq 0, \quad \forall\, E.$$

Delaunay Triangulation and M-Matrix

Definition (Delaunay triangle)

In the context of a finite point set V, a triangle is Delaunay if its vertices are in V and its open circumdisk is empty—namely, it contains no point of V.

- Any number of points in V can lie on a Delaunay triangle's circumcircle.
- An edge is Delaunay if its vertices are in V and it has at least one empty open circumdisk.
- A Delaunay triangulation of V is a triangulation in which every triangle is Delaunay.

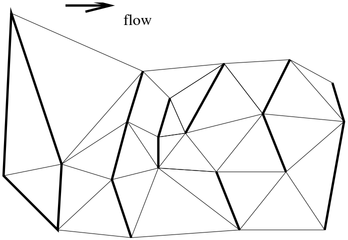
Delaunay triangulation is a foundational geometric structure:

- In 2D, among all possible triangulations of given points, the Delaunay triangulation maximizes the minimum angle, thereby reducing the occurrence of narrow and sharp angles.
- The Delaunay triangulation improves interpolation accuracy.
- Extensive research has been dedicated to Delaunay triangulations, resulting in highly efficient algorithms for constructing and updating these meshes.

Cross-Wind Ordering

Reminder: Relaxation order is very important for the local relaxation methods like GS.

Example: Convection-diffusion problem



$$\Longrightarrow A = \begin{bmatrix} \boxed{A_{11}} & \approx \epsilon & \approx \epsilon & \cdots & \cdots \\ \boxed{A_{21}} & \boxed{A_{22}} & \approx \epsilon & \cdots & \cdots \\ \boxed{A_{31}} & \boxed{A_{32}} & \boxed{A_{33}} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \ddots & \cdots \\ \boxed{A_{K1}} & \boxed{A_{K2}} & \boxed{A_{K3}} & \cdots & \boxed{A_{KK}} \end{bmatrix}$$

- In the above figure, the bold edges represent the cross-wind connections:
- If we number the blocks sequentially from left to right, with the degrees of freedom within each block ordered arbitrarily, the stiffness matrix will have a lower triangular structure.
- Based on this idea, the Cross-Wind-Block method has been proposed for convection-dominated problems [Wang and Xu 1999]

Examples of Cross-Wind Ordering

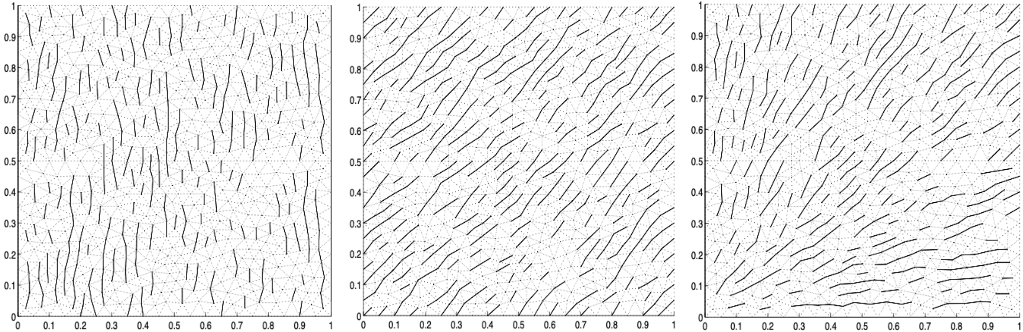


Figure: Flow velocity $v = (1, 0)$ (left), $v = (-1, 1)$ (middle), and $v = (-y, x)$ (right) [Wang and Xu 1999]

Coefficient Matrix and Graph

Strongly connected component:

- A maximal subset of vertices such that there is a path between any two vertices in the subset.
- Tarjan's algorithm is used to find strongly connected components in a graph.

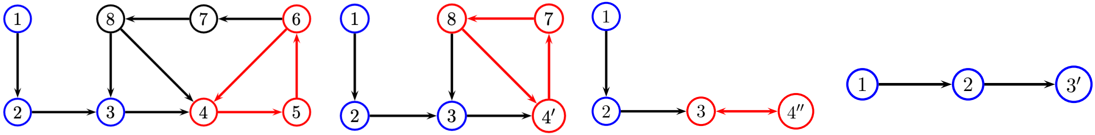
Tarjan's algorithm uses depth-first search on the digraph:

1. Drop some of the non-essential entries from the matrix A — Transform the graph corresponding to A to a directed graph.
2. Find the strongly connected components in this directed graph — Each one of these components will correspond to a diagonal block in the stiffness matrix after permutation.

> **Remark (Preprocessing to a get directed graph):**
>
> Sometimes the graph corresponding to A is undirected (for example, the finite element stiffness matrix of the Poisson's equation). If we consider a non symmetric problem, situation could be very different. For instance, we can make the graph directed by dropping some of the "insignificant" entries of A. For example, by setting a threshold $\epsilon \in (0,1)$, we drop all a_{ki} if $|a_{ki}/a_{ik}| < \epsilon$.

A Simple Example of Graph Ordering

Consider a 2D flow problem



- Start from vertex 1 and then follow the path

$$1 \to 2 \to 3 \to 4 \to 5 \to 6 \to 4.$$

- Collapse $\{4, 5, 6\}$ as a single vertex $v_{\text{macro}} = 4'$.
- Following the path and continue until

$$1 \to 2 \to 3 \to 4' \to 7 \to 8 \to 4'.$$

- Collapse the cycle and set $4'' = \{4', 7, 8\}$.
- Collapse the cycle $\{4'', 3, 4''\}$ to a vertex $3'$.
- What is left now is a simple graph with three vertices.

Tarjan's Algorithm

Algorithm (Simplified Tarjan's algorithm)

Given a directed graph G with N vertices.

1. If all vertices of G have been numbered, stop.
2. Set $i = 0$.
3. Choose any unnumbered vertex $v \in$ G.
4. If v has no edge out, we number it $N - i$, set $i = i + 1$, and return to Step 3.
5. If v has been visited before (encounter a cycle), then
    - Collapse all the vertices in the cycle as a single vertex v_{macro};
    - Connect v_{macro} with all vertices which were connected to member(s) of v_{macro};
    - Thus we obtain a new graph G$'$. Goto Step 2 and continue with G$'$.

There might be a lot of renumbering when collapsing the cycles. The fix is to use a stack and do not renumber anything until the whole connected component is in the stack.

Algebraic Setup Methods

Q: How to construct multilevel hierarchy in an algebraic fashion? How to construct AMG to improve efficiency (convergence and complexity)?

Algorithm (Setup step for algebraic multigrid methods)

For a given sparse matrix $A \in \mathbb{R}^{N\times N}$, we apply the following steps:

Step 1. Selecting a smoother: Choose a smoother S for $Au = f$.

Step 2. Coarsening: Identify a coarse space $V_c \subset V$, which contains smooth vectors.

Step 3. Constructing a prolongation: Construct a prolongation P in two steps:

3a. Decide, for each fine variable, which coarse variables are used for interpolation;

3b. Determine the weights for prolongation P.

Step 4. Multilevel cycling: Apply the same algorithm one or more times for the coarse problem $A_c u_c = f_c$, where $A_c = P^T AP$ and $f_c = P^T f$.

Standard Setting for AMG

Assumption (Standard AMG setting)

The coefficient matrix A is SPD, the prolongation P has full column-rank, and the smoother S is given.

- Let $V = \mathbb{R}^N$ and $V_c = \mathbb{R}^{N_c}$ denote the fine and coarse spaces, respectively.
- For simplicity, we only consider the $\mathrm{V}(0,1)$ exact two-grid method.
- The CGC operator corresponds to the matrix $I - \Pi_c$ and

$$\Pi_c = PA_c^{-1}P^TA = P(P^TAP)^{-1}P^TA$$

  is a projection onto $\mathrm{range}(P)$.

- The error reduction matrix for the TG method can be written as

$$E_{\mathrm{TG}} := (I - SA)(I - \Pi_c).$$

The convergence rate of TG depends on (1) effectiveness of the smoother S and (2) approximability of the coarse space $\mathrm{range}(P)$.

Convergence Theory for TG

Theorem (Convergence rate of TG algorithm)

If there exists $\delta > 0$ such that

$$\big\|(I - SA)e\big\|_A^2 \le \|e\|_A^2 - \delta\big\|(I - \Pi_c)e\big\|_A^2, \quad \forall\, e \in V, \tag{64}$$

then the $\mathrm{V}(0,1)$ two-grid method satisfies that

$$\|E_{\mathrm{TG}}\|_A^2 = 1 - \hat{\delta}, \quad \text{with } \hat{\delta} := \inf_{(I-\Pi_c)e \neq 0} \frac{\|e\|_A^2 - \|(I - SA)e\|_A^2}{\|(I - \Pi_c)e\|_A^2} \ge \delta.$$

Sketch of the proof:

Notice that $\|e\|_A^2 = \|\Pi_c e\|_A^2 + \|(I - \Pi_c)e\|_A^2$. $(I - \Pi_c)e = 0$ yields $(I - SA)(I - \Pi_c)e = 0$. Then

$$\|E_{\mathrm{TG}}\|_A^2 = \sup_{e \neq 0} \frac{\|(I - SA)(I - \Pi_c)e\|_A^2}{\|e\|_A^2} = \sup_{(I-\Pi_c)e \neq 0} \frac{\|(I - SA)(I - \Pi_c)e\|_A^2}{\|(I - \Pi_c)e\|_A^2 + \|\Pi_c e\|_A^2}.$$

If $\hat{e}$ achieves the above sup, then $(I - \Pi_c)\hat{e}$ also achieves the sup. So the convergence factor achieves the supremum when $\Pi_c \hat{e} = 0$. Hence, $\big\|E_{\mathrm{TG}}\big\|_A^2 = \sup_{(I-\Pi_c)e \neq 0} \big\|(I - SA)e\big\|_A^2 / \big\|(I - \Pi_c)e\big\|_A^2$. □

Remarks on TG Convergence

Condition (64) is essential for TG to be effieicnet

- For the error components that cannot be reduced by CGC, the smoother S must be effective uniformly.
- For the error components that can be reduced by CGC efficiently, S could be ineffective.
- The components for which S is ineffective are called smooth and they should be in $\mathrm{range}(P)$.

Modification: Such a $\hat{\delta}$ is difficult to obtain and we need to find some positive lower bounds of $\hat{\delta}$. Introduce a nonnegative function $g(e) \geq 0$, such that

$$\alpha_g(e) := \frac{\|e\|_A^2 - \|(I - SA)e\|_A^2}{g(e)} \qquad \text{and} \qquad \beta_g(e) := \frac{\|(I - \Pi_c)e\|_A^2}{g(e)}.$$

Remark: Let $\hat{\alpha}_g := \inf_{g(e)\neq 0} \alpha_g(e)$ and $\hat{\beta}_g := \sup_{g(e)\neq 0} \beta_g(e)$. Due to the fact

$$\begin{aligned}\|E_{\mathrm{TG}}e\|_A^2 &\leq \left\|(I-\Pi_c)e\right\|_A^2 - \hat{\alpha}_g g\big((I-\Pi_c)e\big) \leq \left\|(I-\Pi_c)e\right\|_A^2 - \hat{\alpha}_g\hat{\beta}_g^{-1}\left\|(I-\Pi_c)e\right\|_A^2 \\ &= \left(1 - \hat{\alpha}_g\hat{\beta}_g^{-1}\right)\left\|(I-\Pi_c)e\right\|_A^2 \leq \left(1 - \hat{\alpha}_g\hat{\beta}_g^{-1}\right)\left\|e\right\|_A^2,\end{aligned}$$

we have $\hat{\delta} \geq \hat{\alpha}_g\hat{\beta}_g^{-1}$. $\Longrightarrow$ More practical TG convergence theorem.

More Practical Assumptions

Smoothing effect:

$$\left\|(I-SA)e\right\|_A^2 \le \|e\|_A^2 - \bar{\alpha}_g\, g(e), \quad \forall\, e \in V, \tag{65}$$

Approximability of coarse space:

- Strong approximation assumption (SAA):

$$\exists\, \bar{\beta}_{g,\,\mathrm{s}}, \text{ such that } \left\|(I-\Pi_c)e\right\|_A^2 \le \bar{\beta}_{g,\,\mathrm{s}}\, g(e), \quad \forall\, e \in V. \tag{66}$$

- Weak approximation assumption (WAA):

$$\exists\, \bar{\beta}_{g,\,\mathrm{w}}, \text{ such that } \left\|(I-\Pi_c)e\right\|_A^2 \le \bar{\beta}_{g,\,\mathrm{w}}\, g\big((I-\Pi_c)e\big), \quad \forall\, e \in V. \tag{67}$$

Theorem (Convergence estimate of two-level AMG)

If both (65) and (66) (or its weaker version (67)) hold, then V(0, 1) two-grid method satisfies

$$\|E_{\mathrm{TG}}\|_A^2 \le 1 - \bar{\alpha}_g\, \bar{\beta}_g^{-1}.$$

Typical Choices of g

Q: How to choose the function $g(e)$? How to enforce convergence conditions on the prolongation P?

- A trivial choice $g(e) := \|(I - \Pi_c)e\|_A^2$. Then we have $\hat{\alpha}_g = \hat{\delta}$ and $\hat{\beta}_g \equiv 1$.
- Another possibility is $g(e) := \|e\|_{AD^{-1}A}^2$ [Ruge and Stüben 1987]. SAA becomes

$$\inf_{e_c \in V_c} \|e - Pe_c\|_A^2 \leq \bar{\beta}_{\mathrm{s}} \|e\|_{AD^{-1}A}^2, \quad \forall\, e \in V. \tag{68}$$

- Noticing $\Pi_c = P(P^TAP)^{-1}P^TA$ and let $e_c \in V_c$, we have

$$\left\|(I - \Pi_c)e\right\|_A^2 \leq \left\|(I - \Pi_c)e\right\|_{AD^{-1}A} \left\|(I - \Pi_c)e - Pe_c\right\|_D.$$

- If we assume, instead of (68), that

$$\inf_{e_c \in V_c} \|e - Pe_c\|_D^2 \leq \bar{\beta}_{\mathrm{w}} \|e\|_A^2, \quad \forall\, e \in V, \tag{69}$$

then WAA holds:

$$\left\|(I - \Pi_c)e\right\|_A^2 \leq \left\|(I - \Pi_c)e\right\|_{AD^{-1}A} \left\|(I - \Pi_c)e - Pe_c\right\|_D \leq \bar{\beta}_{\mathrm{w}}^{\frac{1}{2}} \left\|(I - \Pi_c)e\right\|_{AD^{-1}A} \left\|(I - \Pi_c)e\right\|_A.$$

Defining Coarse Space

Reminder: The desirable coarse space should well approximate the lower end of the spectrum of $\overline{S}A$.

- However, it is very difficult, if not impossible, to obtain small eigen-pairs of $\overline{S}A$.
- The standard local relaxation methods satisfy that

$$\rho_A^{-1}\big(v,v\big)_A \lesssim \big(\overline{S}Av,v\big)_A \lesssim \big(v,v\big)_A.$$

It motivates the following definition of the algebraic smooth vector:

Definition (Algebraic smoothness)

Let $\varepsilon \in (0,1)$ be a small parameter. If $e \in V$ satisfies

$$\big(\overline{S}Ae, e\big)_A \le \varepsilon\big(e,e\big)_A, \quad \text{i.e. } \big(\overline{S}Ae, Ae\big) \le \varepsilon\big\|e\big\|_A^2,$$

then e is algebraically ε-smooth (or the ε-algebraic low-frequency) with respect to A.

Remarks on Algebraic Smoothness

Algebraic smooth error:

- For algebraic smooth error component e, we have

$$\left((I-\overline{S}A)e, e\right)_A \geq (1-\varepsilon)\left(e, e\right)_A \quad \Longrightarrow \quad \frac{\left\|(I-SA)e\right\|_A^2}{\left\|e\right\|_A^2} \geq 1-\varepsilon.$$

- This suggests that the smoother is not effective for e.

An alternative definition:

- Since $\overline{S}$ is SPD, the algebraically smooth vectors satisfy that

$$\left\|e\right\|_A^2 = (\overline{S}^{\frac{1}{2}}Ae, \overline{S}^{-\frac{1}{2}}e) \leq \left(\overline{S}Ae, Ae\right)^{1/2}\left(\overline{S}^{-1}e, e\right)^{1/2} \leq \varepsilon^{1/2}\left\|e\right\|_A\left(\overline{S}^{-1}e, e\right)^{1/2}.$$

- We can derive the following characterisation of algebraically smooth (low-frequency) vectors:

$$\left\|e\right\|_A^2 \leq \varepsilon\left\|e\right\|_{\overline{S}^{-1}}^2.$$

Key: Smooth error components in algebraic sense is whatever left after the smoothing steps.

Where To Find Smooth Error

- FE system of the Poisson's equation with Neumann boundary condition:

$$\frac{1}{h}\begin{pmatrix} 1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & \ddots & -1 \\ & & & -1 & 1 \end{pmatrix}\begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{N-1} \\ u_N \end{pmatrix} = h\begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_{N-1} \\ f_N \end{pmatrix}.$$

- FE discretization of the second-order elliptic equation with mixed boundary condition:

$$(Au, v) = \sum_{\substack{(i,j)\in \mathsf{E} \\ i<j}} -a_{i,j}(u_i - u_j)(v_i - v_j), \quad \text{if } \ u_j = v_j = 0, \ \forall\, x_j \in \Gamma_D. \tag{70}$$

- According to (70), the algebraically smooth error e satisfies that

$$\sum_{i<j} -a_{i,j}(e_i - e_j)^2 = (Ae, e) \le \varepsilon \|e\|_{S^{-1}}^2 \ll 1. \tag{71}$$

$\Longrightarrow$ Smooth error varies slowly in the direction of relatively "large" (negative) $a_{i,j}$.

Strongly Coupled Variables

The above observation motivates strongly n-coupled variables:

- Let $\theta_{\text{str}} \in (0,1)$ be a given real number, usually called relative strength parameter.
- If a pair of indices (i,j) satisfies that

  $$-a_{i,j} \geqslant \theta_{\text{str}} \Big| \min_k a_{i,k} \Big|,$$

  then we say that the variable i is strongly n-coupled to the variable j.

Remarks on strongly coupled variables:

- This definition considers the i-th row of A and (i,j) and (j,i) are two different pairs.
- This definition can be generalized to strongly coupled variables by considering all coupling.
- S_j denotes the set of indices which affects j and S_j^T denotes the ones which are affected by j:

$$\boxed{\mathsf{S}_j := \big\{ i \in \mathsf{N}_j : \ i \ \text{ strongly coupled to } \ j \big\} \quad \text{and} \quad \mathsf{S}_j^T := \big\{ i \in \mathsf{V} : \ j \in \mathsf{S}_i \big\}.}$$

Long-Distance Coupled Variables

In many cases, only considering direct neighbors for strong coupling is not enough! In order to reduce complexity, we can consider longer distance neighbors.

- After finding the strongly coupled variables, we can filter the coefficient matrix to obtain a filtered matrix A_{S} by removing non-strongly coupled connections.

- The above definition of strongly coupled variables applies to the direct connections; sometimes, we also need to consider indirect (i.e., long-range) connections.

- A typical example is the AMG method with aggressive coarsening.

- A variable i is said strongly coupled to another variable j along a path of length ℓ if there exists a sequence of edges
$$\big\{(i,\, k_1),\, (k_1, k_2),\, \ldots,\, (k_{\ell-2}, k_{\ell-1}),\, (k_{\ell-1}, j)\big\} \subseteq \mathsf{E}$$
such that $k_{l+1} \in \mathsf{S}_{k_l}$ for $l = 1, 2, \ldots, \ell - 2$.

- If there exist a path of length ℓ such that i is strongly coupled to j, then we say that i is ℓ-strongly coupled to j and denoted by $j \in \mathsf{S}_i^\ell$.

A Simple Example: 2D Five-Point Stencil

Consider five-point stencil finite difference on the mesh given in the following figure (left). Consider the vertex at the center (point 13). Then

$$\mathsf{S}_{13} = \{12, 8, 14, 18\} \quad \text{and} \quad \mathsf{S}^2_{13} = \{12, 8, 14, 18, 11, 3, 15, 23, 7, 9, 19, 17\}.$$

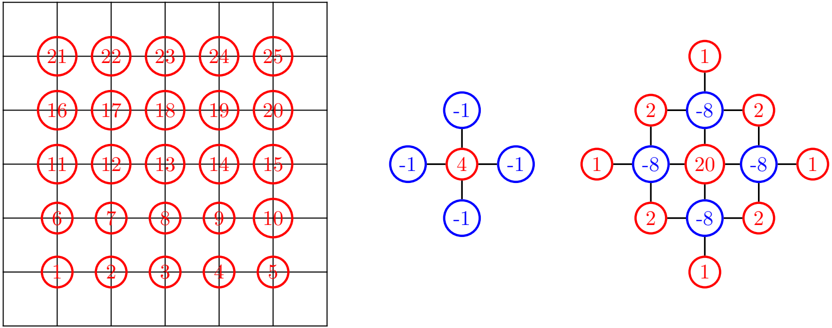
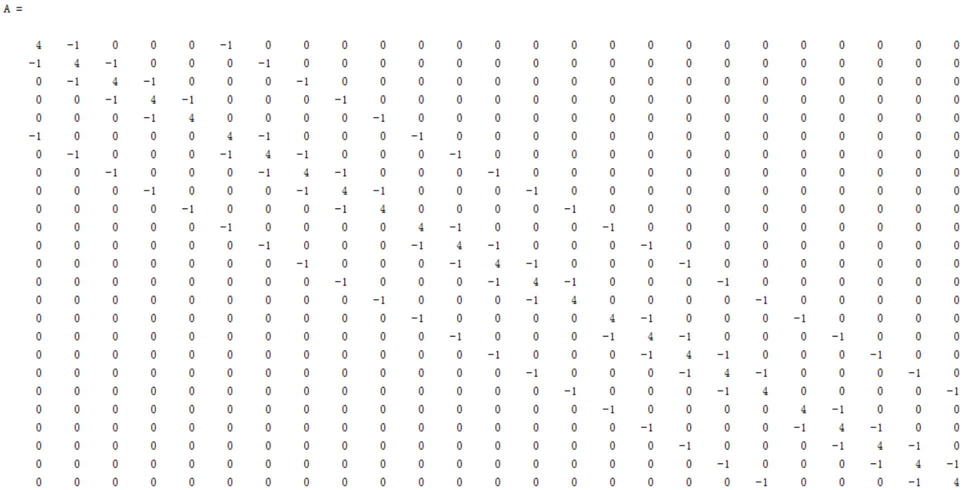


Figure: Finite difference grid (left), strong connections and weights in A of vertex 13 (middle), and 2-strong connections and weights in A^2 of 13 (right).

Example: Stiffness Matrix

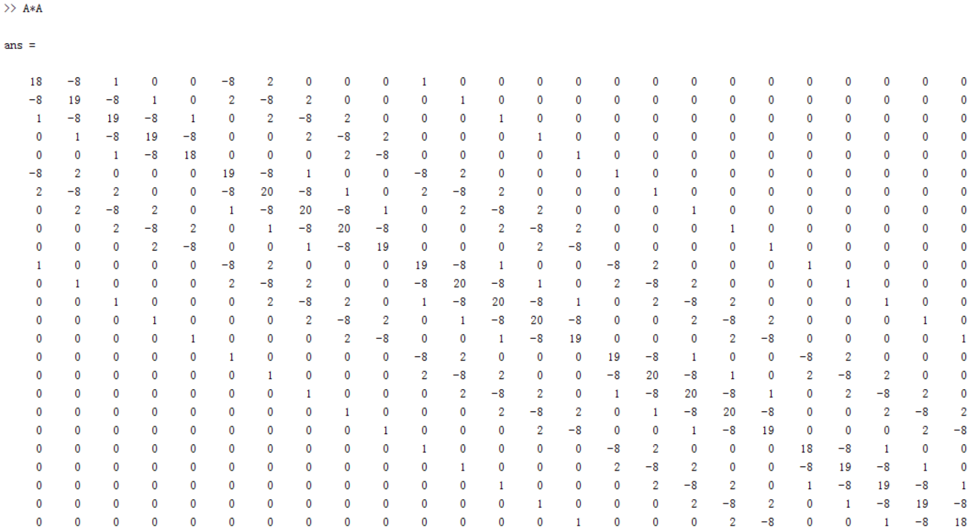


```
A =

    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0    0   -1    4    0    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0
   -1    0    0    0    0    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0    0
    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0    0
    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0    0
    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0    0    0
    0    0    0    0   -1    0    0    0   -1    4    0    0    0    0   -1    0    0    0    0    0    0    0    0    0    0
    0    0    0    0    0   -1    0    0    0    0    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0    0
    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0    0
    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0    0
    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0    0    0    0
    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4    0    0    0    0   -1    0    0    0    0    0
    0    0    0    0    0    0    0    0    0    0   -1    0    0    0    0    4   -1    0    0    0   -1    0    0    0    0
    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0    0
    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0    0   -1    0
    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4    0    0    0    0   -1
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0    0    4   -1    0    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0    0
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1    0
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4   -1
    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0    0   -1    0    0    0   -1    4
```

Figure: The stiffness matrix A for five-point stencil finite difference scheme on the above mesh.

Example: Stiffness Matrix^2



```
>> A*A

ans =

    18    -8     1     0     0    -8     2     0     0     0     1     0     0     0     0     0     0     0     0     0     0     0     0     0     0
    -8    19    -8     1     0     2    -8     2     0     0     0     1     0     0     0     0     0     0     0     0     0     0     0     0     0
     1    -8    19    -8     1     0     2    -8     2     0     0     0     1     0     0     0     0     0     0     0     0     0     0     0     0
     0     1    -8    19    -8     0     0     2    -8     2     0     0     0     1     0     0     0     0     0     0     0     0     0     0     0
     0     0     1    -8    18     0     0     0     2    -8     0     0     0     0     1     0     0     0     0     0     0     0     0     0     0
    -8     2     0     0     0    19    -8     1     0     0    -8     2     0     0     0     1     0     0     0     0     0     0     0     0     0
     2    -8     2     0     0    -8    20    -8     1     0     2    -8     2     0     0     0     1     0     0     0     0     0     0     0     0
     0     2    -8     2     0     1    -8    20    -8     1     0     2    -8     2     0     0     0     1     0     0     0     0     0     0     0
     0     0     2    -8     2     0     1    -8    20    -8     0     0     2    -8     2     0     0     0     1     0     0     0     0     0     0
     0     0     0     2    -8     0     0     1    -8    19     0     0     0     2    -8     0     0     0     0     1     0     0     0     0     0
     1     0     0     0     0    -8     2     0     0     0    19    -8     1     0     0    -8     2     0     0     0     1     0     0     0     0
     0     1     0     0     0     2    -8     2     0     0    -8    20    -8     1     0     2    -8     2     0     0     0     1     0     0     0
     0     0     1     0     0     0     2    -8     2     0     1    -8    20    -8     1     0     2    -8     2     0     0     0     1     0     0
     0     0     0     1     0     0     0     2    -8     2     0     1    -8    20    -8     0     0     2    -8     2     0     0     0     1     0
     0     0     0     0     1     0     0     0     2    -8     0     0     1    -8    19     0     0     0     2    -8     0     0     0     0     1
     0     0     0     0     0     1     0     0     0     0    -8     2     0     0     0    19    -8     1     0     0    -8     2     0     0     0
     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     0    -8    20    -8     1     0     2    -8     2     0     0
     0     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     1    -8    20    -8     1     0     2    -8     2     0
     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     1    -8    20    -8     0     0     2    -8     2
     0     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     0     0     1    -8    19     0     0     0     2    -8
     0     0     0     0     0     0     0     0     0     0     1     0     0     0     0    -8     2     0     0     0    18    -8     1     0     0
     0     0     0     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     0    -8    19    -8     1     0
     0     0     0     0     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     1    -8    19    -8     1
     0     0     0     0     0     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     2     0     1    -8    19    -8
     0     0     0     0     0     0     0     0     0     0     0     0     0     0     1     0     0     0     2    -8     0     0     1    -8    18
```

Figure: The stiffness matrix A^2 for five-point stencil finite difference scheme on the above mesh.

Convergence Estimate on Space Decomposition

Definition (Algebraic high-frequency)

Let $\zeta \in (0,1]$. If $e \in V$ satisfies that

$$\|e\|_A^2 \geq \zeta \|e\|_{S^{-1}}^2,$$

then e is called the ζ-algebraic high-frequency vector with respect to A.

Theorem (Convergence estimate based on space decomposition)

Let $V_c \subset V$ be the coarse space and $V_{\rm hf}$ consist of ζ-algebraic high frequencies. For any $v \in V$, there exist $v_c \in V_c$ and $v_{\rm hf} \in V_{\rm hf}$ such that $v = Pv_c + v_{\rm hf}$ and $\|v_{\rm hf}\|_A^2 \leq \beta \|v\|_A^2$. Then the resulting two-level AMG satisfies

$$\|E_{\rm TG}\|_A \leq 1 - \zeta\beta^{-1}.$$

Sketch of the proof:

Notice that we have the following estimate

$$\inf_{w_c \in V_c} \|v - Pw_c\|_{S^{-1}}^2 \leq \|v_{\rm hf}\|_{S^{-1}}^2 \leq \frac{1}{\zeta}\|v_{\rm hf}\|_A^2 \leq \frac{\beta}{\zeta}\|v\|_A^2.$$

Constructing Prolongation and Restriction

Iteration matrices for CGC

Coarse-grid correction matrix: $u' = u + P(RAP)^{-1}R(f - Au)$

Error propagation matrix: $e' = \big(I - P(RAP)^{-1}RA\big)e$

Residual propagation matrix: $r' = \big(I - AP(RAP)^{-1}R\big)r$

Remark: If $e \in \mathrm{range}(P)$, i.e., $e = Pv_c$ for some v_c, then we have

$$e' = e - P(RAP)^{-1}RAe = e - P(RAP)^{-1}RAPv_c = e - Pv_c \equiv 0.$$

$\Longrightarrow$ Exact CGC is exact on $\mathrm{range}(P)$ $\Longrightarrow$ Find P such that (smooth) error sits in $\mathrm{range}(P)$!

Prolongation and restriction for AMG

$$A = \begin{pmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{pmatrix}, \quad P := \begin{pmatrix} W \\ I \end{pmatrix}, \quad \text{and} \quad R := \begin{pmatrix} Z & I \end{pmatrix}.$$

Q: How to find good and practical prolongation operator P and restriction operator R accordingly?

Natural Prolongation and Restriction

Iteration matrices for F-relaxation (keep C-variables)

$$B = \begin{pmatrix} S_f & 0 \\ 0 & 0 \end{pmatrix} \implies e' = \begin{pmatrix} I - S_f A_{ff} & -S_f A_{fc} \\ 0 & I \end{pmatrix} e \implies r' = \begin{pmatrix} I - A_{ff} S_f & 0 \\ -A_{cf} S_f & I \end{pmatrix} r.$$

If A_{ff} is well-conditioned, smoother S_f should work well for the fine error, i.e.

$$r_f = A_{ff} e_f + A_{fc} e_c \approx 0 \quad \implies \quad e_f \approx -A_{ff}^{-1} A_{fc} e_c.$$

> Remark: This gives an ideal interpolation [Falgout, Vassilevski 2004]:
>
> $$\text{If } S_f = A_{ff}^{-1}, \quad P_* := \begin{pmatrix} -A_{ff}^{-1} A_{fc} \\ I \end{pmatrix} \text{ represents fine error exactly } (e' = P_* e_c).$$

How to choose restriction R for prolongation P_*? Since $RAP_* \equiv A_{cc} - A_{cf} A_{ff}^{-1} A_{fc}$, CGC results in

$$r'_f = r_f, \quad r'_c = r_c - (A_{cc} - A_{cf} A_{ff}^{-1} A_{fc})(RAP_*)^{-1}(Z r_f + r_c) \equiv -Z r_f.$$

$\implies$ Z should be set to zero! $\implies$ R is an injector.

Motivating Good Interpolations

- Let $Q \in \mathbb{R}^{N\times N}$ be a projection onto $\mathrm{range}(P)$: $Q = PT$, where $T \in \mathbb{R}^{N_c\times N}$ and $TP = I_c$.
- If $T_{\mathrm{s}} := (P^TAP)^{-1}P^TA$, then it is easy to see that $Q_{\mathrm{s}} = PT_{\mathrm{s}} = \Pi_c$.
- A simplified choice $T_{\mathrm{w}} := (P^TDP)^{-1}P^TD$ can be used.
- For any vector $0 \neq e \in V$, we can assume (strong and weak approximability) that

$$\inf_{e_c\in V_c} \frac{\left\|e - Pe_c\right\|_A^2}{\|e\|_{AD^{-1}A}^2} \le \frac{\left\|e - Qe\right\|_A^2}{\|e\|_{AD^{-1}A}^2} \le \bar{\beta}_{\mathrm{s}} \qquad \text{or} \qquad \inf_{e_c\in V_c} \frac{\left\|e - Pe_c\right\|_D^2}{\|e\|_A^2} \le \frac{\left\|e - Qe\right\|_D^2}{\|e\|_A^2} \le \bar{\beta}_{\mathrm{w}}.$$

Assumption (Weak approximability)

$$\left\|(I - PT)v\right\|_D \le \beta\|v\|_A, \quad \forall v \in V.$$

- Above inequalities give bounds for constructing P such that the TG method converges well.
- The error becomes pretty “smooth” after a few relaxation steps.
- An appropriate coarse space should approximate a smooth vector v accurately.

Ideal Interpolation

- According to the above WAA inequality, we want to minimize

$$\mu_D(Q,e) := \frac{\|(I-Q)e\|_D^2}{\|e\|_A^2}, \quad \forall\, e \neq 0.$$

D can be generalized to an SPD matrix X and

$$\mu_X(Q,e) := \frac{\|(I-Q)e\|_X^2}{\|e\|_A^2}, \quad \forall\, e \neq 0.$$

- Assume that $\mu_X(Q,e) \le \kappa$. Then

$$\inf_{e_c \in V_c} \frac{\left\|e - Pe_c\right\|_X^2}{\|e\|_A^2} \le \frac{\left\|e - Qe\right\|_X^2}{\|e\|_A^2} = \mu_X(Q,e) \le \kappa. \tag{72}$$

We now minimize

$$\inf_P \sup_{e \neq 0} \mu_X(PT, e)$$

to find the “best possible” prolongation P (ideal interpolation) [Falgout, Vassilevski 2004].

Properties of Ideal Interpolation

Lemma (Falgout, Vassilevski 2004)

Assume $Q = PT$ and $TP = I_{N_c}$. So Q is a projection onto range(P). Then $\|E_{TG}\|_A^2 \leq 1 - \frac{1}{\mu}$, where

$$\mu := \sup_{e \neq 0} \frac{e^T (I-Q)^T \overline{S}^{-1} (I-Q) e}{e^T A e} \geq 1.$$

The magnitude of μ reflects the approximation property of the AMG method $\Longrightarrow$ Make it small.

Theorem (Falgout, Vassilevski 2004)

Given a matrix $G \in \mathbb{R}^{N \times (N-N_c)}$ and rank$(G) = N - N_c$ such that $TG = 0$. Let $X \in \mathbb{R}^{N \times N}$ be SPD. Define

$$\mu_X(Q, e) := \frac{e^T (I-Q)^T X (I-Q) e}{e^T A e}.$$

We have

$$\mu_X^* := \min_P \max_{e \neq 0} \mu_X(Q, e) = \frac{1}{\lambda_{\min}\big((G^T X G)^{-1} G^T A G\big)}$$

and the unique minimizer is P_* that satisfies $P_*^T A G = 0$.

A Simple Counterexample

This theory is misleading in several ways:

- The ideal interpolation is unique and not sparse in general.
- It is not practical to use ideal interpolation.
- Q: What role does G play and where it enters in an algorithm?

Example [Xuefeng Xu, Z. 2018]. Let $N = 3$, $N_c = 1$. Assume that

$$A = \begin{pmatrix} 2 & -1 & 1 \\ -1 & 2 & -1 \\ 1 & -1 & 2 \end{pmatrix}, \quad P_o = R_o^T = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad G = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}, \quad X = D.$$

Applying the theory by Falgout and Vassilevski, one can obtain that

$$\mu_X^* = 2 \quad \text{and} \quad P_* = \begin{pmatrix} -\frac{1}{3} \\ \frac{1}{3} \\ 1 \end{pmatrix} \quad \Longrightarrow \quad P_*^T A G = 0.$$

However, we have $\max_{e \neq 0} \mu_X(P_o R_o, e) = 2 = \mu_X^*$. $\Longrightarrow$ P_o is also ideal, yet sparse!

Theory of Ideal Interpolation

Definition: The set of ideal interpolations is $\mathbb{P}_* := \left\{ P \,:\, \mu_X^* = \min_P \max_{e\neq 0} \mu_X(PR, e) \right\}$

Let $B := A - AP(P^TAP)^{-1}P^TA$, $Y_X := (G^TXG)^{-\frac{1}{2}}G^TYG(G^TXG)^{-\frac{1}{2}}$. Then

$$\mathbb{P}_* = \{P \,:\, \lambda_{\min}(A_X) = \lambda_{\min}(B_X)\}.$$

Theorem (Ideal Interpolation Set [Xuefeng Xu, Z. 2018; Xuefeng Xu, Thesis 2019])

We have $\mathbb{P}_0 \subseteq \mathbb{P}_* = \mathbb{P}_2 \subseteq \mathbb{P}_1$. If $P_* \in \mathbb{P}_0$, then $P_* = A^{-1}R^T(RA^{-1}R^T)^{-1}$. Here

$$\begin{aligned}
\mathbb{P}_0 &:= \{P \in \mathbb{R}^{N\times N_c} \,:\, P^TAG = 0\},\\
\mathbb{P}_1 &:= \{P \in \mathbb{R}^{N\times N_c} \,:\, \text{null}(P^TAG) \cap V_A \neq \emptyset\},\\
\mathbb{P}_2 &:= \{P \in \mathbb{R}^{N\times N_c} \,:\, \text{null}(P^TAG) \cap V_B \neq \emptyset\},\\
V_A &:= \{v \in \mathbb{R}^{N-N_c}\backslash\{0\} \,:\, G^TAGv = \lambda_{\min}(A_X)G^TXGv\},\\
V_B &:= \{v \in \mathbb{R}^{N-N_c}\backslash\{0\} \,:\, G^TBGv = \lambda_{\min}(B_X)G^TXGv\}.
\end{aligned}$$

Unfortunately, the ideal interpolation P_* is usually expensive to compute! Q: How to simplify it?

Simplified Assumption on P

Q: How to give the assumption more practical?

- By the weak approximation assumption and choosing $D := \|A\| I$, we have
$$\|A\|^{1/2}\|v - Q\,v\| \le \beta\|v\|_A.$$
- If v is in the near-null space of A, i.e., $\|v\|_A \approx 0$, then $Q\,v \approx v$. $\Longrightarrow$ We can employ a simplified assumption — Constant-preserving!

Remark (Cannonical prolongation operator)

Let $1_N := (1, 1, \ldots, 1)^T$. Since the basis functions form a partition of unity, it follows that

$$(\phi_1, \cdots, \phi_N)\,1_N = \sum_{i=1}^{N} \phi_i = 1 = \sum_{l=1}^{N_c} \phi_l^c = (\phi_1^c, \cdots, \phi_{N_c}^c) 1_{N_c} = (\phi_1, \cdots, \phi_N) P\,1_{N_c}.$$

Hence we have that the prolongation matrix preserves constant away from the boundary $P\,1_{N_c} = 1_N$.

Assumption (Constant preserving)

$$P\,1_{N_c} = 1_N.$$

Direct Interpolation

It is “reasonable” [Stüben 2000; 2001] to assume, for smooth error components, that

$$\boxed{A_{ff}e_f + A_{fc}e_c \approx 0 \qquad \text{or} \qquad \textstyle\sum_{j=1}^{N} a_{ij}e_j \approx 0, \quad i \in \mathsf{F}}$$

Hence we have

$$a_{ii}e_i + \sum_{j\in \mathsf{N}_i} a_{ij}e_j = 0, \quad i \in \mathsf{F}.$$

This is not an interpolation though (j could be in F too)! We modify it as

$$a_{ii}e_i + \alpha_i \sum_{j\in \mathsf{N}_i\cap\mathsf{C}} a_{ij}e_j = 0, \quad \alpha_i := \frac{\sum_{k\in\mathsf{N}_i} a_{ik}}{\sum_{k\in\mathsf{N}_i\cap\mathsf{C}} a_{ik}}, \quad i \in \mathsf{F}.$$

If $A_{i,:}$ has zero row-sum, we have $\alpha_i := \frac{-a_{ii}}{\sum_{k\in\mathsf{N}_i\cap\mathsf{C}} a_{ik}}$. So we obtain

$$e_i = \sum_{j\in\mathsf{N}_i\cap\mathsf{C}} w_{ij}e_j, \quad w_{ij} = \frac{a_{ij}}{\sum_{k\in\mathsf{N}_i\cap\mathsf{C}} a_{ik}} \quad \Longrightarrow \quad W = \mathrm{diag}(A_{fc}\mathbf{1}_c)^{-1}A_{fc}.$$

Ideal Restriction

Now we turn to a general interpolation operator P. Any error can be written as

$$e = \begin{pmatrix} e_f \\ e_c \end{pmatrix} = \begin{pmatrix} We_c \\ e_c \end{pmatrix} + \begin{pmatrix} \epsilon_f \\ 0 \end{pmatrix} \quad \sim \quad \mathrm{range}(P) + \text{perturbation}$$

Because CGC is exact on $\mathrm{range}(P)$, the new error can be given as

$$e' = \left(I - P(RAP)^{-1}RA\right) \begin{pmatrix} \epsilon_f \\ 0 \end{pmatrix}.$$

> Remark: We can construct R_*, for all ϵ_f, such that
>
> $$0 \equiv R_* A \begin{pmatrix} \epsilon_f \\ 0 \end{pmatrix} = \begin{pmatrix} Z_* & I \end{pmatrix} \begin{pmatrix} A_{ff} & A_{fc} \\ A_{cf} & A_{cc} \end{pmatrix} \begin{pmatrix} \epsilon_f \\ 0 \end{pmatrix} \implies Z_* = -A_{cf} A_{ff}^{-1}.$$
>
> This yields the so-called ideal restriction $R_* := (Z_* \;\; I)$.

Apparently, in this case, we can show that

$$e'_f = \epsilon_f = e_f - We_c \quad \text{and} \quad e'_c = 0.$$

Approximated Ideal Restriction

Local Approximation of Ideal Restriction: Notice that the ideal restriction satisfies

$$R_* A = \begin{pmatrix} Z_* A_{ff} + A_{cf} & Z_* A_{fc} + A_{cc} \end{pmatrix} = \begin{pmatrix} 0 & S_c \end{pmatrix}.$$

Motivated by this relation, we can construct Z locally: For any $i \in \mathsf{C}$, let

$$\sum_{k \in \mathsf{N}_i \cap \mathsf{F}} a_{kj} z_{ik} = -a_{ij}, \quad \forall j \in \mathsf{N}_i \cap \mathsf{F}.$$

That is to say,

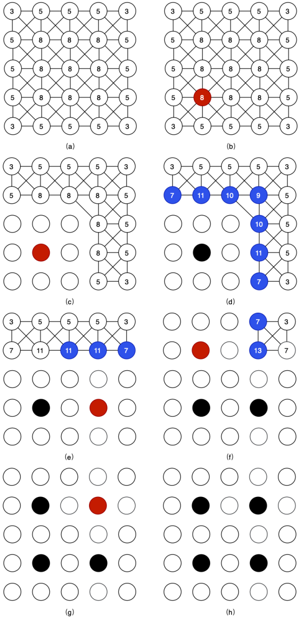
$$(RA)_{ij} = 0, \quad \forall i \in \mathsf{C}, j \in \mathsf{N}_i \cap \mathsf{F}.$$

This is the main motivation of the so-called ℓAIR scheme [Manteuffel, Ruge, Southworth 2018].

> Remark: With appropriate C/F splitting, A_{ff} is diagonally dominant and the local problems are usually nonsingular. When they are singular or nearly singular, a minimal norm solution in the least-square sense can be used.

[Brezina et al. 2010; Manteuffel, Southworth 2019; Notay 2020]

Classical Coarsening Methods



- Strong n-coupling: $-a_{i,j} \geqslant \theta_{\text{str}} \left| \min_k a_{i,k} \right|$
  - $A = (a_{i,j}) \in \mathbb{R}^{N \times N}$ is an M-matrix
  - $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ is the corresponding graph of A
  - $\mathsf{S}_i := \{ j \in \mathsf{N}_i : \ j \text{ strongly coupled to } i \}$
  - $\mathsf{S}_i^T := \{ j \in \mathsf{V} : \ i \in \mathsf{S}_j \}$, set affected by i
- C/F splitting algorithm
  - Split variables into F and C
  - Need a measure of importance λ_i for variable i

```
$\mathsf{U} \leftarrow \mathsf{V}$, $\mathsf{C} \leftarrow \emptyset$, $\mathsf{F} \leftarrow \emptyset$;
while $\mathsf{U} \neq \emptyset$
    $\lambda_i \leftarrow 2|\mathsf{S}_i^T \bigcap \mathsf{F}| + |\mathsf{S}_i^T \bigcap \mathsf{U}|$, $i \in \mathsf{U}$;
    $k \leftarrow \arg\max\{\lambda_i, i \in \mathsf{U}\}$;
    $\mathsf{C} \leftarrow \mathsf{C} \bigcup \{k\}$, $\mathsf{U} \leftarrow \mathsf{U} \backslash \{k\}$;
    $\mathsf{F} \leftarrow \mathsf{F} \bigcup \mathsf{S}_k^T$, $\mathsf{U} \leftarrow \mathsf{U} \backslash \mathsf{S}_k^T$;
end
```

Example: Classical Coarsening on Anisotropic Problem

Example: Consider an anisotropic diffusion example on a unit square:

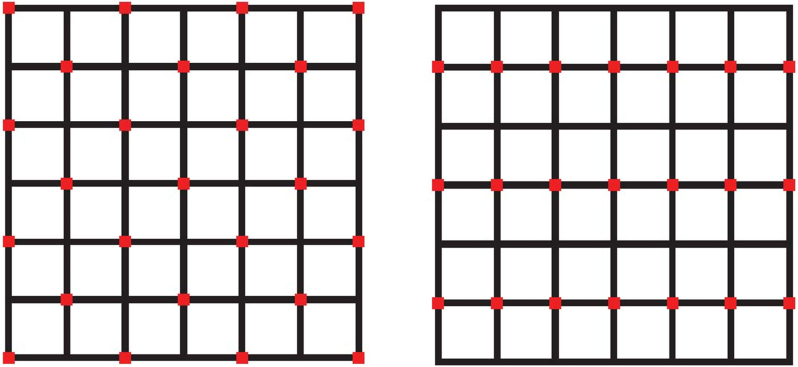
$$-\epsilon u_{xx} - u_{yy} = 0 \quad (\epsilon > 0).$$

- We have $\epsilon \|u_{xx}\| \approx \|u_{yy}\|$: The solution is smooth in y-direction; but rough in x-direction.
- We consider the five-point stencil. The difference equation at the node (x_i, y_j) is

$$-\epsilon \frac{2u_{i,j} - u_{i+1,j} - u_{i-1,j}}{h_x^2} - \frac{2u_{i,j} - u_{i,j-1} - u_{i,j+1}}{h_y^2} = 0.$$

- How to solve such an equation efficiently?
  1. Employ an anisotropic grid
  2. Use GMG with line smoother
  3. Use GMG with semi-coarsening
  4. Use AMG

Example: Results After Coarsening



- C/F splitting for the 2D elliptic problem with $\epsilon = 1$ (left) and $\epsilon \ll 1$ (right), where the red points are C-variables and the black points are F-variables.

- If $\frac{\epsilon}{h_x^2} \ll \frac{1}{h_y^2}$, then $u_{i,j}$ depends on $u_{i,j+1}$ and $u_{i,j-1}$ mainly. If we apply the C/F procedure, the coarsening will take place indeed in one direction only (semi-coarsening).

Complexity of AMG Methods

- Considering convergence alone is not meaningful in the algebraic multigrid context.
- An equally important aspect is the complexity of the coarser level matrices in AMG.
- This is because the computational complexity of each AMG cycle could be prohibitively large.
- Usually, the coarser matrices A_{l-1} becomes more dense than A_l.
- When constructing the prolongation P, we must control the sparsity of the coarse level matrices.
- We define a measurement of sparsity, i.e., the operator complexity

  $$C_A := \frac{\sum_{l=0}^{L} \mathrm{nnz}(A_l)}{\mathrm{nnz}(A)},$$

  where $\mathrm{nnz}(\cdot)$ is the number of nonzeros of a matrix.
- Apparently, $C_A \geq 1$ is always true and $C_A = 1$ corresponds to the one-level case.
- We would like to make C_A as close to 1 as possible while keeping good convergence performance.

Reducing Complexity of Classical AMG

Reducing cost $\Longrightarrow$ Reducing complexity $\Longrightarrow$ Make the coarse-level problems more sparse!

1. Truncated Prolongation:

   Some times (e.g. Jacobi interpolation), many interpolatory variables are used for interpolation. But weights for variables “far away” are usually relatively small. We can drop relatively small weights and rescale the remaining weights to keep the sum unchanged. $\Longrightarrow$ Reduce complexity of setup.

2. Aggressive Coarsening:

   Some times (e.g. small stencils), standard C/F splitting leads to high complexity. Instead of direct neighbors, we can consider long-range strong connections. $\Longrightarrow$ Increasing number of “neighbors”, reduce number of C-points.

3. Approximate Coarse Problem:

   We can specify a sparsity pattern for the coarse problem and then construct a method accordingly. $\Longrightarrow$ But this might cause serious convergence problem. $\Longrightarrow$ Requires great caution. Need theory!

[Stüben 2001; Falgout, Schroder 2014; ...]

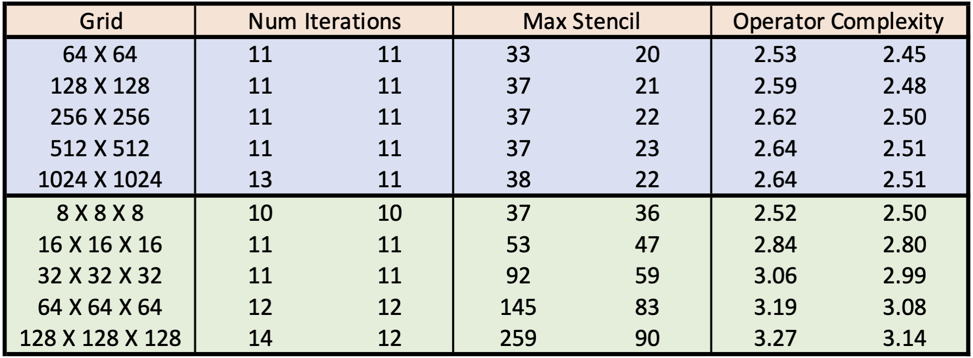
Example: Aggressive Coarsening

Coarsening method	Standard	Aggressive
Operator complexity	2.889	1.606
Setup time (sec)	1.536	1.036
Number of iterations	6	38
Solve time (sec)	0.791	3.293
Time per iteration (sec)	0.132	0.087

Table: Solving 2D five-point stencil of the Poisson's equation with 1 million DOF using different coarsening methods in the classical AMG method (stopping criteria for PCG is the relative residual smaller than 10^{-6}).

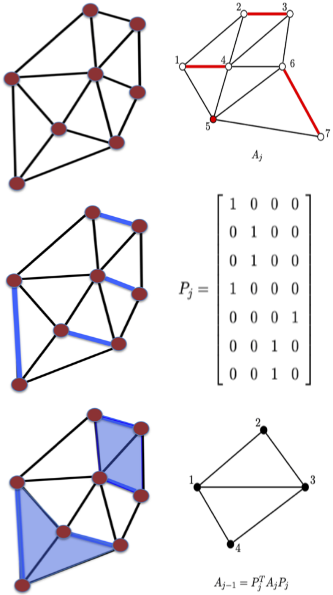
Example: Non-Galerkin AMG

- Galerkin operators: not easy to keep sparsity
- Truncation directly often results in deteriorated convergence
- Non-Galerkin AMG method proposed in [Falgout and Schroder 2014]
- Numerical tests on the Poisson's equation [X. Yue, et al. Tech Report, 2020]



Grid	Num Iterations		Max Stencil		Operator Complexity	
64 X 64	11	11	33	20	2.53	2.45
128 X 128	11	11	37	21	2.59	2.48
256 X 256	11	11	37	22	2.62	2.50
512 X 512	11	11	37	23	2.64	2.51
1024 X 1024	13	11	38	22	2.64	2.51
8 X 8 X 8	10	10	37	36	2.52	2.50
16 X 16 X 16	11	11	53	47	2.84	2.80
32 X 32 X 32	11	11	92	59	3.06	2.99
64 X 64 X 64	12	12	145	83	3.19	3.08
128 X 128 X 128	14	12	259	90	3.27	3.14

Aggregation-Based Coarsening Methods



$$P_j = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$A_{j-1} = P_j^T A_j P_j$

- Greedy aggregation algorithms

```
$N_c \leftarrow 0$, $U \leftarrow V$;
for $i \in U$
    if $N_i \subseteq U$
        $N_c \leftarrow N_c + 1$;
        $C_{N_c} \leftarrow \{i\} \bigcup N_i$, $U \leftarrow U \backslash C_{N_c}$;
    end
end
```

```
$U \leftarrow V$;
for $i \in U$
    $S_i \leftarrow \{j \in U$ : $j$ is strongly coupled to $i\}$;
    construct a column of prolongation $P$ based on $S_i$;
    $U \leftarrow U \backslash (\{i\} \bigcup S_i)$;
end
```

- Constructing prolongation operators:

$$(P)_{i,j} = \begin{cases} 1, & \text{if } i \in C_j \\ 0, & \text{if } i \notin C_j \end{cases} \quad \text{for } 1 \le i \le N, 1 \le j \le N_c$$

  - But simple prolongation yields nonuniform convergence!
  - Cure 1: Smooth out the piecewise constant basis functions:

$$\psi_i = (I - \omega D^{-1} A)\, \mathbf{1}_i;$$

Hence we have the partition of unity

$$\sum_i \psi_i = (I - \omega D^{-1} A) \sum_i \mathbf{1}_i = (I - \omega D^{-1} A)\, \mathbf{1} = \mathbf{1}.$$

  - Cure 2: Apply more complex multilevel iterations.

Example: Aggregation AMG

- After forming aggregates one can apply UA or SA to give prolongation.
- Apply aggregation methods to solve the 2D Poisson's equation using the five-point stencil.
- For the SA method, we use the standard V-cycle multigrid in the solve phase.
- For the UA methods, we use the AMLI-cycle or K-cycle multigrid for better convergence.

Aggregation method	SA	UA	Pairwise UA
Number of levels	5	5	7
Operator complexity	1.364	1.264	1.332
Setup time (sec)	0.557	0.171	0.277
Number of iterations	16	21	12
Solve time (sec)	1.223	1.696	1.336

Table: Solving 2D five-point stencil of the Poisson's equation with 1 million DOF using aggregation-based AMG methods (stopping criteria for PCG is the relative residual smaller than 10^{-6}).

Algebraic Convergence Analysis

Assumptions on smoothers:

$$\rho(I - S_l A_l) < 1, \quad l = 1, 2, \ldots, L$$

A-projection on level l:

$$\Pi_l := P_{l-1} A_{l-1}^{-1} P_{l-1}^T A_l$$

Convergence factor of MG for SPD problems:

- For multilevel AMG methods, we have very limited theoretical results.

> The asymptotic convergence factor estimate [McCormick 1985]
>
> $$\rho(E_{\mathrm{MG}}^L) \le 1 - \min_{1 \le l \le L} \min_{v_l \in \mathbb{R}^{n_l}} \frac{\left\| v_l \right\|_{A_l}^2 - \left\| (I - S_l A_l) v_l \right\|_{A_l}^2}{\left\| (I - \Pi_l) v_l \right\|_{A_l}^2}.$$

- We still do not have enough theoretical understanding to AMG methods.
- Usually two-level convergence results are utilized for constructing new AMG's in practice.

[Napov, Notay 2010; Maclachlan, Olson 2014; Notay 2015; Xu, Zikatanov 2017]

Convergence Estimates

Convergence theory for PTG: Assume that

$$\Delta_1 := \min\left\{1, \min_{v_c \in \mathcal{V}_c \setminus \{0\}} \frac{v_c^T B_c^{-1} v_c}{v_c^T A_c^{-1} v_c}\right\} \quad \text{and} \quad \Delta_2 := \max\left\{1, \max_{v_c \in \mathcal{V}_c \setminus \{0\}} \frac{v_c^T B_c^{-1} v_c}{v_c^T A_c^{-1} v_c}\right\},$$

where $\mathcal{V}_c := \operatorname{range}\left(P^T(I - AS)\right)$.

Theorem (Convergence rate of PTG [Xuefeng Xu, Thesis 2019; Xuefeng Xu, Z. 2022])

The convergence rate of PTG can be estimated by

$$1 - \frac{\Delta_2}{K_{\mathrm{TG}}} \le \|E_{\mathrm{PTG}}\|_A \le \max\left\{1 - \frac{\Delta_1}{K_{\mathrm{TG}}}, \Delta_2 - 1\right\}$$

Remark: It is easy to see that

$$\min\left\{1, \lambda_{\min}(B_c^{-1} A_c)\right\} \le \Delta_1 \le 1 \le \Delta_2 \le \max\left\{1, \lambda_{\max}(B_c^{-1} A_c)\right\}.$$

Hence we can immediately obtain the results [Notay 2007; Falgout, Schroder 2014]:

$$\lambda_{\min}(B_{\mathrm{PTG}}^{-1} A) \ge \lambda_{\min}(B_{\mathrm{TG}}^{-1} A)\ \min\left\{1, \lambda_{\min}(B_c^{-1} A_c)\right\},$$
$$\lambda_{\max}(B_{\mathrm{PTG}}^{-1} A) \le \lambda_{\max}(B_{\mathrm{TG}}^{-1} A)\ \max\left\{1, \lambda_{\max}(B_c^{-1} A_c)\right\}.$$

Convergence Theory: From TG to MG

- Perturbed TG analysis: If TG converges uniformly with factor $\sigma < 0.5$, then W-cycle convergence factor ρ is bounded by $\frac{\sigma}{1-\sigma}$ [Notay 2007].
- Q: When can optimality carry over to the V-cycle? Optimal TG is not enough! Need uniform boundedness of the CGC projector Π_l [Napov, Notay 2010].
- Apply the above two-sided bound for the convergence rate of inexact TG method [Xu, Z. 2022]:

Smoother type	Cycle type	Convergence factor	#Iterations
Gauss-Seidel with CF-ordering	TG	0.459	25
	W	0.552	30
	V	0.899	122
Weighted Jacobi with weight 0.5	TG	0.536	36
	W	0.640	47
	V	0.931	227
Weighted Jacobi with weight 0.7	TG	0.859	147
	W	0.866	154
	V	0.941	326