Lecture Note #14 **MTH652: Advanced Numerical Analysis**
Winter 2025 Instructor: Will Pazner

Two-Level Domain Decomposition Methods

1 Two-Level Preconditioner with Vertex Patches

We define and analyze a standard "two-level" additive domain decomposition method. Let V_h be the piecewise linear finite element space defined on a mesh $\mathcal{T}$. The preconditioner will be defined by

$$B = \sum_{j=0}^{J} A_j^{-1} Q_j \tag{1}$$

where Q_j is the projection onto V_j in the standard inner product, and A_j is the restriction of A to the subspace V_j. The definition of this preconditioner is complete once the space decomposition

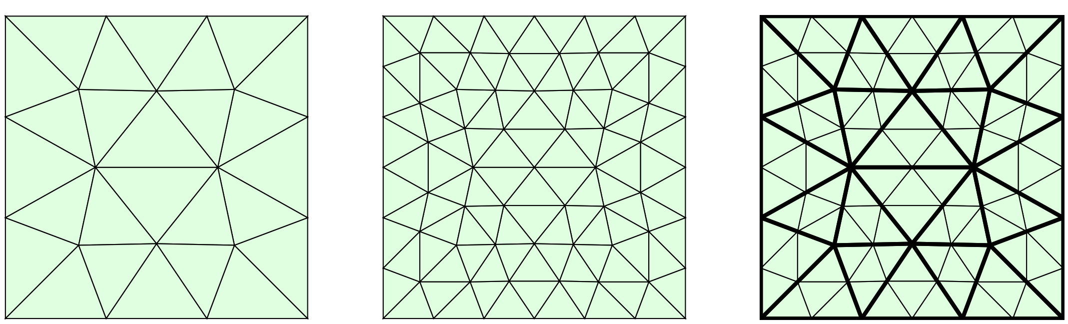
$$V_h = \sum_{j=0}^{J} V_j = V_0 + \sum_{j=1}^{J} V_j \tag{2}$$

is specified. We list the space V_0 separately on the right-hand side of (2) because it differs from the other subspaces, as we will see.

1.1 Coarse Space

We will assume that the mesh $\mathcal{T}$ has been obtained by a **refinement** of a coarse mesh $\mathcal{T}_0$. What we mean by this is that $\mathcal{T}$ can be obtained from $\mathcal{T}_0$ by adding vertices and edges (in the process creating new triangles). So every vertex in $\mathcal{T}_0$ is also a vertex in $\mathcal{T}$, and every triangle in $\mathcal{T}$ is entirely contained within a single triangle of $\mathcal{T}_0$.

Define the **coarse space** V_0 to be the piecewise linear finite element space on the coarse mesh $\mathcal{T}_0$. By the assumptions on the meshes $\mathcal{T}$ and $\mathcal{T}_0$, it follows that $V_0 \subseteq V_h$. Since elements of V_0 are piecewise linear on the coarse mesh, they are also piecewise linear on the fine mesh, and so they must also belong to V_h.



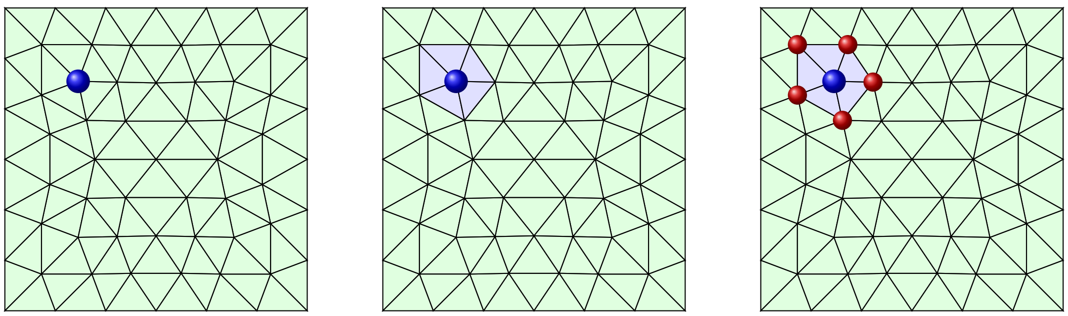
$\mathcal{T}_0$ $\mathcal{T}$ Both meshes overlaid

1.2 Vertex Patch Spaces

The remaining spaces V_j for $j > 0$ will be associated with **vertex patches**. A vertex patch is the collection of elements (triangles) that are incident to a given vertex. Let $\mathcal{T}_j$ be the vertex patch associated with vertex j. The space V_j is the space of piecewise linear functions on $\mathcal{T}_j$ that vanish on its boundary $\partial\Omega_j$. It can be seen immediately that $V_j = \operatorname{span}\{\phi_j\}$, where $\phi_j \in V_h$ is the basis function (hat function) associated with vertex j.

To verify (2) (i.e. to check that this indeed gives us the entire space V_h), we note that $V_h = \sum_{j=1}^{J} V_j$. The coarse space V_0 is not needed to span V_h, the vertex patches are enough, but the presence of V_0 will be essential for the preconditioner (1) to be effective.

Take as an example the following mesh. We will construct the vertex patch corresponding to the indicated blue vertex.



$\mathcal{T}$ $\qquad$ $\mathcal{T}_i$ $\qquad$ $\mathcal{V}_i$

Notation:

- The domain is denoted Ω and the mesh is $\mathcal{T}$
- The union of the elements incident to vertex i is denoted Ω_i
- The corresponding subset of the mesh is $\mathcal{T}_i$
- The set of all vertices in $\mathcal{T}_i$ (the red and blue vertices) is denoted $\mathcal{V}_i$

The space V_i is defined as

$$V_i = \{v_h \in V_h : v_h = 0 \text{ outside } \Omega_i\}.$$

2 Analysis

We now proceed to analyze the preconditioner. Our goal is to estimate the condition number of the preconditioned system $T = BA$. The main tool will be identity we proved in the previous lecture:

$$(T^{-1}v, v)_A = \inf_{\sum v_j = v} \sum_{j=0}^{J} (T_j^{-1} v_j, v_j)_A \tag{3}$$

where the infimum is taken over all decompositions $v = \sum_{j=0}^{J} v_j$ with $v_j \in V_j$.

In practice, the vertex patch spaces will use exact solvers $B_j = A_j^{-1}$ and so $T_j = P_j$ for $i > 0$, since these are only one-dimensional. The coarse space is potentially large, and we may use some approximation $B_0 \approx A_0^{-1}$. The general assumption is that B_j is spectrally equivalent to A_j^{-1}, so T_j restricted to V_j is well conditioned,

$$c(v_j, v_j)_A \leq (T_j v_j, v_j)_A \leq C(v_j, v_j)_A.$$

From this relation, we have that

$$c\sum_{j=0}^{J}(v_j, v_j)_A \leq \sum_{j=0}^{J}(T_j^{-1}v_j, v_j)_A \leq C\sum_{j=0}^{J}(v_j, v_j)_A,$$

and so it will suffice for us to estimate

$$\inf_{\sum v_j = v} \sum_{j=0}^{J}(v_j, v_j)_A$$

neglecting the approximation $T_i \approx P_i$.

This identity allows us to bound the maximum and minimum eigenvalues of T^{-1}, which immediately give us bounds on the maximum and minimum eigenvalues of T, and so we can estimate $\kappa(BA)$. The maximum and minimum eigenvalues of T are bounded, respectively, using two main tools:

1. **Finite Overlap**
2. **Stable Decomposition**

Finite Overlap

The **finite overlap property** is the property that

$$a(v_i, v_j) = 0 \quad \text{for all but } \textbf{finitely many} \text{ (bounded) } i, j > 0.$$

Let $i > 0$. Notice that the support of ϕ_i is the vertex patch subdomain Ω_i. Let $\mathcal{V}_i$ denote the vertices in the mesh patch $\mathcal{T}_i$. We see that $a(\phi_i, \phi_j)$ is nonzero only when $i, j \in \mathcal{V}_i$. For given i, the number of $j > 0$, $j \neq i$ such that $a(i, j) \neq 0$ is exactly the valence of the vertex i (the number of edges incident to i). Let D denote the maximum valence of any vertex in the mesh.

We use the finite overlap property to bound the maximal eigenvalue of T. This is equivalent to bounding the minimal eigenvalue of T^{-1}. So, using the identity (3) (and the following discussion), we want to provide a *lower bound* on the expression

$$\inf_{\sum v_j = v} \sum_{j=0}^{J}(v_j, v_j)_A.$$

We first separate out the terms involving v_0, since the coarse space overlaps all the vertex patch spaces. Given $v \in V_h$ and the decomposition $v = \sum_{j=0}^{J} v_j$,

$$
\begin{aligned}
(v, v)_A &= \left(\sum_{j=0}^{J} v_j, \sum_{j=0}^{J} v_j \right)_A \\
&= \left(v_0 + \sum_{j=1}^{J} v_j, v_0 + \sum_{j=1}^{J} v_j \right)_A \\
&= (v_0, v_0)_A + 2 \left(v_0, \sum_{j=1}^{J} v_j \right)_A + \left(\sum_{j=1}^{J} v_j, \sum_{j=1}^{J} v_j \right)_A \\
&\leq (v_0, v_0)_A + 2\|v_0\|_A \left\| \sum_{j=1}^{J} v_j \right\|_A + \left(\sum_{j=1}^{J} v_j, \sum_{j=1}^{J} v_j \right)_A \qquad \text{(Cauchy–Schwarz)} \\
&\leq 2(v_0, v_0)_A + 2 \left(\sum_{j=1}^{J} v_j, \sum_{j=1}^{J} v_j \right)_A \qquad \text{(Young's inequality)}
\end{aligned}
$$

We now focus on the second term in the right-hand side.

$$
\begin{aligned}
\left(\sum_{j=1}^{J} v_j, \sum_{j=1}^{J} v_j \right)_A &= \sum_{i,j=1}^{J} (v_i, v_j)_A \\
&= \sum_{i=1}^{J} \sum_{j \in \mathcal{V}_i} (v_i, v_j)_A \\
&\leq \sum_{i=1}^{J} \sum_{j \in \mathcal{V}_i} \|v_i\|_A \|v_j\|_A \\
&\leq \frac{1}{2} \sum_{i=1}^{J} \sum_{j \in \mathcal{V}_i} \left(\|v_i\|_A^2 + \|v_j\|_A^2 \right) \\
&\leq (D+1) \sum_{j=1}^{J} (v_j, v_j)_A
\end{aligned}
$$

Combined, these give

$$
(v, v)_A \leq 2(v_0, v_0)_A + 2(D+1) \sum_{j=1}^{J} (v_j, v_j)_A \leq 2(D+1) \sum_{j=0}^{J} (v_j, v_j)_A,
$$

and so

$$
\inf_{\sum v_j = v} \sum_{j=0}^{J} (v_j, v_j)_A \geq \frac{1}{2(D+1)} (v, v)_A,
$$

giving a bound of $2(D+1)$ on the largest eigenvalue of T.

Alternative Argument

Note: a better bound can be proven using an alternative argument. We want to bound the largest eigenvalue of T, which is spectrally equivalent to P. So, we try to find a constant C such that

$$a(Pu, u) \leq Ca(u, u)$$

for all $u \in V_h$. Since $P = \sum_{j=0}^{J} P_j$, we have

$$\begin{aligned}
a(Pu, u) &= a\left(\sum_{j=0}^{J} P_j u, u\right) \\
&= \sum_{j=0}^{J} a(P_j u, u) \\
&= \sum_{j=0}^{J} a(P_j u, P_j u) && \text{(definition of } P_j\text{)} \\
&= \sum_{j=0}^{J} a_j(P_j u, P_j u)
\end{aligned}$$

where

$$a_j(u, v) := (\nabla u, \nabla v)_{\Omega_j}$$

since $P_j u \in V_j$, and all $v_j \in V_j$ vanish outside Ω_j.
Note further that, for any j,

$$\begin{aligned}
\|P_j u\|_{A_j}^2 = a_j(P_j u, P_j u) &= a_j(P_j u, u) && \text{(definition of } P_j\text{)} \\
&\leq \|P_j u\|_{A_j} \|u\|_{A_j} && \text{(Cauchy–Schwarz)}
\end{aligned}$$

from which we see that

$$\|P_j u\|_{A_j} \leq \|u\|_{A_j}$$

i.e.

$$a_j(P_j u, P_j u) \leq a_j(u, u).$$

Therefore,

$$a(Pu, u) \leq \sum_{j=0}^{J} a_j(u, u).$$

Every triangle $\kappa \in \mathcal{T}$ is included in exactly 4 subdomains Ω_j (the coarse subdomain $\Omega_0 = \Omega$ and the three vertex subdomains Ω_i corresponding to the three vertices of κ). Therefore,

$$a(Pu, u) \leq \sum_{j=0}^{J} a_j(u, u) \leq 4a(u, u).$$

Stable Decomposition

It remains to bound the smallest eigenvalue of T. This is equivalent to demonstrating a **stable decomposition** of V_h into $\sum_{j=0}^J V_j$, meaning that, for all $v \in V_h$, we wish to find $v_j \in V_j$ with $\sum v_j = v$ and

$$\sum_{j=0}^{J} a(v_j, v_j) \lesssim a(v, v).$$

Let $v \in V_h$ be given. Define $v_0 = Q_0 v$ to be the *coarse grid interpolant* of v. In other words, v_0 is the unique element of V_0 such that v_0 takes the same value as v at the coarse grid vertices. A standard result on the stability and accuracy of piecewise linear approximation gives that

$$\begin{aligned} a(v_0, v_0) &\lesssim a(v, v), \\ \|v - v_0\|^2_{L^2(\Omega)} &\lesssim h^2 a(v, v). \end{aligned}$$

We then wish to decompose $v_r = v - v_0$ as a sum $\sum_{j=1}^J v_j$. Since $V = \sum_{j=1}^J V_j$ (omitting the coarse space) is a direct sum, this decomposition is unique. We see that for the jth vertex $\boldsymbol{x}_j$, $v_j = v_r(\boldsymbol{x}_j)\phi_j$, where ϕ_j is the corresponding fine space basis function.

We write $v_j = Q_j(v_r)$, and wish to bound the norm of the operator Q_j. We consider $Q_j : H^1(\Omega_j) \to V_j$. We work on a single triangle $\kappa \in \mathcal{T}_j$ and change variables to the reference triangle $\hat{\kappa}$, so that

$$v_j(\boldsymbol{x}) = \hat{v}_j(T_\kappa^{-1}(\hat{\boldsymbol{x}})).$$

Here, $T : \hat{\kappa} \mapsto \kappa$ is the affine element transformation. The function T will scale coordinates by a factor of h, and T^{-1} will scale by a factor of h^{-1}. From this change of variables, we have

$$\|v_j\|^2_{L^2(\kappa)} \lesssim h^2 \|\hat{v}_j\|^2_{L^2(\hat{\kappa})}, \qquad \|\nabla v_j\|^2_{L^2(\kappa)} \lesssim \|\nabla \hat{v}_j\|^2_{L^2(\hat{\kappa})}.$$

Let $\widehat{Q} : \mathcal{P}^1(\hat{\kappa}) \to \mathcal{P}^1(\hat{\kappa})$ denote the mapping

$$\widehat{Q} : a_1\phi_1 + a_2\phi_2 + a_3\phi_3 \mapsto a_1\phi_1.$$

Then, on κ, mapping vertex j of $\mathcal{T}$ to vertex 1 of $\hat{\kappa}$, we have

$$v_j(\boldsymbol{x}) = \widehat{Q}\hat{v}_r(T^{-1}\boldsymbol{x}).$$

Since $\widehat{Q}$ is a fixed finite dimensional linear operator, it is bounded, and we have

$$\|\widehat{Q}\hat{v}\|_{H^1(\hat{\kappa})} \lesssim \|\hat{v}\|_{H^1(\hat{\kappa})},$$

which then implies

$$\|\nabla v_j\|^2_{L^2(\kappa)} \lesssim \|\nabla \hat{v}_j\|^2_{L^2(\kappa)} \leq \|\widehat{Q}\hat{v}_r\|^2_{H^1(\kappa)}\| \lesssim \|\hat{v}_r\|^2_{H^1(\hat{\kappa})}.$$

Furthermore,

$$\begin{aligned} \|\hat{v}_r\|^2_{H^1(\hat{\kappa})} &= \|\hat{v}_r\|^2_{L^2(\hat{\kappa})} + \|\nabla \hat{v}_r\|^2_{L^2(\hat{\kappa})} \\ &\lesssim h^{-2}\|v_r\|^2_{L^2(\kappa)} + \|\nabla v_r\|^2_{L^2(\kappa)}. \end{aligned}$$

Summing over all triangles $\kappa \in \mathcal{T}_j$, we have

$$\begin{aligned}\|\nabla v_j\|^2_{L^2(\Omega)} = \|\nabla v_j\|^2_{L^2(\Omega_j)} &\lesssim h^{-2}\|v - v_0\|^2_{L^2(\Omega_j)} + \|\nabla v_r\|^2_{L^2(\Omega_j)} \\ &\lesssim \|\nabla v\|^2_{L^2(\Omega_j)}.\end{aligned}$$

Summing over all j and again using the finite overlap property,

$$\begin{aligned}\sum_{j=0}^{J} a(v_j, v_j) &= a(v_0, v_0) + \sum_{j=1}^{J} a(v_j, v_j) \\ &\lesssim a(v, v) + \sum_{j=1}^{J} \|\nabla v\|^2_{L^2(\Omega_j)} \\ &\leq a(v, v) + 3\|\nabla v\|^2_{L^2(\Omega)} \\ &= 4a(v, v),\end{aligned}$$

thereby proving the existence of a stable decomposition (independent of h).

From this, we conclude that both extremal eigenvalues of the preconditioned operator T are bounded independent of h, and so an asymptotically constant number of conjugate gradient iterations are required to solve the system when using the two-level domain decomposition preconditioner.

2.1 Computational Complexity and Scalability

The preconditioner $\widetilde{B}$ can be written in the form

$$\widetilde{B} = \sum_{j=0}^{J} I_j B_j I_j^T,$$

where $B_j \approx A_j^{-1}$. For $j > 0$, A_j is a 1×1 matrix, and so we take $B_j = A_j^{-1}$. Furthermore, I_j^T simply extracts the jth entry. From these considerations, we see that the computational cost of the vertex patch solvers is almost negligible.

The main cost is the coarse grid solve,

$$I_0 B_0 I_0^T.$$

In particular, A_0 will be approximately 1/4 the size of the original matrix A. Although this means that A_0^{-1} can potentially be computed (e.g. using a sparse direct factorization) much faster than A, in the limit as the mesh size increases, the method will not be scalable (computing A_0 has the same asymptotic scaling as A).