Lecture Note #13 **MTH652: Advanced Numerical Analysis**
Winter 2025 Instructor: Will Pazner

Preconditioning SPD Problems

1 Motivation

In the previous lecture, we saw that saddle-point systems of the form

$$\mathcal{A} = \begin{pmatrix} A & B^T \\ B & -C \end{pmatrix}$$

where A and C are symmetric and positive-definite can be solved using the MINRES Krylov method. The block-diagonal preconditioner

$$\mathcal{B} = \begin{pmatrix} A^{-1} & 0 \\ 0 & -S^{-1} \end{pmatrix}$$

for Schur complement $S = -C - BA^{-1}B^T$ gives a *uniformly well-conditioned system*, in the sense that the eigenvalues λ of $\mathcal{BA}$ satisfy

$$\lambda \in \left[-1, (1-\sqrt{5})/2\right) \cup \left[1, (1+\sqrt{5})/2\right).$$

This bound is **uniform** meaning that it does not depend on the spectrum or condition number of the blocks of $\mathcal{A}$. In particular, even if A is ill-conditioned (defined on a very fine finite element mesh, for example), then $\mathcal{BA}$ will be well-conditioned.

The preconditioner $\mathcal{B}$ is not practical, because, in general, none of A^{-1}, S, or S^{-1} can be formed efficiently. We can instead replace A^{-1} with B_A, a preconditioner for A, and replace $-S^{-1}$ with B_S, a preconditioner for $-S$, to obtain the block-diagonal preconditioned

$$\widetilde{\mathcal{B}} = \begin{pmatrix} B_A & 0 \\ 0 & B_S \end{pmatrix}.$$

We have therefore reduced the problem of finding a good preconditioner for an indefinite saddle-point system to the problem of finding good preconditioners for symmetric positive-definite systems.

2 Preliminaries

In the following, A will be a SPD matrix. The main quantity we will be interested in is its **condition number**

$$\kappa(A) := \frac{\lambda_{\max}(A)}{\lambda_{\min}(A)} \geq 1.$$

The condition number of A controls, among other things, the speed of convergence of the conjugate gradient algorithm for solving the linear system $A\boldsymbol{x} = \boldsymbol{b}$.

2.1 Generalized Eigenvalues

The extremal eigenvalues of SPD A are given by the maximum and minimum values of the Rayleigh quotient,

$$\lambda_{\min}(A) = \inf_{\boldsymbol{x}} \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{x}}, \qquad \lambda_{\max}(A) = \sup_{\boldsymbol{x}} \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{x}}.$$

Proposition 1. *Given another SPD matrix M, the following eigenvalue problems have the same eigenvalues*

$$\begin{aligned} A\boldsymbol{x} &= \lambda M \boldsymbol{x}, \\ M^{-1} A \boldsymbol{x} &= \lambda \boldsymbol{x}, \\ M^{-1/2} A M^{-1/2} \boldsymbol{x} &= \lambda \boldsymbol{x}, \\ A^{1/2} M^{-1} A^{1/2} \boldsymbol{x} &= \lambda \boldsymbol{x}. \end{aligned}$$

The first of these is a "generalized eigenvalue problem". In all cases, the eigenvalues are real and positive. The extremal eigenvalues satisfy

$$\lambda_{\min}(M^{-1}A) = \inf_{\boldsymbol{x}} \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T M \boldsymbol{x}}, \qquad \lambda_{\max}(M^{-1}A) = \sup_{\boldsymbol{x}} \frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T M \boldsymbol{x}}. \tag{1}$$

2.2 Spectral Equivalence

We say that A is spectrally equivalent to M (another SPD matrix) with constants c and C if

$$c\boldsymbol{x}^T M \boldsymbol{x} \le \boldsymbol{x}^T A \boldsymbol{x} \le C \boldsymbol{x}^T M \boldsymbol{x} \tag{2}$$

for all $\boldsymbol{x} \in \mathbb{R}^N$.

Proposition 2. *If A and M are spectrally equivalence with constants c and C, then the spectrum of $M^{-1}A$ is contained in $[c, C]$ (and so $\kappa(M^{-1}A) \le C/c$).*

Proof. Note that (2) provides upper and lower bounds on the Rayleigh quotients (1), so the result follows from Proposition 1. We give a direct proof here for completeness. Let $\boldsymbol{v}$ be an eigenvector of $M^{-1}A$, $M^{-1}A\boldsymbol{v} = \lambda \boldsymbol{v}$. Then,

$$A\boldsymbol{v} = \lambda M \boldsymbol{v}$$

and so

$$\boldsymbol{v}^T A \boldsymbol{v} = \lambda \boldsymbol{v}^T M \boldsymbol{v}$$

giving

$$\lambda = \frac{\boldsymbol{v}^T A \boldsymbol{v}}{\boldsymbol{v}^T M \boldsymbol{v}}.$$

from which (together with (2)) it follows that

$$c \le \lambda \le C. \qquad \square$$

This means that if M is spectrally equivalent to A (with constants such that C/c is not too large), then $B = M^{-1}$ is a good preconditioner for A. The goal in constructing a preconditioner is, given A, to find a matrix or linear operator M such that:

- A and M are spectrally equivalent
- The action of M^{-1} is "easy" to compute

These two criteria are somewhat in tension. We can take two extremes. If $M = I$, then the action of M^{-1} is trivial to compute (it is a "no-op"), but the constants are $c = \lambda_{\min}$ and $C = \lambda_{\max}$, which result in no improvement at all. On the other hand, if $M = A$, then $c = C = 1$, and the preconditioned system $M^{-1}A = I$ has condition number 1; however, computing the action of $M^{-1} = A^{-1}$ is the same as the original problem. A *good preconditioner* is (much) cheaper to compute than A^{-1}, while simultaneously satisfying $\kappa(M^{-1}A) \ll \kappa(A)$. We want to trade some modest amount of computational effort for a significant improvement in conditioning.

2.3 Preconditioned Conjugate Gradient

The conjugate gradient method works only on SPD matrices. Although the matrix $M^{-1}A$ may be well-conditioned in terms of its extremal eigenvalues, it will generally not be symmetric. However, we can address this issue by working in the inner product induced by M (recall that M must also be SPD, so it induces an inner product). Let $(\cdot, \cdot)_M$ denote the M-inner product,

$$(\boldsymbol{x}, \boldsymbol{y})_M := \boldsymbol{y}^T M \boldsymbol{x}.$$

An important fact is that the matrix $M^{-1}A$ is symmetric *with respect to the M-inner product*, meaning that

$$(M^{-1}A\boldsymbol{x}, \boldsymbol{y})_M = (\boldsymbol{x}, M^{-1}A\boldsymbol{y})_M.$$

This can be verified directly,

$$\begin{aligned}
(M^{-1}A\boldsymbol{x}, \boldsymbol{y})_M &= \boldsymbol{y}^T M M^{-1} A \boldsymbol{x} \\
&= \boldsymbol{y}^T A \boldsymbol{x} \\
&= \boldsymbol{y}^T A M^{-1} M \boldsymbol{x} \\
&= (M^{-1}A\boldsymbol{y})^T M \boldsymbol{x} \\
&= (\boldsymbol{x}, M^{-1}A\boldsymbol{y})_M,
\end{aligned}$$

using symmetry of A and M. Consequently, the matrix $M^{-1}A$ is SPD in the M-inner product. This means that instead of applying the (standard) conjugate gradient method to the system $A\boldsymbol{x} = \boldsymbol{b}$, we can apply a modified conjugate gradient method to the system

$$M^{-1}A\boldsymbol{x} = M^{-1}\boldsymbol{b},$$

where the modification is to use the M-inner product instead of the standard inner product. This is called the **preconditioned conjugate gradient** (PCG) method. Some relatively straightforward algebraic manipulations show that although the inner product $(\cdot, \cdot)_M$ is used, matrix M is **not needed** in the PCG algorithm; every iteration requires computing once the action of each of A and M^{-1}. The action of M is not required.

From the above discussion, we conclude that if we have a SPD preconditioner $B = M^{-1}$ such that

$$\kappa(BA) \ll \kappa(A),$$

then the convergence of the CG algorithm can be accelerated. The cost is one additional application of B per iteration.

2.4 Parameter Independence

Typically, in finite element problems, the goal is to find a preconditioner such that the constants of equivalence c and C are *independent of discretization parameters*, most importantly, independent of the mesh size h. Such preconditioners are called **uniform**.

Recall that the condition number of the stiffness matrix scales as $\kappa(A) = \mathcal{O}(h^{-2})$. If we can find a uniform preconditioner B, then $\kappa(BA) = \mathcal{O}(1)$, and we can solve the finite element problems in a constant number of iterations, *independent of mesh size*. The construction of such preconditioners is usually problem-dependent. We have to leverage some specific structure of the problem giving rise to the matrix A in order to construct a good preconditioner. For elliptic problems, two common (related) approaches are **domain decomposition** and **multigrid**.

2.5 Change of Inner Product

Our main tool for estimate condition numbers will be estimating bounds for the Rayleigh quotient

$$\frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T \boldsymbol{x}} = \frac{(A\boldsymbol{x}, \boldsymbol{x})}{(\boldsymbol{x}, \boldsymbol{x})}.$$

As seen above, the eigenvalues of the preconditioned operator $M^{-1}A$ can be bounded using the Rayleigh quotient

$$\frac{\boldsymbol{x}^T A \boldsymbol{x}}{\boldsymbol{x}^T M \boldsymbol{x}} = \frac{(A\boldsymbol{x}, \boldsymbol{x})}{(M\boldsymbol{x}, \boldsymbol{x})}.$$

Since A is SPD, it induces an inner product

$$(\boldsymbol{x}, \boldsymbol{y})_A := \boldsymbol{y}^T A \boldsymbol{x}.$$

Let T be an operator that is positive-definite and symmetric with respect to the A-inner product. As one example, we could take $T = M^{-1}A$. The eigenvalues of T can be bounded using the Rayleigh quotient

$$\frac{(T\boldsymbol{x}, \boldsymbol{x})_A}{(\boldsymbol{x}, \boldsymbol{x})_A} = \frac{\boldsymbol{x}^T A T \boldsymbol{x}}{\boldsymbol{x}^T A \boldsymbol{x}}.$$

This can also be seen directly using Proposition 1. Since T is symmetric with respect to A, AT is symmetric. By Proposition 1, the eigenvalues of $A^{-1}AT = T$ are the same as the generalized eigenvalues $AT\boldsymbol{x} = \lambda A\boldsymbol{x}$, which are bounded by the same Rayleigh quotient. It will often be more convenient to work with the inner product induced by the operator A.

3 Domain Decomposition

Domain decomposition methods are sometimes also called **subspace correction** methods. We will describe here the "additive" version of these preconditioners. These sometimes called **additive Schwarz** preconditioners or **parallel subspace correction** preconditioners; both terms refer to the same concept.

Suppose that A is SPD with respect to the inner product $(\cdot\,,\cdot)$ defined on the finite-dimensional space V. (For the stiffness matrix, the space V can be the finite element space V_h, and the inner product can be the L^2 inner product). We conditioner a decomposition of V into $J+1$ subspaces,

$$V = \sum_{j=0}^{J} V_j.$$

This decomposition will define the preconditioner.

For each j, let $Q_j : V \to V_j$ be the **projection** in the inner product $(\cdot\,,\cdot)$ onto V_j,

$$(Q_j u, v_j) = (u, v_j) \qquad \text{for all } v_j \in V_j,$$

and let $P_j : V \to V_j$ be the **elliptic projection** onto V_j,

$$(AP_j u, v_j) = (Au, v_j) \qquad \text{for all } v_j \in V_j.$$

Let $A_j : V_j \to V_j$ be the restriction of A onto V_j defined by

$$(A_j u_j, v_j) = (Au_j, v_j) \qquad \text{for all } v_j \in V_j.$$

We can compute

$$\begin{aligned}(A_j P_j u, v_j) &= (AP_j u, v_j)\\ &= (Au, v_j)\\ &= (Q_j Au, v_j),\end{aligned}$$

which holds for all $u \in V$ and $v_j \in V_j$, from which we derive the identity

$$A_j P_j = Q_j A,$$

or, equivalently,

$$P_j = A_j^{-1} Q_j A$$

The **preconditioned system** $P = BA$ is defined by

$$P = \sum_{j=0}^{J} P_j = \sum_{j=0}^{J} A_j^{-1} Q_j A = \left(\sum_{j=0}^{J} A_j^{-1} Q_j \right) A,$$

and so the "additive Schwarz" or "parallel subspace correction" preconditioner associated with the decomposition $V = \sum V_j$ is

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

This construction allows us to precondition a large $N \times N$ system by solving a sequence of J smaller systems, each of size $\dim(V_j)$.

3.1 Note on Implementation

Consider $A \in \mathbb{R}^{N\times N}$, and suppose that each subspace V_j is given by

$$V_j = \operatorname{span}\{\boldsymbol{e}_i : i \in \mathcal{I}_j\},$$

in other words, V_j is the subspace spanned by a subset $\mathcal{I}_j$ of the standard basis vectors. Then, the projection Q_j simply extracts the subvector with indices in $\mathcal{I}_j$, and the operator A_j is the submatrix obtained by selecting rows and columns in $\mathcal{I}_j$. The simplest such decomposition is

$$V = \sum_{j=1}^{N} V_j, \qquad V_j := \operatorname{span}\{\boldsymbol{e}_j\}.$$

Here, there are N subspaces, each one-dimensional. In this case, $A_j = a_{jj}$ and $Q_j\boldsymbol{u} = u_j\boldsymbol{e}_j$. The preconditioner B is simply diagonal scaling with the reciprocal of the diagonal of A (i.e. Jacobi preconditioning).

More generally, for each subspace V_j, we have the inclusion $I_j : V_j \hookrightarrow V$. The projection Q_j with respect to the ℓ^2 inner product is simply I_j^T. If $A_j : V_j \to V_j$ is the restriction of A to V_j, then the preconditioner B is given by

$$B = \sum_{j=0}^{J} I_j A_j^{-1} I_j^T.$$

3.2 Conditioning

Note that if the subspaces V_j are themselves large, we might want to replace A_j^{-1} with a a SPD preconditioner B_j, to obtain

$$\widetilde{B} := \sum_{j=0}^{J} B_j Q_j.$$

This can be viewed as approximating the projections $P_j = A_j^{-1}Q_jA$ with approximations $T_j = B_jQ_jA$, to obtain

$$T := \sum_{j=0}^{J} T_j = \widetilde{B}A.$$

We wish to estimate the condition number of T (which is the preconditioned system $\widetilde{B}A$).

Let $(\cdot\,,\cdot)_A$ denote the inner product induced by A, i.e.

$$(u, v)_A := (Au, v).$$

Proposition 3. *Let B_j be symmetric and positive-definite. The approximate projection $T_j = B_jQ_jA$ restricted to V_j is symmetric and positive definite in the inner product induced by A.*

Proof. For any $u_j, v_j \in V_j$, notice that

$$\begin{aligned}
(T_j u_j, v_j)_A &= (B_j Q_j A u_j, v_j)_A \\
&= (A B_j Q_j A u_j, v_j) \\
&= (B_j Q_j A u_j, A v_j) && \text{(symmetry of } A\text{)} \\
&= (B_j Q_j A u_j, Q_j A v_j) && \text{(definition of } Q_j\text{)} \\
&= (Q_j A u_j, B_j Q_j A v_j) && \text{(symmetry of } B_j\text{)} \\
&= (A u_j, B_j Q_j A v_j) && \text{(definition of } Q_j\text{)} \\
&= (u_j, T_j v_j)_A,
\end{aligned}$$

so T_j is symmetric in the inner product induced by A. Furthermore, from the previous derivation

$$(T_j u_j, u_j)_A = (B_j Q_j A u_j, Q_j A u_j) \gtrsim \|Q_j A u_j\|^2$$

since B_j is positive-definite. Therefore, T_j is SPD in the A-inner product. $\square$

Lemma 1. *It holds that*

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

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

Proof. Let $v \in V$ and define $v_j = T_j T^{-1} v$, so that

$$\sum_{j=0}^{J} v_j = \sum_{j=0}^{J} T_j T^{-1} v = \left(\sum_{j=0}^{J} T_j \right) T^{-1} v = v.$$

Then,

$$\begin{aligned}
\inf_{\sum v_j = v} \sum_{j=0}^{J} (T_j^{-1} v_j, v_j)_A &= \inf_{\sum w_j = 0} \sum_{j=0}^{J} (T_j^{-1}(v_j + w_j), v_j + w_j)_A \\
&= \inf_{\sum w_j = 0} \sum_{j=0}^{J} (T^{-1} v + T_j^{-1} w_j, v_j + w_j)_A \\
&= \inf_{\sum w_j = 0} \sum_{j=0}^{J} \left((T^{-1} v, v_j)_A + (T^{-1} v, w_j)_A + (T_j^{-1} w_j, v_j)_A + (T_j^{-1} w_j, w_j)_A \right) \\
&= (T^{-1} v, v)_A + \inf_{\sum w_j = 0} \sum_{j=0}^{J} \left((T_j^{-1} w_j, v_j)_A + (T_j^{-1} w_j, w_j)_A \right) \\
&= (T^{-1} v, v)_A + \inf_{\sum w_j = 0} \sum_{j=0}^{J} \left((w_j, T v)_A + (T_j^{-1} w_j, w_j)_A \right) \\
&= (T^{-1} v, v)_A + \inf_{\sum w_j = 0} \sum_{j=0}^{J} (T_j^{-1} w_j, w_j)_A \\
&= (T^{-1} v, v)_A.
\end{aligned}$$

$\square$

Note the special case when the projections are exact, i.e. $T_j = P_j$. Then, $P_j|_{V_j} = I$, and

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

Recall that $T_j = B_j Q_j A$. If B_j is spectrally equivalent to A_j^{-1}, then

$$(T_j^{-1} v_j, v_j)_A \approx (P_j^{-1} v_j, v_j)_A = (v_j, v_j),$$

and so

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

The above arguments show that estimating the condition number of the preconditioned system is reduced to bounding above and below the quantity

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

in terms of $(v, v)_A$. These will give bounds on the extremal eigenvalues of T^{-1} (and hence on the preconditioned operator T).