CLASSICAL ITERATIVE METHODS

LONG CHEN

In this notes we discuss classic iterative methods on solving the linear operator equation

$$Au = f, \tag{1}$$

posed on a finite dimensional Hilbert space $\mathbb{V} \cong \mathbb{R}^N$ equipped with an inner product $(\cdot,\cdot)$. Here $A : \mathbb{V} \mapsto \mathbb{V}$ is an *symmetric and positive definite (SPD)* operator, $f \in \mathbb{V}$ is given, and we are looking for $u \in \mathbb{V}$ such that (1) holds.

The direct method to solve (1) is to form A^{-1} or the action of A^{-1}. For example, the Gaussian elimination or LU factorization still remains the most commonly used methods in practice. It is a black-box as it can be applied to any problem in principle. For general dense matrices, a matrix-vector product requires $\mathcal{O}(N^2)$ operations and the straightforward implementation of Gauss elimination is $\mathcal{O}(N^3)$, which is prohibitive large when N is large. The state-of-the-art of direct solvers can achieve the nearly linear complexity for certain structured sparse matrices; see for example [2].

When A is sparse, the nonzero entries of A is $\mathcal{O}(N)$ and the basic matrix-vector product reduces to $\mathcal{O}(N)$ operation. Then it is desirable to design optimally scaled solvers, say, with $\mathcal{O}(N)$ or $\mathcal{O}(N \log N)$ computational cost. Namely computing $A^{-1}f$ is just a few number of Ax. To this end, we first introduce a basic residual-correction iterative method and study classic iterative methods.

To see the huge saving of an $\mathcal{O}(N)$ algorithm comparing with an $\mathcal{O}(N^2)$ one when N is large, let us do the following calculation. Suppose $N = 10^6$ and a standard PC can do the summation of 10^6 numbers in 1 minute. Then an $\mathcal{O}(N)$ algorithm will finish in few minutes while an $\mathcal{O}(N^2)$ algorithm will take nearly two years (10^6 minutes $\approx$ 694 days).

1. RESIDUAL-CORRECTION METHOD

We follow closely Xu [3, 4, 5] to introduce an iterative method in the residual-correction form. Starting from an initial guess $u_0 \in \mathbb{V}$, one such iteration consists of three steps to get a new approximation u_{k+1} from the current one u_k:

(1) form the residual $r = f - Au_k$;
(2) compute a correction $e = Br$ with $B \approx A^{-1}$;
(3) obtain the next approximation $u_{k+1} = u_k + e$.

The operator B is called *iterator* and assumed to be nonsingular. Given an iterator B, we define $\Phi_B(u; f) = u + B(f - Au) = (I - BA)u + Bf$. The residual-correction iterative method can be written as

$$u_{k+1} = \Phi_B(u_k; f) = u_k + B(f - Au_k). \tag{2}$$

The mapping $\Phi_B(\cdot; 0)$ is linear with respect to u and thus called linear iterative methods. Since the exact solution u is a fixed point of the mapping $\Phi_B(\cdot; f)$, i.e., $u = \Phi_B(u; f) = (I - BA)u + Bf$, we then have the error formula

$$u - u_{k+1} = (I - BA)(u - u_k). \tag{3}$$

The matrix $S = I - BA$ will be called the *iterative matrix* which is also the amplification matrix for the error.

Another popular linear iterative method is based on a splitting of A [1]. Let $A = M - N$ with nonsingular M. Usually M is chosen as a dominant part comparing with N. We rewrite the equation as

$$Mu - Nu = f,$$

and derive the iterative method based on the matrix-splitting

$$u_{k+1} = M^{-1}(Nu_k + f). \tag{4}$$

Comparing the residual-correction form and the matrix-splitting form, one can easily derive the relation $B = M^{-1}$ and $N = M - A = B^{-1} - A$. The matrix-splitting method is more computational efficient since Nx requires less operations than Ax. The residual-correction method emphasize the role of solving the residual equation $Ae = r$. In contrast iteration (4) updates u directly and thus will be also called direct updated form.

The art of constructing *efficient* iterative methods lies on the design of B which captures the essential information of A^{-1} and its action is easily computable. In this context the notion of "efficient" implies two essential requirements:

(1) One iteration require only $\mathcal{O}(N)$ or $\mathcal{O}(N \log N)$ operations.
(2) The contraction rate is well below 1 and independent of N.

2. Classic iterative methods

Let us consider the case $\mathbb{V} = \mathbb{R}^N$ and $\boldsymbol{A}$ a SPD matrix. We will derive several linear iterative methods based on the splitting

$$\boldsymbol{A} = \boldsymbol{D} + \boldsymbol{L} + \boldsymbol{U}.$$

Here $\boldsymbol{D}, \boldsymbol{L}, \boldsymbol{U}$ are diagonal matrix, lower triangular matrix and upper triangular matrix of A. A list of iterators is presented below

- Richardson $\quad \boldsymbol{B}_R = \omega \boldsymbol{I}$
- Jacobi $\quad \boldsymbol{B}_J = \boldsymbol{D}^{-1}$.
- Weighted Jacobi $\quad \boldsymbol{B}_{DJ} = \omega \boldsymbol{D}^{-1}$.
- Forward Gauss-Seidel $\quad \boldsymbol{B}_{GS} = (\boldsymbol{D} + \boldsymbol{L})^{-1}$
- Backward Gauss-Seidel $\quad \boldsymbol{B}_{GS} = (\boldsymbol{D} + \boldsymbol{U})^{-1}$
- Symmetric Gauss-Seidel $\quad \bar{\boldsymbol{B}}_{GS} = (\boldsymbol{D} + \boldsymbol{U})^{-1}\boldsymbol{D}(\boldsymbol{D} + \boldsymbol{L})^{-1}$
- Successive Over-Relaxation (SOR) $\quad \boldsymbol{B}_{SOR} = \omega(\boldsymbol{D} + \omega \boldsymbol{L})^{-1}$.
- Symmetric SOR $\quad \boldsymbol{B}_{SSOR} = \bar{\boldsymbol{B}}_{SOR} = \omega(2-\omega)(\boldsymbol{D} + \omega \boldsymbol{U})^{-1}\boldsymbol{D}(\boldsymbol{D} + \omega \boldsymbol{L})^{-1}$

We use forward Gauss-Seidel as an example to write out its algorithmic description. Multiply $D + L$ to $u_{k+1} = u_k + (D + L)^{-1}(f - Au_k)$ to get $(D + L)u_{k+1} = (D + L)u_k + f - Au_k$. Using the fact $A = D + L + U$, it can be formally written as

$$u_{k+1} = D^{-1}(f - Lu_{k+1} - Uu_k). \tag{5}$$

We then end up with the following in-place implementation of Gauss-Seidel iteration.

```
for i=1:N
    u(i) = a_ii^{-1} (b(i) - sum_{j=1}^{i-1} a_ij u(j) - sum_{j=i+1}^{N} a_ij u(j));
end
```

In the above algorithm, we use only one vector $\boldsymbol{u}$ to store both u_{k+1} and u_k. The transition from u_k to u_{k+1} is built into the loop. To reveal the transition more explicitly, let us

introduce the dynamically updated vector

$$v^i = (u_{k+1}^1, \ldots, u_{k+1}^i, u_k^{i+1}, \ldots, u_k^N),$$

i.e., v^i is the vector of iteration when the i-th component is updated by the Gauss-Sedel iteration. Then $v^0 = u_k$ and $v^N = u_{k+1}$ and

$$v^i - v^{i-1} = u_{k+1}^i - u_k^i.$$

The Gauss-Sedel iteration can be also written in terms of v as

```
for i=1:N
```
$v^i = v^{i-1} + a_{ii}^{-1}(b_i - (Av^{i-1})_i);$
```
end
```

Again in implementation, only one vector $\boldsymbol{v}$ is needed.

The form (5) is called direct update form. One iteration requires almost the same operations as computing Ax. The correction form

$$u_{k+1} = u_k + (D + L)^{-1}(f - Au_k),$$

requires an evaluation of the residual and a forward substitution which is almost two times slower. On the other hand, in MATLAB, it is much easier and faster to implement the correction form

```
u = u + tril(A)\(f-A*u);
```

Exercise 2.1. Derive the direct updated form of the Jacobi iteration and write its algorithmic description. Compare with G-S and list the main difference.

3. Convergence analysis of residual-correction methods

In this section, we shall analyze the convergence of the linear residual-correction iterative method and its variants. Given an SPD operator A in $\mathbb{V}$, we can define a new inner product $(\cdot,\cdot)_A$ of $\mathbb{V}$:

$$(u, v)_A = (Au, v), \quad \text{for all } u, v \in \mathbb{V}.$$

We shall use $(\mathbb{V}, I)$ and $(\mathbb{V}, A)$ to denote the same linear space $\mathbb{V}$ but with two different inner product structures. It turns out that $(\mathbb{V}, A)$ plays an important role in the convergence analysis. We shall use * for the adjoint with respect to $(\mathbb{V}, A)$ and $^\intercal$ to $(\mathbb{V}, I)$, i.e.,

$$(B^\intercal u, v) = (u, Bv) \quad \text{for all } u, v \in \mathbb{V}$$
$$(B^* u, v)_A = (u, Bv)_A \quad \text{for all } u, v \in \mathbb{V}.$$

For two symmetric operators X, Y, we introduce the notation $X \geq Y$, if $(Xu, u) \geq (Yu, u)$ for all $u \in \mathbb{V}$. We can generalize to A-symmetric operators and write inequality $X \geq_A Y$ when X, Y are symmetric in the $(\cdot,\cdot)_A$ inner product.

3.1. General convergence analysis.

Let $e_k = u - u_k$. Recall that the error equation of the residual-correction iterative method is

$$e_{k+1} = (I - BA)e_k = (I - BA)^{k+1} e_0.$$

The residual-correction iterative method converges if and only if $\rho(I - BA) < 1$ which is equivalent to

$$|1 - \lambda| < 1 \quad \text{for all } \lambda \in \sigma(BA).$$

Namely the spectrum of BA is inside the unit disk in the complex domain at center $(1, 0)$. Estimate of eigenvalues of BA is thus the crucial ingredient. Here recall that for a linear operator $T \in \mathcal{L}(\mathbb{V}, \mathbb{V})$, the *spectrum* of T is defined as $\sigma(T) = \{\lambda : \lambda \text{ is an eigenvalue of } T\}$. The *spectral radius* of T is $\rho(T) = \sup_{\lambda \in \sigma(T)} |\lambda|$.

We emphasize that by the definition of eigenvalue it depends only the linear structure of the operator not the inner product. Closing a right inner product, however, can help us on the study of eigenvalues.

3.2. **Symmetric scheme.** Eigenvalues of operator BA could be complex and thus the estimate is difficult. When B is symmetric, BA may not be symmetric but will be symmetric in a different inner product.

Exercise 3.1. Suppose A and B are SPD, prove that BA is SPD with respect to the inner product $(\cdot,\cdot)_A$ or $(\cdot,\cdot)_{B^{-1}}$.

The right Hilbert space to work with is $(\mathbb{V}, A)$ not the default one $(\mathbb{V}, I)$. In the right space, everything becomes transparent. One immediate benefit of the fact BA is symmetric in A-inner product is that all eigenvalues of BA are real numbers. Thus

$$\rho(I - BA) = \max\{|1 - \lambda_{\min}(BA)|, |1 - \lambda_{\max}(BA)|\}. \tag{6}$$

From (6), we get a characterization of the convergence rate of a symmetric scheme.

Theorem 3.2. *For symmetric iterator B, the iterative scheme Φ_B converges if and only if*

$$0 < \lambda_{\min}(BA) \leq \lambda_{\max}(BA) < 2.$$

On the other hand, from $\rho(I - BA) < 1$ we can derive bound on the eigenvalues.

Corollary 3.3. *For a symmetric iterator B, if $\rho = \rho(I - BA) < 1$. Then*

$$1 - \rho \leq \lambda_{\min}(BA) \leq \lambda_{\max}(BA) \leq 1 + \rho.$$

Proof. It is from the inequality $|1 - x| \leq \rho$ for $x \in [\lambda_{\min}, \lambda_{\max}]$. □

To get more quantitative information, we need to estimate $\lambda_{\min}(BA)$ and $\lambda_{\max}(BA)$ by comparing B^{-1} with A or B with A^{-1}.

Lemma 3.4. *When B is symmetric and nonsingular in $(\mathbb{V}, I)$,*

$$\lambda_{\min}(BA) = \inf_{u\in\mathbb{V}\backslash\{0\}} \frac{(ABAu,u)}{(Au,u)} = \inf_{u\in\mathbb{V}\backslash\{0\}} \frac{(Bu,u)}{(A^{-1}u,u)} = \left[\sup_{u\in\mathbb{V}\backslash\{0\}} \frac{(B^{-1}u,u)}{(Au,u)}\right]^{-1},$$

$$\lambda_{\max}(BA) = \sup_{u\in\mathbb{V}\backslash\{0\}} \frac{(ABAu,u)}{(Au,u)} = \sup_{u\in\mathbb{V}\backslash\{0\}} \frac{(Bu,u)}{(A^{-1}u,u)} = \left[\inf_{u\in\mathbb{V}\backslash\{0\}} \frac{(B^{-1}u,u)}{(Au,u)}\right]^{-1}.$$

Proof. Due to the similarity, we only prove the formulae for $\lambda_{\min}(BA)$. The first two comes from the fact BA is symmetric in $(\mathbb{V}, A)$. The third identity can be proved as

$$\lambda_{\min}^{-1}(BA) \quad = \quad \lambda_{\max}((BA)^{-1}) = \sup_{u\in\mathbb{V}\backslash\{0\}} \frac{((BA)^{-1}u,u)_A}{(u,u)_A} = \sup_{u\in\mathbb{V}\backslash\{0\}} \frac{(B^{-1}u,u)}{(Au,u)}.$$

□

Using the ordering notation for symmetric operators, the estimate $\lambda_{\min}(BA) \geq c_0$ is equivalent to $BA \geq_A c_0 I$ and can be derived from the inequalities

$$c_0 B^{-1} \leq A,\ B \geq c_0 A^{-1},\ \text{or}\ ABA \geq c_0 A.$$

That is formally we can manipulate symmetric operators as numbers in inequalities.

3.3. **Symmetrization of general schemes.** For general non-symmetric iterator B, eigenvalues of BA could be complex and not easy to estimate. We define its symmetrization $\Phi_{\overline{B}} = \Phi_{B^\intercal}\Phi_B$, i.e.

(1) $u_{k+\frac{1}{2}} = u_k + B(f - Au_k)$;
(2) $u_{k+1} = u_{k+\frac{1}{2}} + B^\intercal(f - Au_{k+\frac{1}{2}})$.

The symmetry will bring more structure to the estimate of eigenvalues. By the definition

$$I - \overline{B}A = (I - B^\intercal A)(I - BA), \tag{7}$$

and therefore

$$\overline{B} = B^\intercal(B^{-T} + B^{-1} - A)B. \tag{8}$$

Since $\overline{B}$ is symmetric in $(\mathbb{V}, I)$, $I - \overline{B}A$ is symmetric in $(\mathbb{V}, A)$. By the following exercise, it is also semi-positive definite.

Exercise 3.5. Prove that

$$I - \overline{B}A = (I - BA)^*(I - BA). \tag{9}$$

For the symmetrized scheme $\overline{B}$, eigenvalues of $\overline{B}A$ are real numbers and thus

$$\rho(I - \overline{B}A) = \max\{|1 - \lambda_{\min}(\overline{B}A)|, |1 - \lambda_{\max}(\overline{B}A)|\}. \tag{10}$$

By (9), $I - \overline{B}A$ is symmetric and semi-positive definite and thus $\lambda_{\min}(I - \overline{B}A) \geq 0$ which is equivalent to $\lambda_{\max}(\overline{B}A) \leq 1$. Therefore we have the following result.

Lemma 3.6. *For the symmetrized scheme $\Phi_{\overline{B}}$,*

$$\rho(I - \overline{B}A) = 1 - \lambda_{\min}(\overline{B}A). \tag{11}$$

We then present a criterion for the convergence of the symmetrized scheme.

Theorem 3.7. *The symmetrized iterative method $\Phi_{\overline{B}}$ converges if and only if*

$$B^{-1} + B^{-T} - A \text{ is SPD.} \tag{12}$$

Proof. By (11), the following statements are equivalent

(1) $\Phi_{\overline{B}}$ converges,
(2) $\lambda_{\min}(\overline{B}A) > 0$,
(3) $\overline{B}A$ is SPD in $(\mathbb{V}, A)$,
(4) $\overline{B}$ is SPD in $(\mathbb{V}, I)$,
(5) $B^{-1} + B^{-T} - A$ is SPD in $(\mathbb{V}, I)$.

The equivalence of (4) and (5) is from the formula

$$\overline{B} = B^\intercal(B^{-T} + B^{-1} - A)B.$$

□

We summarize the result for the symmetrized scheme $\Phi_{\overline{B}}$ in the following theorem.

Theorem 3.8. *For iterative scheme Φ_B,*

$$\|I - BA\|_A^2 = \rho(I - \overline{B}A) = 1 - \left[\sup_{u \in \mathbb{V}\setminus\{0\}} \frac{(\overline{B}^{-1}u, u)}{(Au, u)}\right]^{-1}.$$

Consequently if

$$(\overline{B}^{-1}u, u) \leq K(Au, u) \quad \text{for all } u \in \mathbb{V}, \tag{13}$$

then

$$\|I - BA\|_A^2 \leq 1 - \frac{1}{K}.$$

Exercise 3.9. Prove that if $B^{-1} > \frac{1}{2}A$, then Φ_B converges with a rate $\|I - BA\|_A \leq (1 - \lambda_{\min}(B^{-1} + B^{-T} - A)\lambda_{\min}(A)\|B^{-1}\|^{-2})^{1/2}$. In view of matrix-splitting method, this condition means the matrix $M = B^{-1}$ is dominant (more than half).

3.4. **Relation of a scheme and its symmetrization.** The convergence of Φ_B and $\Phi_{\overline{B}}$ is connected by the following inequality.

Lemma 3.10.

$$\rho(I - BA) \leq \sqrt{\rho(I - \overline{B}A)},$$

and the equality holds if $B = B^{\intercal}$.

Proof. We use the relation between spectral radius and norms to get

$$\rho(I - BA)^2 \leq \|I - BA\|_A^2 = \|(I - BA)^*(I - BA)\|_A = \rho(I - \overline{B}A).$$

The first inequality holds if B is symmetric. □

Therefore the convergence of symmetrized scheme $\Phi_{\overline{B}}$ will imply the convergence of the original algorithm Φ_B. When B is non-symmetric, e.g. the Gauss-Seidel method, we shall study its symmetrization $\overline{B}$. But we should be cautious that when B is non symmetric, it is possible that Φ_B converges while $\Phi_{\overline{B}}$ does not.

For symmetric B, (12) is a sufficient and necessary condition since the equality holds in Lemma 3.10. We could estimate $\lambda_{\min}(BA)$ and $\lambda_{\max}(BA)$ or $\lambda_{\min}(\overline{B}A)$. Note that even B is symmetric, it symmetrization is different and $\overline{B} = 2B - BAB$ is a better but more expensive iterator.

4. Convergence analysis of classic iterative methods

We shall apply our theories to analyze the convergence of classic iterative methods. We begin with Richardson method for which $B = \omega I$ and discuss the optimal choice of the damping parameter. For an SPD operator, we define the condition number $\kappa(A) = \lambda_{\max}(A)/\lambda_{\min}(A)$.

Theorem 4.1. *Richardson method with $B = \omega I$ converges if and only if $0 < \omega < 2/\lambda_{\max}(A)$. Furthermore, the optimal convergence rate is achieved when*

$$\omega^* = \frac{2}{\lambda_{\min}(A) + \lambda_{\max}(A)},$$

and the optimal convergence rate is

$$\rho_{\omega^*} = \frac{\kappa(A) - 1}{\kappa(A) + 1}.$$

Proof. Since A is SPD, all eigenvalues of A are real numbers, and $\lambda_{\min}(A) > 0$. $\rho(I - \omega A) = \max\{|1 - \omega\lambda_{\min}(A)|, |1 - \omega\lambda_{\max}(A)|\}$. The optimal ω^* minimizes $\max\{|1 - \omega\lambda_{\min}(A)|, |1 - \omega\lambda_{\max}(A)|\}$, and it satisfies

$$\omega^*\lambda_{\max}(A) - 1 = 1 - \omega^*\lambda_{\min}(A).$$

□

Exercise 4.2. Consider k-steps of Richardson methods with different parameters $\omega_1, \ldots, \omega_k$. Then the error equation is

$$e^k = (I - \omega_k A) \cdots (I - \omega_1 A) e^0.$$

Consider the optimization problem of choosing k-parameters:

$$\min_{\omega_i \in \mathbb{R}, i=1,\ldots,k} \left\{ \max_{\lambda \in [\lambda_{\min}(A), \lambda_{\max}(A)]} |(I - \omega_k \lambda) \cdots (I - \omega_1 \lambda)| \right\}. \tag{14}$$

Find the solution of (14) and derive the rate. This trick is known as Chebyshev acceleration.

We now analyze the convergence rate of Jacobi and weighted Jacobi iteration.

Theorem 4.3. *Jacobi method converges if and only if $2\boldsymbol{D} - \boldsymbol{A} = \boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U}$ is an SPD matrix.*

Proof. Since $\boldsymbol{B}_J = \boldsymbol{D}^{-1}$ is an SPD matrix, the characterization of its converges is from Theorem 3.7. □

A matrix is called diagonally dominated if $a_{ii} \geq \sum_{j \neq i} |a_{ij}|$ for all i and is strictly diagonally dominated if it is diagonally dominated and for at least one i, $a_{ii} > \sum_{j \neq i} |a_{ij}|$. One can easily prove that a symmetric and strictly diagonal dominated matrix is SPD.

Corollary 4.4. *If A is strictly diagonally dominated, then Jacobi iteration always converges.*

Proof. Note that if $\boldsymbol{A} = \boldsymbol{D} + \boldsymbol{L} + \boldsymbol{U}$ is strictly diagonal dominated, so is $2\boldsymbol{D} - \boldsymbol{A} = \boldsymbol{D} - \boldsymbol{L} - \boldsymbol{U}$. □

To study the weighted Jacobi iteration, we introduce a scaled matrix $A_D = D^{-1/2} A D^{-1/2}$. By the following exercise, $\sigma(A_D) = \sigma(D^{-1}A)$. We therefore reduce the analysis of the weighted Jacobi method to Richardson method.

Exercise 4.5. Let $A_{n \times r}$ and $B_{r \times n}$ be two matrices. Prove

$$\sigma(AB) \backslash \{0\} = \sigma(BA) \backslash \{0\}.$$

Theorem 4.6. *Weighted Jacobi method with $B = \omega D^{-1}$ converges if and only if $0 < \omega < 2/\lambda_{\max}(A_D)$. Furthermore, the optimal convergence rate is achieved when*

$$\omega^* = \frac{2}{\lambda_{\min}(A_D) + \lambda_{\max}(A_D)},$$

and the optimal convergence rate is

$$\rho_{\omega^*} = \frac{\kappa(A_D) - 1}{\kappa(A_D) + 1}.$$

The diagonal entry of the scaled matrix A_D is always 1. An estimate of $\lambda_{\max}(A_D)$ can be obtained by the Gershgorin circle theorem. For example, if A is diagonally dominated, then $\lambda_{\max}(A_D) \leq 2$

Theorem 4.7. *Gauss-Seidel method always converges. For forward Gauss-Sediel method $\boldsymbol{B} = (\boldsymbol{D} + \boldsymbol{L})^{-1}$,*

$$\|I - \boldsymbol{B}\boldsymbol{A}\|_{\boldsymbol{A}}^2 = \frac{c_0}{1 + c_0},$$

where

$$c_0 = \sup_{\boldsymbol{u} \in \mathbb{R}^N, \|\boldsymbol{u}\|_{\boldsymbol{A}} = 1} (\boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{u}, \boldsymbol{U}\boldsymbol{u}) = \sup_{\boldsymbol{u} \in \mathbb{R}^N, \boldsymbol{u} \neq \boldsymbol{0}} \frac{(\boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{u}, \boldsymbol{U}\boldsymbol{u})}{(\boldsymbol{A}\boldsymbol{u}, \boldsymbol{u})}.$$

Proof. By direct computation, we have

$$\boldsymbol{B}^{-T} + \boldsymbol{B}^{-1} - \boldsymbol{A} = \boldsymbol{D}$$

is an SPD. Thus, by Theorem 3.7, Gauss-Sediel method always converges. Furthermore

$$\overline{\boldsymbol{B}}^{-1} = \boldsymbol{A} + \boldsymbol{L}\boldsymbol{D}^{-1}\boldsymbol{U}.$$

Thus

$$\lambda_{\min}(\overline{\boldsymbol{B}}\boldsymbol{A}) = \sup_{\boldsymbol{u}\neq 0} \frac{(\bar{\boldsymbol{B}}^{-1}\boldsymbol{u}, \boldsymbol{u})}{(\boldsymbol{A}\boldsymbol{u}, \boldsymbol{u})} = 1 + \sup_{\boldsymbol{u}\neq 0} \frac{(\boldsymbol{L}\boldsymbol{D}^{-1}\boldsymbol{U}\boldsymbol{u}, \boldsymbol{u})}{(\boldsymbol{A}\boldsymbol{u}, \boldsymbol{u})}.$$

Results follows from Theorem 3.8 and the fact $\boldsymbol{U} = \boldsymbol{L}^{\intercal}$. □

Exercise 4.8. Prove that the convergence rate of Richardson, weighted Jacobi method, and Gauss-Seidal method for the 5-point stencil finite difference method of the Poisson equation on a uniform mesh with size h, is like

$$\rho \leq 1 - Ch^2.$$

Thus when $h \to 0$, we will observe slow convergence of those classical iterative methods.

Hint: For G-S, use the Hölder inequality of the 2-norm of a matrix M:

$$\|M\|^2 \leq \|M\|_\infty \|M\|_1.$$

REFERENCES

[1] R. S. Varga. *Matrix iterative analysis*. Prentice-Hall, Englewood Cliffs, N. J., 1962.
[2] J. Xia, S. Chandrasekaran, M. Gu, and X. Li. Superfast multifrontal method for large structured linear systems of equations. *SIAM Journal on Matrix Analysis and Applications*, 31(3):1382–1411, 2009.
[3] J. Xu. Iterative methods by space decomposition and subspace correction. *SIAM Rev.*, 34:581–613, 1992.
[4] J. Xu. An introduction to multilevel methods. 1997. published by Oxford University Press, New York.
[5] J. Xu. *Multilevel Finite Element Methods*. Lecutre Notes, 2004.