A BRIEF NOTE ON MULTIGRID METHODS

YIPING LU

luyiping9712@pku.edu.cn
School of Mathematic Science, Peking University.
No.5 Yiheyuan Road Haidian District,Beijing,P.R.China 100871.

This lecture note is based on Volker John's Lecture nots on multigrid and Jinchao Xu's review paper 'Iterative methods: by space decomposition and subspace correction'. This lecture note is the appendix of the project of **Introduction to applied math(2017Spring)** instructed by Prof. Jun Hu at Peiking Univ.

1. Detialed Investigation of Classical Iterative Schemes

1.1. **General Aspects of Classical Iterative Scheme.** To solve the linear system $Au = f$ we can give a general approach: let $A = M - N$, the iterative scheme can be formula as

$$u^* = M^{-1}Nu + M^{-1}f := Su + M^{-1}f$$
$$u^{(m+1)} = \omega u^* + (1-\omega)u^{(m)}$$

such that $u^{(m+1)} = (\omega S + (1-\omega)I)u^{(m)} + \omega M^{-1}f$ and we have residual equation $Se^{(m)} = e^{(m+1)}$

This scheme can conclude Damped Jacobi method and the SOR method.

1.2. **Converge Analysis.** First apply Discrete Fourier Method to analysis the scheme. For a give function b on $[0,1]$, we can expanded in the form

$$b(x) = \sum_{k=1}^{\infty} b_k \sin(k\pi x)$$

Here we can analyze $u^{(0)} = (u_1^{(0)}, \cdots, u_{N-1}^{(0)})^T, u_j^{(0)} = \sin(\frac{jk\pi}{N})(j,k = 1,\cdots,N-1)$ This discrete Fourier modes are also the **eigenvectors** of the matrix A.

- For $1 \leq k < N/2$ are called the low frequency or smooth modes.
- For $N/2 \leq k \leq N-1$ are called high frequency or oscillating modes.

There are several important observation

- On a fixed grid, there is a good damping of the high frequency errors whereas there is almost no damping of the low frequency errors.
- For a fixed wave number, the error is reduced on a coarser grid better than on a finer grid.
- The logarithm of the error decays linearly.

1.2.1. *damped Jacobi methods.* For damped Jacobi methods the iteration matrix is $S_{jac,\omega} = I - \omega D^{-1}A = I - \frac{\omega h}{2}A$, it has eigenvalue

$$\lambda_k(S_{jac,\omega}) = 1 - \frac{\omega h}{2}\lambda_k(A) = 1 - 2\omega\sin^2(\frac{k\pi h}{2})$$

It is easy to see that **damped Jacobi method converges fastest for** $\omega = 1$ which only need to solve a min-max problem and the error has the form $e^{(n)} = \sum_{k=1}^{N-1} c_k\lambda_k(S_{jac,\omega})w_k$

1.2.2. *SOR methods.* The first thing we need to calculate is the eigenvalue of S_{GS}:

$$\lambda_k(S_{GS}) = \cos^2(\frac{k\pi}{N})$$

Proof. Inserting the decomposition of S_{GS} gives

$$-(D+L)^{-1}Uw_k = \lambda_k(S_{GS})w_k \Leftrightarrow \lambda_k(S_{GS})(D+L)w_k = -Uw_k$$

Considering the model problem and inserting the representation of the k-th eigenvector

$$\lambda_k(S_{GS})\left[2\lambda_k(S_{GS})^{1/2}\sin(\frac{jk\pi}{N}) - \sin(\frac{(j-1)k\pi}{N})\right] = (\lambda_k(S_{GS}))^{(j+1)/2}\sin(\frac{(j+1)k\pi}{N})$$

if we let

$$\lambda_k(S_{GS}) = \cos^2(\frac{k\pi}{N})$$

It becomes a well-known relation

$$2\sin(\frac{\alpha+\beta}{2})\cos(\frac{\alpha-\beta}{2}) = \sin\alpha + \sin\beta$$

with $\alpha = (j+1)k\pi/N, \beta = (j-1)k\pi/N$ □

With the calculated eigenvalue the converge analysis is so naive that anyone could do it.

2. Multigrid Methods

Above all, I want to say the multigrid methods may not be the fastest method in solving the implicit scheme of $u_t = \Delta u$ but is suitable for Laplace equation $\Delta u = f$

2.1. **Grid Transfer.** We give some properties of the restriction and interpolation operator. I_{2h}^h, I_h^{2h}

- I_{2h}^h is full rank and the trivial kernel
- $I_h^{2h} = 2(I_h^{2h})^T$
- Dual Operator:$\left\langle I_{2h}^h v^{2h}, r^h \right\rangle_{V^h,(V^h)^*} = \left\langle v^2h, I_h^{2h}r^h \right\rangle_{V^{2h},(V^{2h})^*}$
- For the 1D model problem we have $A^{2h} = I_h^{2h}A^hI_{2h}^h$(Galerkin Projection)

2.2. **Two Level Methods.** The algorithm is give as

- $A^{2h}u^{2h} = f^{2h}$ compute an approximation v^{2h} and compute the residual $r^{2h} = f^{2h} - A^{2h}v^{2h}$
- Solve the coarse grid equation $A^h e^h = I_{2h}^h(r^{2h})$
- $v^h = v^h + I_h^{2h}(e^h)$

2.2.1. *Iteration Matrix.* Let S_{sm} be the iteration matrix of the smoother. The calculation can be seen at **Multilevel Method** in the next section. The iteration matrix can be given as

$$S_{2lev} = (I - I_h^{2h}(A^h)^{-1}I_{2h}^h A^h)S_{sm}$$

2.2.2. *Converge Analysis.* Here we give some definition used in the converge analysis

- **Smoothing property**

$$|||A^h S_{sm}||| \leq Ch^{-\alpha}$$

- **Approxiamation property**

$$|||(A^{2h})^{-1} - I_h^{2h}(A^h)^{-2}I_{2h}^h||| \leq C_a h^{\alpha}$$

Following the above two property we can give a converge speed to the algorithm as $|||S_{2lev}||| \leq CC_a$.For every specific algorithm used in the multi-grid framework, you should analyze the above two properties.

2.3. **Multigrid.** The W-multigrid can be analyzed by the processing before maybe it can be the appendix of my report for Numerical Algebra. The V-multigrid can be analyzed as a multi-level subspace correction algorithm.

3. Subspace Correction

Solving the linear equation can be seen as the following three steps:

- $r^{old} = f - Au^{old}$
- $\hat{e} = Br^{old}$ with $B \approx A^{-1}$
- $u^{new} = u^{old} + \hat{e}$

The choice of B is the core of this type of alogrithm. The point is to choice B by solving appropriate subspace problems. The subspaces are provided by a decomposition of V: $V = \sum_{i=1}^{J} V_i$. Here V_i are the subspaces of V. Assume $A_i : V_i \to V_i$ is restriction operator of A on V_i, then we can let $B = R_i \approx A_i^{-1}$ **Multigrid and domain decomposition methods can be viewed under this perspective.** In this section we only consider the linear iteration methods like $u^{k+1} = u^k + B(f - Au^k)$.

Here B is a approximate of the inverse matrix A^{-1} and the sufficient condition for the convergence of the scheme is

$$||I - BA||_A < 1$$

which can be seen in the appendix of the ppt for Iteration Methods.

3.1. **Subspace correction and subspace equations.** For subspace decomposition$V = \sum_{i=1}^{J} V_i$, for each i we define $Q_i, P_i : V \to V_i$ and $A_i : V_i \to V_i$ by

$$(Q_i u, v_i) = (u, v_i), (P_i u, v_i)_A = (u, v_i)_A, (A_i u_i, v_i) = (Au_i, v_i)$$

Here P_i, Q_i are both orthogonal projections and A_i is the restriction of A on V_i and is SPD. It follows the definition that $A_i u_i = f_i$ with $u_i = P_i u, f_i = Q_i f$ At the same time we use R_i to represent an approximate inverse of A_i in certain sense. Thus an approximate solution is given by $\hat{u}_i = R_i f_i$

Basic Idea: Consider the residual equation $Ae = r^{old}$ Instead of $u = u^{old} + e$ we solve the restricted equation to each subspace $A_i e_i = Q_i r^{old}$, while using the subspace solver R_i described earlier equally the process can be written as $\hat{e}_i = R_i Q_i r^{old}$

3.1.1. *PSC:Parallel Subspace Correction.* Similar to Jacobi Methods.(When $V = \sum span(\{e_i\})$, PSC becomes Jacobi)

An update of the approximation of u is obtained by

$$u^{new} = u^{old} + \sum_{i=1}^{J} \hat{e}_i$$

which can be equally written as

$$u^{new} = u^{old} + B(f - Au^{old})$$

where $B = \sum_{i=1}^{J} R_i Q_i$

Lemma. The operator B is SPD.

Proof.

$$(Bv, v) = \sum_{i=1}^{J} (R_i Q_i v, Q_i v) \geq 0$$

And the symmetry of B follows from the symmetry of R_i □

As a simple corollary, B can be used as a preconditioner liker CG methods. .(When $V = \sum span(\{e_i\})$, B becomes the simplest preconditioner $diag(a_{11}^{-1}, \cdots, a_{nn}^{-1})$

3.1.2. *SSC:Successive Subspace Correction.* Similar to Gauss-Seidel Methods.(When $V = \sum span(\{e_i\})$, SSC becomes G-S) This method is used as

$$v^1 = v^0 + R_1 Q_1 (f - Av^0)$$
$$v^2 = v^1 + R_2 Q_2 (f - Av^1)$$
$$\cdots$$

Formerly the algorithm can be written as $u^{(k+i)/J} = u^{(k+i-1)/J} + R_iQ_i(f - Au^{(k+i-1)/J})$

Let $T_i = R_iQ_iA$ Then we have

$$u - u^{(k+i)/J} = (I - T_i)(u - u^{(k+i-1)/J})$$

A successive application of this identity yields

$$u - u^{k+1} = E_J(u - u^k)$$

where $E_J = (I - T_J)(I - T_{J-1}) \cdots (I - T_1)$

Like SOR method we can also have an algorithm as$u^{(k+i)/J} = u^{(k+i-1)/J} + \omega R_iQ_i(f - Au^{(k+i-1)/J})$

3.1.3. *Multilevel Methods.* Multilevel algorithms are based on a nested sequence of subspaces

$$M_1 \subset M_2 \subset \cdots \subset M_J = V$$

Algorithm.

- Correction:$v^1 = \hat{B}_{k-1}\hat{Q}_{k-1}g$
- Smoothing:$\hat{B}_kg = v^1 + \hat{R}_k(g - \hat{A}_kv^1)$

Next we want to show that **the multilevel method is equivalent to the SSC algorithm.**

Suppose $M_k = \sum_{i=1}^k V_i$ It is easy to show that the two algorithm is equivalent.

3.2. **Converge Theory.**

- For PSC we need to estimate the condition number of $T = BA = \sum_{i=1}^J T_i$.
- For SSC we need to estimate $||E_J||_A < 1$

We define two parameters K_0, K_1 at the beginning of the section.

- For any $v = \sum_{i=1}^J v_i \in V$ we have $\sum_{i=1}^J (R_i^{-1}v_i, v_i) \leq K_0(Av, v)$
- For any u_i, v_i we have

$$\sum_{\{1,2,\cdots,J\}^2} (T_iu_i, T_jv_j) \leq K_1(\sum_{i=1}^J (T_iv_i, v_i)_A)^{1/2}(\sum_{j=1}^J (T_jv_j, v_j)_A)^{1/2}$$

3.2.1. *PSC.*

. **Theorem.** Assume that B is the SSC preconditioner then

$$\kappa(BA) \leq K_0K_1$$

Proof. Follow directly from the definition of K_1 that

$$||Tv||_A^2 = \sum_{i,j=1}^J (T_iv, T_jv)_A \leq K_1(Tv, v)_A \leq K_1 ||Tv||_A ||v||_A$$

which implies $\lambda_{\max}(BA) \leq K_1$

At the same time

$$
\begin{aligned}
(v, v)_A = \sum_{i=1}^{J}(v_i, P_i v)_A &\leq \sum_{i=1}^{J} \left(R_i^{-1} v_i, v_i\right)^{1/2} (R_i A_i P_i v_i, v)_A^{1/2} \\
&\leq \left(\sum_{i=1}^{J} \left(R_i^{-1} v_i, v_i\right)\right)^{1/2} \left(\sum_{i=1}^{J} (R_i A_i P_i v_i, v)_A\right)^{1/2} \\
&\leq \sqrt{K_0}\, ||v||_A\, (Tv, v)_A^{1/2}
\end{aligned}
$$

which implies $\lambda_{\min}(BA) \geq K_0$ and $\kappa(BA) \leq K_0 K_1$

□

3.2.2. *SSC.*

. For $E_i = (I - T_i)(I - T_{i-1}) \cdots (I - T_1)$ and $E_0 = I$ Then

$$
I - E_i = \sum_{j=1}^{i} T_j E_{j-1}
$$

Lemma.

$$
(2 - \omega_1) \sum_{i=1}^{J} (T_i E_{i-1} v, E_{i-1} v)_A \leq ||v||_A^2 - ||E_J v||_A^2
$$

Proof.

$$
\begin{aligned}
||E_{i-1}v||_A^2 - ||E_i v||_A^2 &= ||T_i E_i v||_A^2 + 2(T_i E_{i-1} v, E_i v)_A \\
&= (T_i E_{i-1} A, T_i E_{i-1} v)_A + 2(T_i (I - T_i) E_{i-1} v, E_{i-1} v)_A \\
&= ((2I - T_i) T_i E_{i-1} v, E_{i-1} v)_A \geq (2 - \omega_1)(T_i E_{i-1} v, E_{i-1} v)_A
\end{aligned}
$$

□

Theorem.

$$
||E_J||_A^2 \leq 1 - \frac{2 - \omega_1}{K_0 (1 + K_1)^2}
$$

Proof. First it is easy to show that

$$
\sum_{i=1}^{J} (T_i v, v)_A \leq (1 + K_1)^2 \sum_{i=1}^{J} (T_i E_{i-1} v, E_{i-1} v)_A
$$

At the same time we have

$$
\sum_{i=1}^{J} (T_i v, E_{i-1} v)_A \leq \left(\sum_{i=1}^{J} (T_i v, v)_A\right)^{1/2} \left(\sum_{i=1}^{J} (T_i E_{i-1} v, E_{i-1} v)_A\right)^{1/2}
$$

and

$$\sum_{i=1}^{J}\sum_{j=1}^{i=1}(T_i v, T_j E_{j-1} v)_A \le K_1 \left(\sum_{i=1}^{J}(T_i v, v)_A\right)^{1/2} \left(\sum_{i=1}^{J}(T_i E_{i-1} v, E_{i-1} v)_A\right)^{1/2}$$

Combining these three formulas leads to the theorem. □

At last I want to introduce a prefect websit:http://www.mgnet.org/