JSIAM Letters Vol.4 (2012) pp.41–44 ©2012 Japan Society for Industrial and Applied Mathematics JSIAM Letters

A parameter optimization technique for a weighted Jacobi-type preconditioner

Akira Imakura[1], Tetsuya Sakurai[1], Kohsuke Sumiyoshi[2] and Hideo Matsufuru[3]

[1] University of Tsukuba, 1-1-1, Tennodai Tsukuba, Ibaraki 305-8577, Japan
[2] Numazu College of Technology, Ooka 3600, Numazu, Shizuoka 410-8501, Japan
[3] High Energy Accelerator Research Organization, 1-1 Oho, Tsukuba, Ibaraki 305-0801, Japan

E-mail *imakura@ccs.tsukuba.ac.jp*

Received May 9, 2012, Accepted June 7, 2012

Abstract

The Jacobi preconditioner is well known as a preconditioner with high parallel efficiency to solve very large linear systems. However, the Jacobi preconditioner does not always show the great improvement of the convergence rate, because of the poor convergence property of the Jacobi method. In this paper, in order to improve the quality of the Jacobi preconditioner without loss its parallel efficiency, we introduce a weighted Jacobi-type preconditioner, and propose an optimization technique for the weight parameter. The numerical experiments indicate that the proposed preconditioner has higher quality and is more efficient than the traditional Jacobi preconditioner.

Keywords large linear systems, a weighted Jacobi-type preconditioner, parameter optimization, highly parallel computation

Research Activity Group Algorithms for Matrix / Eigenvalue Problems and their Applications

1. Introduction

In this paper, we consider the preconditioning techniques for the Krylov subspace methods to solve the very large, but sparse, linear systems of the form:

$$A\boldsymbol{x} = \boldsymbol{b}, \quad A \in \mathbb{R}^{n\times n}, \quad \boldsymbol{x}, \boldsymbol{b} \in \mathbb{R}^n, \tag{1}$$

where the coefficient matrix A is assumed to be nonsymmetric and nonsingular. These linear systems often arise from the discretization of the partial differential equations in the fields of the computational science and engineering.

Recent large scale simulations require to solve very large linear systems (1), and these are often one of the most time-consuming parts of the simulations. In this case, high parallel efficiency is recognized as extremely important more than high speed and/or high accuracy for the Krylov subspace methods and also preconditioning techniques.

The Jacobi preconditioner, which is the variable-type preconditioning technique using some iterations of the Jacobi method, especially shows the very high parallel efficiency, because the Jacobi method does not have any sequential operations like forward and/or backward substitution in each iteration. The Jacobi preconditioner also has some advantages for solving very large linear systems (1) that it is not required to construct any preconditioning matrices and it is available even if the coefficient matrix is used only as the matrix-vector formula.

However, since the Jacobi method has the strict convergence condition and the poor convergence property, the Jacobi preconditioner does not always show the great improvement of the convergence rate. In this paper, in order to improve the quality of the Jacobi preconditioner without loss of its parallel efficiency, we introduce a weighted Jacobi-type preconditioner, and propose an optimization technique for the weight parameter of the weighted Jacobi-type preconditioner.

This paper is organized as follows. In the next section, we briefly describe the Jacobi preconditioner and introduce a weighted Jacobi-type preconditioner. In Section 3, we propose a parameter optimization technique for the weighted Jacobi-type preconditioner. Then, we test the performance of the proposed preconditioner from some numerical experiments in Section 4 and finally we make some conclusions in Section 5.

2. The Jacobi preconditioner and a weighted Jacobi-type preconditioner

Preconditioning techniques play a very important role in improving the convergence rate of the Krylov subspace methods. They transform the linear systems (1) into more suitable systems for the Krylov subspace methods, i.e.,

$$K_1^{-1}AK_2^{-1}\boldsymbol{y} = K_1^{-1}\boldsymbol{b}, \quad \boldsymbol{x} = K_2^{-1}\boldsymbol{y},$$

where $K = K_1K_2$ is called as the preconditioning matrix, and is generally required to $K_1^{-1}AK_2^{-1} \approx I$ in some sense. For details we refer to [1] and references therein.

The incomplete factorization-type preconditioners, typified by the ILU(0) preconditioner, construct the preconditioning matrix $K \approx A$, such that the systems $K\boldsymbol{z} = \boldsymbol{w}$ appeared in each iterations of the Krylov subspace methods should be easy to solve, e.g., by using the incomplete LU decomposition

On the other hand, the variable-type preconditioners

roughly solve the system

$$A\boldsymbol{z} = \boldsymbol{w} \tag{2}$$

by an iterative method to obtain the approximation of $A^{-1}\boldsymbol{w}$, instead of solving $K\boldsymbol{z} = \boldsymbol{w}$. For solving (2), the stationary iterative methods, such as the Jacobi method, the Gauss-Seidel method and the SOR method, are widely used and reported their efficiency [2].

In what follows, we briefly describe and introduce two variable-type preconditioners: the Jacobi preconditioner and a weighted Jacobi-type preconditioner.

2.1 The Jacobi preconditioner

The stationary iterative methods are based on transforming (1) into a fixed-point equation:

$$\boldsymbol{x} = \boldsymbol{f}(\boldsymbol{x}),$$

and the solution of the linear system (1) is computed as the fixed-point of a vector-valued function $\boldsymbol{f}$ by an iterative method.

Let M be a nonsingular matrix and N be a matrix such that $A = M - N$. Then by setting $\boldsymbol{f}(\boldsymbol{x}) = M^{-1}N\boldsymbol{x} + M^{-1}\boldsymbol{b}$, the fixed-point (solution of (1)) can be computed from the recurrence formula

$$\boldsymbol{x}_{k+1} = \boldsymbol{f}(\boldsymbol{x}_k) = M^{-1}N\boldsymbol{x}_k + M^{-1}\boldsymbol{b}, \quad k = 0, 1, 2, \ldots$$

with an initial vector $\boldsymbol{x}_0$.

Let $G := M^{-1}N$ and $\rho(G)$ be the iteration matrix and its spectral radius. Then the stationary iterative methods converge to the exact solution $\boldsymbol{x}^* = A^{-1}\boldsymbol{b}$ for any initial vector $\boldsymbol{x}_0$ if and only if the spectral radius of the iteration matrix satisfies the inequality $\rho(G) < 1$. For the convergence rate of the stationary iterative methods, we also have the following relation:

$$\lim_{k\to\infty} \left(\max_{\boldsymbol{x}_0 \in \mathbb{R}^n} \frac{\|\boldsymbol{e}_k\|_2}{\|\boldsymbol{e}_0\|_2} \right)^{1/k} = \rho(G),$$

where $\boldsymbol{e}_0 := \boldsymbol{x}_0 - \boldsymbol{x}^*$ and $\boldsymbol{e}_k := \boldsymbol{x}_k - \boldsymbol{x}^*$ are the initial error vector and the error vector at the kth step, respectively [3].

The Jacobi method is the simplest stationary iterative method. Let $A_D, -A_L, -A_U$ be the diagonal part, the strict lower triangular part and the strict upper triangular part of A, respectively. Then the Jacobi method defines the matrices $M := A_D, N := A_L + A_U$, and its recurrence formula is given next.

$$\boldsymbol{x}_{k+1} = A_D^{-1}(A_L + A_U)\boldsymbol{x}_k + A_D^{-1}\boldsymbol{b}. \tag{3}$$

The Jacobi preconditioner is termed as the variable-type preconditioner using some iterations of the Jacobi method to solve (2). Since any sequential operations like forward and/or backward substitution does not exist in the recurrence formula (3), the Jacobi method and also the Jacobi preconditioner have high parallel efficiency.

2.2 A weighted Jacobi-type preconditioner

The Jacobi preconditioner has high parallel efficiency; however, the Jacobi preconditioner does not always show the great improvement of the convergence rate, because the spectrum radius of the iteration matrix of the Jacobi method is often $\rho(G) > 1$.

In this section in order to improve the quality of the Jacobi preconditioner, we introduce an improvement of the Jacobi method, which is named as a weighted Jacobi-type method. We also introduce a weighted Jacobi-type preconditioner.

The weighted Jacobi method is well known as an improvement of the Jacobi method, and its recurrence formula is shown by

$$\begin{aligned}\boldsymbol{x}_{k+1} &= \omega \left[A_D^{-1}(A_L + A_U)\boldsymbol{x}_k + A_D^{-1}\boldsymbol{b}\right] + (1-\omega)\boldsymbol{x}_k \\ &= \boldsymbol{x}_k + \omega A_D^{-1}(\boldsymbol{b} - A\boldsymbol{x}_k),\end{aligned}$$

where $\omega \in \mathbb{R}$ is called the weight parameter [3, Section 13.2]. This improvement method is also called the damped Jacobi method or the relaxed Jacobi method.

Here, we note that the high parallel efficiency of the (weighted) Jacobi method comes from the fact that the matrix M is a diagonal. From this observation, we can extend the (weighted) Jacobi method without loss of its high parallel efficiency as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + \omega D^{-1}(\boldsymbol{b} - A\boldsymbol{x}_k), \tag{4}$$

where $D \in \mathbb{R}^{n\times n}$ is any nonsingular diagonal matrix.

In this paper, we name the method based on recurrence formula (4) as a weighted Jacobi-type method. We also name the variable-type preconditioner using some iterations of the weighted Jacobi-type method to solve (2) as a weighted Jacobi-type preconditioner.

3. A parameter optimization technique

The weighted Jacobi-type preconditioner introduced in Section 2 may have a high potential to achieve a great improvement of the Krylov subspace method. However, the quality depends strongly on the weight parameter ω and also the diagonal matrix D.

In this section, we analyse the relationship between the weight parameter and the convergence rate of the weighted Jacobi-type method. Then from the analysis, we propose a parameter optimization technique for the weighted Jacobi-type preconditioner.

3.1 Convergence analysis on the weighted Jacobi-type method

The weighted Jacobi-type method can also be recognized as the stationary iterative method with the initial matrix partition

$$A = M_\omega - N_\omega, \quad M_\omega = \frac{1}{\omega}D, \quad N_\omega = \frac{1}{\omega}D - A,$$

and its iteration matrix can be written by

$$G_\omega := M_\omega^{-1}N_\omega = I - \omega D^{-1}A. \tag{5}$$

Therefore, the convergence rate of the weighted Jacobi-type method is based on $\rho(G_\omega)$.

Here, for the relationship between the weight parameter ω and the corresponding spectral radius $\rho(G_\omega)$, we derive the following theorem.

Theorem 3.1 *Let $A \in \mathbb{R}^{n\times n}$ be a nonsingular matrix, and $D \in \mathbb{R}^{n\times n}$ be a nonsingular diagonal matrix. We also let $C(\gamma, \rho)$ be the inner region of the circle with the center $\gamma \in \mathbb{R}$ and the radius $\rho \in \mathbb{R}$ on the complex plane,*

and the pair of γ^*, ρ^* be defined by

$$(\gamma^*, \rho^*) := \arg\min_{\gamma,\rho\in\mathbb{R}} \left|\frac{\rho}{\gamma}\right|$$

such that

$$\lambda_i(D^{-1}A) \in C(\gamma, \rho), \quad i = 1, 2, \ldots, n,$$

where $\lambda_i(D^{-1}A)$ are the eigenvalues of $D^{-1}A$.

Then we have

$$\arg\min_{\omega\in\mathbb{R}} \rho(G_\omega) = \frac{1}{\gamma^*}, \quad \min_{\omega\in\mathbb{R}} \rho(G_\omega) = \left|\frac{\rho^*}{\gamma^*}\right|. \tag{6}$$

Proof From (5), the spectral radius $\rho(G_\omega)$ can be rewritten as follows:

$$\begin{aligned}\rho(G_\omega) &= \rho(I - \omega D^{-1}A) \\ &= \max_i |1 - \omega\lambda_i(D^{-1}A)| \\ &= |\omega| \max_i |1/\omega - \lambda_i(D^{-1}A)|.\end{aligned}$$

Then, we have

$$\begin{aligned}\min_{\omega\in\mathbb{R}} \rho(G_\omega) &= \min_{\omega,\rho\in\mathbb{R}} |\omega\rho| \quad \text{s.t. } \lambda_i(D^{-1}A) \in C(1/\omega, \rho) \\ &= \min_{\gamma,\rho\in\mathbb{R}} \left|\frac{\rho}{\gamma}\right| \quad \text{s.t. } \lambda_i(D^{-1}A) \in C(\gamma, \rho).\end{aligned}$$

Therefore, (6) is proved.

(QED)

We also derive the following theorem for the convergence condition of the weighted Jacobi-type method with the optimized weight parameter.

Theorem 3.2 *Let $\omega_{\rm opt} := 1/\gamma^*$ be the optimized weight parameter of the weighted Jacobi-type method. Then, the spectral radius of the weighted Jacobi-type method with the optimized weight parameter $\omega_{\rm opt}$ satisfies the following inequality:*

$$\rho(G_{\omega_{\rm opt}}) = \left|\frac{\rho^*}{\gamma^*}\right| < 1 \tag{7}$$

if and only if $\lambda_i(D^{-1}A)$ satisfies

$$\mathcal{R}e(\lambda_i(D^{-1}A)) > 0, \quad i = 1, 2, \ldots, n \tag{8}$$

or

$$\mathcal{R}e(\lambda_i(D^{-1}A)) < 0, \quad i = 1, 2, \ldots, n. \tag{9}$$

Proof We firstly prove (8) or (9) $\Rightarrow$ (7). From the relation

$$\{z \in \mathbb{C} | \mathcal{R}e(z) > 0\} \subset \lim_{\gamma\to\infty} C(\gamma, \gamma),$$

$$\{z \in \mathbb{C} | \mathcal{R}e(z) < 0\} \subset \lim_{\gamma\to\infty} C(-\gamma, \gamma),$$

there exists a circle $C(\gamma, \rho)$ such that

$$\lambda_i(D^{-1}A) \in C(\gamma, \rho), \quad \left|\frac{\rho^*}{\gamma^*}\right| \le \left|\frac{\rho}{\gamma}\right| < 1$$

for any $\lambda_i(D^{-1}A)$ satisfying (8) or (9).

Next, we prove (7) $\Rightarrow$ (8) or (9). From (7), there is a circle $C(\gamma, \rho)$ such that

$$\lambda_i(D^{-1}A) \in C(\gamma, \rho), \quad \left|\frac{\rho}{\gamma}\right| < 1.$$

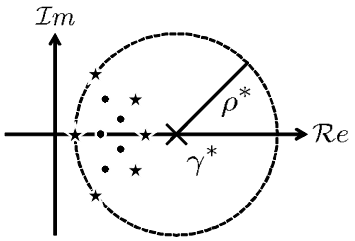

Fig. 1. Relationship between eigenvalues of $D^{-1}A$ and γ^*, ρ^*. The symbols ★ and • denote the extreme and interior eigenvalues, respectively.

We also have

$$C(\gamma, \rho) \subset \{z \in \mathbb{C} | \mathcal{R}e(z) > 0\},$$

$$C(-\gamma, \rho) \subset \{z \in \mathbb{C} | \mathcal{R}e(z) < 0\},$$

for any $\gamma, \rho \in \mathbb{R}, 0 < \rho < \gamma$. Therefore, if $\rho(G_{\omega_{\rm opt}}) < 1$, then all eigenvalues $\lambda_i(D^{-1}A)$ satisfy (8) or (9).

(QED)

Theorems 3.1 and 3.2 mean that the weight parameter ω of the weighted Jacobi-type method can be optimized by the extreme eigenvalues of $D^{-1}A$; see Fig. 1.

3.2 An optimization technique for the weight parameter of the weighted Jacobi-type preconditioner

Based on the results of Theorems 3.1 and 3.2, we propose an optimization technique for the weight parameter of the weighted Jacobi-type preconditioner.

The basic idea of our optimization technique is based on the so-called the off-line tuning, and it can be shown as follows:

Algorithm 1 A parameter optimization technique

1: **Initialization:**
Set an initial guess $\boldsymbol{x}_0$ and a diagonal matrix D.
2: **Optimization:**
a: Compute (approximately) the extreme eigenvalues of $D^{-1}A$.
b: Optimize (approximately) the weight parameter $\omega_{\rm opt}$ from the computed extreme eigenvalues.
3: **Application:**
Apply the weighted Jacobi-type preconditioned Krylov subspace method with the optimized weight parameter $\omega_{\rm opt}$ to the linear systems (1).

Here the extreme eigenvalues can be efficiently computed by some iterations of the Arnoldi method. Parallelization techniques of the Arnoldi method have been widely studied, and are implemented in the Parallel ARPACK [4].

In Algorithm 1, at the same time as optimizing $\omega_{\rm opt}$, we can obtain the corresponding spectral radius $\rho(G_{\omega_{\rm opt}})$. Therefore, if we have some choices for the diagonal matrix D, we can also select the best by applying the optimization with respect to each diagonal matrix D.

It is also expected that the optimization technique can be naturally extended to variable-type preconditioners using other weighted stationary iterative methods by almost the same way.

Table 1. Characteristics of the test problems.

Matrix name	n	Nnz	Application area
AF23560	23560	484256	Fluid dynamics
CHIPCOOL1	20082	281150	Model reduction
POISSON3DA	13514	352762	Fluid dynamics
XENON2	157464	3866688	Materials

4. Numerical experiments and results

In this section, we evaluate the performance of the proposed preconditioner, and compare it with the Jacobi preconditioner by test problems from [5].

The characteristics of the coefficient matrices of the test problems are shown in Table 1. The values n, Nnz denote the number of dimension and the number of nonzero elements, respectively. We set $\boldsymbol{b} = [1, 1, \ldots, 1]^{\mathrm{T}}$ as the right-hand side, $\boldsymbol{x}_0 = [0, 0, \ldots, 0]^{\mathrm{T}}$ for the initial guess, and the stopping criterion was set as $\|\boldsymbol{r}_k\|_2/\|\boldsymbol{b}\|_2 \leq 10^{-10}$.

For the proposed preconditioner, we set the diagonal matrix $D = A_D$ as well as the Jacobi preconditioner. The number of iterations of the Jacobi method and the weighted Jacobi-type method for the preconditioners are set as 20. The number of iterations of the Arnoldi method for optimization is also set as 20. We use the GMRES method for the Krylov subspace method.

The numerical experiments were implemented with the standard Fortran 77 in double precision arithmetic on the Intel Xeon X5550 (2.67GHz).

4.1 Numerical results

We present the numerical results in Table 2. In this table, a symbol † denotes that the method did not converge within 1000 iterations.

Firstly, we consider the relationship between the spectral radius $\rho(G)$ and number of iterations Iter. The spectral radius of the Jacobi preconditioner shows $\rho(G) > 1$, except for CHIPCOOL1. For these problems, the Jacobi preconditioned GMRES method shows the poor convergence property, because the convergence condition of the Jacobi method did not be satisfied.

On the other hand, by optimizing the weight parameter ω, the spectral radius of the proposed preconditioner satisfies $\rho(G) < 1$ for all test problems. Then the proposed preconditioner leads to the better convergence property than the Jacobi preconditioner.

Next we consider the computation time for the optimization t_{optimize} and the total computation time t_{total}. We can see from Table 2 that t_{optimize} is relatively smaller than t_{total} for all test problems. This is based on the fact that the most time-consuming part of the optimization is to compute the extreme eigenvalues by 20 iterations of the Arnoldi method. This computational cost is almost comparable with the cost of one iteration of the Jacobi and the weighted Jacobi-type preconditioned GMRES method.

From the better convergence and the negligibly small computation time for optimization, the proposed preconditioner can solve the linear systems with much smaller computation time than the Jacobi preconditioner.

Table 2. Convergence results (Precond: preconditioner, Iter : number of iterations, t_{optimize} : computation time for optimization, t_{total} : total computation time) of the preconditioned GMRES method.

AF23560

Precond	ω	$\rho(G)$	Iter	Time [sec.]	
				t_{optimize}	t_{total}
Non	−−	−−	†	−−	†
Jacobi	−−	6.491	†	−−	†
Proposed	0.046	0.999	660	6.30×10^{-2}	2.69×10^{1}

CHIPCOOL1

Precond	ω	$\rho(G)$	Iter	Time [sec.]	
				t_{optimize}	t_{total}
Non	−−	−−	†	−−	†
Jacobi	−−	0.994	45	−−	7.35×10^{-1}
Proposed	1.077	0.994	71	4.40×10^{-2}	1.23×10^{0}

POISSON3DA

Precond	ω	$\rho(G)$	Iter	Time [sec.]	
				t_{optimize}	t_{total}
Non	−−	−−	184	−−	7.77×10^{-1}
Jacobi	−−	1.259	239	−−	4.90×10^{0}
Proposed	0.885	0.998	33	4.40×10^{-2}	6.06×10^{-1}

XENON2

Precond	ω	$\rho(G)$	Iter	Time [sec.]	
				t_{optimize}	t_{total}
Non	−−	−−	†	−−	†
Jacobi	−−	2.154	†	−−	†
Proposed	0.634	0.999	386	5.06×10^{-1}	1.09×10^{2}

5. Conclusions

In this paper, in order to improve the quality of the Jacobi preconditioner without loss of its parallel efficiency, we have introduced the weighted Jacobi-type preconditioner, and proposed the optimization technique for the weight parameter of the preconditioner.

From our numerical experiments, we have learned that the proposed preconditioner has higher quality and is more efficiently than the traditional Jacobi preconditioner for solving very large but sparse linear systems.

For future work, we should apply the proposed preconditioner to the problems from the real applications and evaluate its efficiency in highly parallel computation.

Acknowledgments

This work is supported in part by Strategic Programs for Innovative Research Field 5 “The origin of matter and the universe”, CREST and KAKENHI (Grant Nos. 20105004, 20105005, 21246018, 22540296 and 23105702).

References

[1] M. Benzi, Preconditioning techniques for large linear systems: a survey, J. Comput. Phys., **182** (2002), 418–477.

[2] K. Abe and S. -L. Zhang, A variable preconditioning using the SOR method for GCR-like methods, Int. J. Numer. Anal. Mod., **2** (2005), 147–161.

[3] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., SIAM, Philadelphia, 2003.

[4] Parallel ARPACK, `http://www.caam.rice.edu/~kristyn/parpack_home.html`.

[5] The University of Florida Sparse Matrix Collection, `http://www.cise.ufl.edu/research/sparse/matrices/`.