Zhang, Int J Appl Exp Math 2017, 2: 118
http://dx.doi.org/10.15344/2456-8155/2017/118

International Journal of
Applied & Experimental Mathematics

Research Article Open Access

Special Issue: Krylov Subspace-Methods: Applications & Their Issues

A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides

Jianhua Zhang

School of Science, East China University of Technology, Nanchang, Jiangxi, 330013, China

Abstract

In this paper, we study and generalize a preconditioned modified Hermitian and skew-Hermitian splitting (PMHSS) iteration method for solving large sparse complex symmetric linear systems with multiple right-hand sides. Under suitable conditions, we show the new global iteration method is unconditionally convergent. Moreover, an inexact version which employs global conjugate gradient (Gl-CG) method or preconditioned Gl-CG (PGl-CG) as its inner process is constructed and its convergence property is analyzed. Finally, numerical experiments show the effectiveness and robustness of the new global iteration method and the PMHSS preconditioner, in comparison with other popular global Krylov subspace methods.

2000 MSC: 65F10, 65F50

Publication History:

Received: February 28, 2017
Accepted: July 13, 2017
Published: July 15, 2017

Keywords:

Global Krylov subspace method, Multiple right-hand sides, PMHSS iteration, Complex symmetric matrices

Introduction

In many scientific and engineering problems, such as electromagnetics scattering applications [19], finite-element discretizations of time-harmonic acoustic wave problem [13] and Maxwell's equation [7], we often require the solution of large complex symmetric linear systems with the same coefficient matrix and different right-hand sides

$$AX = B \tag{1}$$

where $A \in \mathbb{C}^{N\times N}$ is complex symmetric matrix, $X = [x^{(1)},\ldots,x^{(s)}] \in \mathbb{C}^{N\times s}$, $B = [b^{(1)},\ldots,b^{(s)}] \in \mathbb{C}^{N\times s}$, and $s << N$. We assume that $A = W+iT$, where $W, T \in \mathbb{R}^{N\times N}$ are positive semi-definite symmetric matrices, and at least one of them is positive definite, $i=\sqrt{-1}$ denotes the imaginary unit.

When memory is not a concern and the coefficient matrix can be factorized efficiently, the direct methods, such as LU decomposition [17] method, are very popular and efficient. However, the direct methods become expensive and prohibitive for large systems arising from discretizations of three-dimensional physical models. Therefore, iterative techniques, such as Krylov subspace methods, become popular for solving this class of complex symmetric linear systems. Recently, block methods have been developed to deal with such problems. In [7], Boyse and Seidl proposed the block QMR (Bl-QMR) method based on block Lanczos method. The Bl-QMR method is theoretically superior to the block biconjugate gradient (Bl-BCG) [15] method and often exhibits a nearly monotonically decreasing residual norm. Molhotra, Freund and Pinsky [13] have developed a J-symmetric variant of the Bl-QMR method that includes the complex symmetric version of Bl-QMR as a special case. Some suitable preconditioners and deflation techniques have been employed to enhance the convergence property. Simoncini and Gallopoulos [20] designed the block QMRCG-like and block QMR Lanczos methods based on the indefinite bilinear form. Block GMRES (Bl-GMRES) method [21] and its variants [1,8,10,14] were studied by some authors for solving the problem (1). Some other effective iteration methods, including global Krylov subspace methods and seed methods, have also been studied, see [9,11,12,16,18,22,23] for more details.

Recently, Bai et al.[4,5] considered a class of complex symmetric linear systems

$$Ax = (W + iT)x = b$$

where $x, b \in \mathbb{C}^N$, and $W, T \in \mathbb{R}^{N\times N}$ are real, symmetric and positive semidefinite matrices with at least one of them, e.g., being positive definite. W By making use of the special structure of the coefficient matrix Ax, Bai et al. [4] developed the modified Hermitian and Skew-Hermitian splitting (MHSS) iteration method based on the HSS [2] iteration method. To accelerate convergence rate of the MHSS iteration method, they established the PMHSS [5] iteration method. Under suitable conditions, the PMHSS iteration method converges unconditionally. Furthermore, numerical experiments have shown the PMHSS iteration method and PMHSS preconditioner can lead to better computing efficiency than other iteration methods.

In this paper, motivated by the advantage of the PMHSS iteration method [5], we derive a new iteration method for solving the problem (1). This iteration method is a matrix variant of the PMHSS iteration method [5], named as the global PMHSS (Gl-PMHSS) method. Analogous to the PMHSS iteration method, the problem of (1) can be decomposed two linear sub-systems with real and symmetric positive definite coefficient matrices. Moreover, like the PMHSS iteration method, we show the Gl-PMHSS iteration method also converges unconditionally under the condition that both W and T are symmetric positive semi-definite matrices, at least, one of them is positive definite matrix. An upper bound on the contraction factor of the Gl-PMHSS iteration method and the optimal value of the iteration parameter α are analyzed. In addition, the inexact Gl-PMHSS (IGl-PMHSS) iteration method which employs the Gl-CG method or PGl-CG method as its inner iteration is established, and its convergence property is also studied in detail. Numerical experiments show the effectiveness and robustness of the Gl-PMHSS iteration method and the IGl-PMHSS iteration method. The PMHSS preconditioner

[*]**Corresponding Author:** Dr. Jianhua Zhang, School of Science, East China University of Technology, Nanchang, Jiangxi, 330013, China, E-mail: zhangjhnuaa@163.com

Citation: Zhang J (2017) A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides. Int J Appl Exp Math 2: 118. doi: http://dx.doi.org/10.15344/2456-8155/2017/118

Citation: Zhang J (2017) A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides. Int J Appl Exp Math 2: 118. doi: http://dx.doi.org/10.15344/2456-8155/2017/118

combined with global Krylov subspace methods such as global GMRES (Gl-GMRES) method, global BiCGSTAB (Gl-BiCGSTAB) method also demonstrates mesh-independent and parameter insensitive convergence properties.

The organization of the paper is structured as follows. In section 2, the Gl-PMHSS iteration method is established and its convergence properties are analyzed. In section 3, we propose the inexact Gl-PMHSS iteration method and study its implementation and convergence properties. Section 4 is devoted to numerical experiments. Finally, we make some conclusions and remarks in section 5.

Methodology

This research is studied the wave height over the gulf of Thailand using the SWAN model during 15 November 2013 at 0000 UTC to 18 November 2013 at 0000 UTC as shown in Figure 2. It covers

$$(\alpha I+W)X=(\alpha I-iT)X+B$$

and

$$(\alpha I+T)X=(\alpha I+iW)X-iB$$

Next, we assume that the matrix $V\in\mathbb{R}^{N\times N}$ be symmetric positive definite and define

$$\tilde{W}=V^{-\frac{1}{2}}WV^{-\frac{1}{2}},\tilde{T}=V^{-\frac{1}{2}}TV^{-\frac{1}{2}},\tilde{A}=V^{-\frac{1}{2}}AV^{-\frac{1}{2}},\tilde{X}=V^{\frac{1}{2}}X,\tilde{B}=V^{-\frac{1}{2}}B$$

Therefore, the problem (1) can be equivalently transformed into

$$\tilde{A}\tilde{X}=\tilde{B}$$

where $\tilde{A}=\tilde{W}+i\tilde{T}\in\mathbb{C}^{N\times N}$, and $\tilde{W},\tilde{T}\in\mathbb{R}^{N\times N}$ are real, symmetric and positive semidefinite matrices, with $\tilde{W}$ being positive definite.

Applying the same approach suggested in [5], we have the following two new fixed-point systems which described as follows:

$$(\alpha V+W)X=(\alpha V-iT)X+B$$

and

$$(\alpha V+T)X=(\alpha V+iW)X-iB$$

Based on the above results, we can easily give the global PMHSS (Gl-PMHSS) iteration method as follows.

Algorithm 1: The Gl-PMHSS iteration method

1. Choose $X^{(0)}\in\mathbb{C}^{N\times s}$ be an arbitrary initial guess;

2. For K = 0,1,2,... until $\{X^{(k)}\}_{k=0}^{\infty}\subset\mathbb{C}^{N\times s}$ converges;

3. Compute

$$\begin{cases}(\alpha V+W)X^{(k+\frac{1}{2})}=(\alpha V-iT)X^{(k)}+B,\\(\alpha V+T)X^{(k+1)}=(\alpha V+iW)X^{(k+\frac{1}{2})}-iB,\end{cases}$$

where α is a given positive constant and $V\in\mathbb{R}^{N\times N}$ is a prescribed symmetric positive definite matrix;

Because $V,W\in\mathbb{R}^{N\times N}$ are symmetric positive definite matrices, $T\in\mathbb{R}^{N\times N}$ is symmetric positive is symmetric positive semidefinite matrix and α is real positive constant, thus the matrices $\alpha V+W$ and $\alpha V+T$ are both symmetric positive definite. This implies that it is possible to solve the two linear sub-systems at each step of the Gl-PMHSS iteration by direct methods or global conjugate gradient (Gl-CG) [18] method.

After applying the kronecker product and straight forward derivations, we can reformulate the iteration scheme (8) into the standard form

$$x^{(k+1)}=\hat{L}(V;\alpha)x^{(k)}+R(V;\alpha)b,k=0,1,2,\ldots$$

where

$$\hat{L}(V;\alpha)=(\alpha V+T)^{-1}(\alpha V+iW)(\alpha V+W)^{-1}(\alpha V-iT)$$

and

$$\hat{R}(V;\alpha)=(1-i)\alpha(\alpha V+T)^{-1}V(\alpha V+W)^{-1}$$

Note that $\hat{V}=I\otimes V$ and $\hat{T}=I\otimes T$. Using the result introduced in [5], the coefficient matrix can be splitted into

$$A=F(V;\alpha)-G(V;\alpha)$$

where $F(V;\alpha)=\frac{1+i}{2\alpha}(\alpha V+W)V^{-1}(\alpha V+T)$ and $G(V;\alpha)=\frac{1+i}{2\alpha}$, $G(V;\alpha)=\frac{1+i}{2\alpha}(\alpha V+iW)V^{-1}(\alpha V-iT)$. Therefore, the Gl-PMHSS iteration scheme is induced by the matrix splitting (12). Moreover, the splitting matrix $F(V;\alpha)$ can be used as a preconditioner for the complex symmetric matrix $A\in\mathbb{C}^{N\times N}$, referred as the PMHSS preconditioner.

Using the Theorem 2.1 and Corollary 2.1 in [4], we can prove the convergence property of the Gl-PMHSS iteration method for solving the problem (1).

Theorem 1. Assume that the matrices $W\in\mathbb{R}^{N\times N}$ and $T\in\mathbb{R}^{N\times N}$ be symmetric positive definite and symmetric positive semidefinite, respectively. Let α be a positive constant and $V\in\mathbb{R}^{N\times N}$ be a symmetric positive definite matrix. Then the spectral radius $\rho(\hat{L}(V;\alpha))$ of the Gl-PMHSS iteration matrix $\hat{L}(\hat{V};\alpha)=(\alpha\hat{V}+\hat{T})^{-1}(\alpha\hat{V}+i\hat{W})(\alpha\hat{V}+\hat{W})^{-1}(\alpha\hat{V}-i\hat{T})$ is bounded by $\sigma(\alpha)\equiv\max_{\tilde{\lambda}_j\in sp(V^{-1}W)}\frac{\sqrt{\alpha^2+\tilde{\lambda}_j^2}}{\alpha+\tilde{\lambda}_j}$ where $Sp(V^{-1}W)$ denotes the spectrum of the matrix $V^{-1}W$. Therefore, it holds that $\rho(\hat{L}(V;\alpha))\le\sigma(\alpha)<1,\ \forall\alpha>0$.

Proof. Applying the kronecker product, we can rewrite the Gl-PMHSS iteration as follows

$$\begin{cases}(\alpha(I\otimes V)+I\otimes W)x^{(k+\frac{1}{2})}=(\alpha(I\otimes V)-i(I\otimes T))x^{(k)}+b,\\(\alpha(I\otimes V)+I\otimes T)x^{(k+1)}=(\alpha(I\otimes V)+i(I\otimes W))x^{(k+\frac{1}{2})}-ib,\end{cases}$$

which can be described equivalently as

$$\begin{cases}(\alpha\hat{V}+W)x^{(k+\frac{1}{2})}=(\alpha V-iT)x^{(k)}+b,\\(\alpha\hat{V}+T)x^{(k+1)}=(\alpha V+iW)x^{(k+\frac{1}{2})}-ib,\end{cases}$$

where $\hat{V}=I\otimes V$, $\hat{W}=I\otimes W$, $\hat{T}=I\otimes T$, It is easy to see the iteration scheme (14) is the PMHSS iteration method for solving the system of complex symmetric linear equations $\mathbb{A}x=b$, with $\mathbb{A}=W+iT$. After simple computations, we have

$$x^{(k+1)}=\hat{L}(V;\alpha)x^{(k)}+R(V;\alpha)b,k=0,1,2,\ldots$$

We can easily verify that both $\hat{W}$ and $\hat{T}$ are Hermitian matrices. Moreover, when either $W\in\mathbb{C}^{N\times N},V\in\mathbb{C}^{N\times N}$ or $T\in\mathbb{C}^{N\times N}$ is positive definite, the matrix $\hat{W}\in\mathbb{C}^{Ns\times Ns},V\in\mathbb{C}^{Ns\times Ns}$ or $\hat{T}\in\mathbb{C}^{Ns\times Ns}$ is also positive definite. We also have

Citation: Zhang J (2017) A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides. Int J Appl Exp Math 2: 118. doi: http://dx.doi.org/10.15344/2456-8155/2017/118

$$\hat{L}(V;\alpha)=(\alpha I+V^{-1}T)^{-1}(\alpha I+iV^{-1}W)(\alpha I+V^{-1}W)^{-1}(\alpha I-iV^{-1}T)$$

Therefore, by making use of Theorem 2.1 in [4], we show that the PMHSS iteration method (14) converges unconditionally to the exact solution $x^*\in\mathbb{C}^{Ns}$ of the complex symmetric linear systems $\mathbb{A}x=b$, with the convergence factor being $\rho(\hat{L}(V;\alpha))$. We also obtain that $\rho(\hat{L}(V;\alpha))\le\sigma(\alpha)<1,\forall\alpha>0$. Analagously to the Theorem 2.1 in [6], we can show that the Gl-PMHSS iteration method for the problem (1) also converges unconditionally to the exact solution $X^*\in\mathbb{C}^{N\times s}$, with the convergence factor $\rho(\hat{L}(V;\alpha))$ being bounded by $\sigma(\alpha)$.

Next, using the Corollary 2.1 in [4], we can easily formulate the following Corollary.

Corollary 1. Suppose that the conditions of Theorem 1 be satisfied, $\tilde{\eta}_{\min}$ and $\tilde{\eta}_{\max}$ be the lower and the upper bounds of the eigenvalues of the symmetric positive definite matrix $V^{-1}W\in\mathbb{R}^{N\times N}$, respectively. Then

$$\alpha_*\equiv\arg\min_{\alpha}\left\{\max_{\tilde{\eta}_{\min}\le\tilde{\lambda}\le\tilde{\eta}_{\max}}\frac{\sqrt{\alpha^2+\tilde{\lambda}^2}}{\alpha+\tilde{\lambda}}\right\}=\sqrt{\tilde{\eta}_{\min}\tilde{\eta}_{\max}}$$

and

$$\alpha(\alpha_*)=\frac{\sqrt{\tilde{\eta}_{\min}+\tilde{\eta}_{\max}}}{\sqrt{\tilde{\eta}_{\min}}+\sqrt{\tilde{\eta}_{\max}}}=\frac{\sqrt{k(V^{-1}W)+1}}{\sqrt{k(V^{-1}W)}+1}$$

where $k(V^{-1}W)=\dfrac{\tilde{\eta}_{\max}}{\tilde{\eta}_{\min}}$ is the spectral condition number of the matrix $V^{-1}W$.

Proof. The proof is similar to that of Corollary 2.1 in [4], hence it is omitted.

The inexact Gl-PMHSS iteration method

The Gl-PMHSS method is a two-step iteration scheme which requires the exact solution of two symmetric positive definite systems with matrices $\alpha V+W$ and $\alpha V+T$. This may be costly and impractical for solving the complex symmetric system arising from the discretization of a three-dimensional partial differential equation. Inspired by the advantages of inexact HSS iteration method [2,3], to enhance the computational efficiency of the Gl-PMHSS method, we can solve the two linear systems inexactly by empolying suitable iteration methods, such as the block SOR, Gl-GMRES and Gl-CG methods. This results in the inexact Gl-PMHSS (IGl-PMHSS) iteration method for solving the problem (1).

To simplify numerical implementation and convergence analysis, we can state the IGl-PMHSS iteration method as follows.

Algorithm 2. The IGl-PMHSS iteration method

1. Choose an initial guess $X_0\in\mathbb{C}^{N\times s}$
2. For $K=0,1,2,\cdots$ until $\left\{X^{(k)}\right\}_{k=0}^{\infty}\subset\mathbb{C}^{N\times s}$ converges;
3. Compute $R^{(k)}=B-AX^{(k)}$
4. Approximate the solution of $(\alpha V+W)Z^{(k)}=R^{(k)}$ by iteration method such that the residual $P^{(k)}=R^{(k)}-(\alpha V+W)Z^{(k)}$ satisfies $\left\|P^{(k)}\right\|_F\le\varepsilon_k\left\|R^{(k)}\right\|_F$,
5. Compute $X^{(k+\frac{1}{2})}=X^{(k)}+Z^{(k)}$
6. Compute $R^{(k+\frac{1}{2})}=B-AX^{(k+\frac{1}{2})}$
7. Approximate the solution of $(\alpha V+T)Z^{(k+\frac{1}{2})}=R^{(k+\frac{1}{2})}$ by iteration method such that the residual $Q^{(k+\frac{1}{2})}=R^{(k+\frac{1}{2})}-(\alpha V+T)Z^{(k+\frac{1}{2})}$ satisfies $\left\|Q^{(k+\frac{1}{2})}\right\|_F\le\eta_k\left\|R^{(k+\frac{1}{2})}\right\|_F$
8. Compute $X^{(k+1)}=X^{(k+\frac{1}{2})}-iZ^{(k+\frac{1}{2})}$
9. End for.

Next, we will analyze the convergence properties about the above IGl-PMHSS method based on Theorems 3.1 and 3.2 in [6].

Theorem 2. Let the assumptions in Theorem 1 be satisfied. Suppose that the $\left\{X^{(k)}\right\}_{k=0}^{\infty}\subseteq\mathbb{C}^{N\times s}$ be an iterative sequence generated by the IGl-PMHSS method and $X^*\in\mathbb{C}^{N\times s}$ be the exact solution of the problem (1), then we have

$$\left\|X^{(k+1)}-X^*\right\|_M\le(\sigma(\alpha)+\tilde{\theta}\tilde{\rho}\eta_k)(1+\tilde{\theta}\varepsilon_k)\left\|X^{(k)}-X^*\right\|_M, k=0,1,2,\ldots$$

where the norm $\|\bullet\|_M$ is defined by $\|Y\|_M=\|(\alpha V+T)Y\|_F$ for any matrix $Y\in\mathbb{C}^{N\times s}$, $\tilde{\rho}$, and $\tilde{\theta}$ are computed as

$$\tilde{\rho}=\left\|\left(\alpha\hat{V}+T\right)\left(\alpha V+W\right)^{-1}\right\|_2$$

and

$$\tilde{\theta}=\left\|\left(I\otimes A\right)\left(\alpha\hat{V}+T\right)^{-1}\right\|_2$$

respectively. In particular, if

$$\left(\sigma(\alpha)+\tilde{\theta}\tilde{\rho}\eta_{\max}\right)\left(1+\tilde{\theta}\varepsilon_{\max}\right)<1$$

then the iteration sequence $\left\{X^{(k)}\right\}_{k=0}^{\infty}$ converges to the exact solution $X^*\in\mathbb{C}^{N\times s}$, where $\varepsilon_{\max}=\max_k\{\varepsilon_k\}$ and $\eta_{\max}=\max_k\{\eta_k\}$

Proof. The proof is similar to that of Theorem 3.1 in [6], hence it is omitted.

We remark that it is not necessary for $\{\varepsilon_k\}$ and $\{n_k\}$ to approach to zero as k is increasing. Using the Theorem 2, when the condition (17) is satisfied, we can guarantee the convergence of the IGl-PMHSS iteration method. According to the following Theorem 3, we can propose one possible way of choosing the inner iteration tolerances $\{\varepsilon_k\}$ and $\{n_k\}$ such that the computational work is minimized and the original convergence rate of the Gl-PMHSS iteration can be asymptotically recovered.

Theorem 3. Assume that the the conditions in Theorem 2 be satisfied, let both $\{\tilde{\tau}_1(k)\}$ and $\{\tilde{\tau}_2(k)\}$ be nondecreasing and positive sequences satisfying $\tilde{\tau}_1(k)\ge1,\tilde{\tau}_2(k)\ge1$ and $\limsup_{k\to\infty}\tilde{\tau}_1(k)=\lim_{k\to\infty}$ sup $\tilde{\tau}_2(k)=+\infty$ and that both be real constants satisfying and ,where and are nonnegative constants. Then it follows that $0<\delta_1,\delta_2<1$ be constants satisfying $\varepsilon_k\le\tilde{c}_1\delta_1^{\tilde{\tau}_1(k)}$ and $\eta_k\le\tilde{c}_2\delta_1^{\tilde{\tau}_2(k)},k=0,1,2,\ldots$, where $\tilde{c}_1$ and $\tilde{c}_2$ are nonnegative constants. Then it follows that

$$\left\|X^{(k+1)}-X^*\right\|_S\le\left(\sqrt{\sigma(\alpha)}+\omega\theta\delta^{\tilde{\tau}(k)}\right)^2\left\|X^{(k)}-X^*\right\|_S$$

Citation: Zhang J (2017) A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides. Int J Appl Exp Math 2: 118. doi: http://dx.doi.org/10.15344/2456-8155/2017/118

where ρ and θ are computed in (15) and (16), $\tilde{\tau}(k)$ and δ are defined by $\tilde{\tau}(k)=\min\{\tilde{\tau}_1(k),\tilde{\tau}_2(k)\}$ and $\delta=\max\{\delta_1,\delta_2\}$ and

$$\omega=\max\left\{\sqrt{\tilde{c}_1\tilde{c}_2\rho},\frac{1}{2\sqrt{\sigma(\alpha)}}(\tilde{c}_1\sigma(\alpha)+\tilde{c}_2\rho)\right\}$$

In particular, it holds that $\limsup_{k\to\infty}\frac{\|X^{(k+1)}-X^*\|_S}{\|X^{(k)}-X^*\|_S}\le\sigma(\alpha)$, i.e. the convergence rate of the IGl-PMHSS iteration method is asymptotically the same as that of the Gl-PMHSS iteration method.

Proof. The proof is similar to that of Theorem 3.2 in [6], hence it is omitted.

Numerical Examples

In this section, we perform some test problems from [4,5] to assess the effectiveness and feasibility of the Gl-PMHSS iteration method, when it is employed either as a solver or as a preconditioner for solving the complex symmetric linear systems with multiple right-hand sides. All computations are carried out using double precision floating point arithmetic in MATLAB (version R2010b) with a PC-Intel (R) Core (TM)2 Duo CPU T6570 2.10 GHz, and 2GB RAM. We choose the initial guess X_0=*Zeros(N, s)* and set the right-hand side *B=A*rand(N, s), s=5* where function **rand** creates an $N \times s$ random matrix with coefficients uniformly distributed in [0,1]. Note that Its and CPU denote iterations and CPU-time for computing approximation, respectively. Let the stopping criterion be $\frac{\|R_k\|_F}{\|R_0\|_F}\le 1.e-6$. Gl-GMRES and Gl-GMRES(*) [11] denote the unrestarted global GMRES method and its restarted method, respectively. We only consider the right preconditioner, i.e. $M_1 = I$ and $M_2 = M$ to enhance convergence behavior. We adopt the PMHSS preconditioner defined by

$$F(\alpha)=\frac{(\alpha+1)(1+i)}{2\alpha}(\alpha W+T)$$

As suggested in [5], we choose the **symamd.m** ordering algorithm in actual implementations of the PMHSS preconditioner. Like the PMHSS iteration method in [5], we choose the optimal parameter α for the Gl-PMHSS iteration by performing numerical experiments; for more details, see [4,5]. We denote α_{exp} as the optimal iteration parameters in this section. A symbol ``--" is used to indicate that the method does not obtain the required stopping criterion before maximum iterations or out of memory.

Example 1. In this example, we choose the matrix which arises from Example 4.1 in [5], and the problem of (1) is of the form

$$[(K+\frac{3-\sqrt{3}}{\tau}I)+i(K+\frac{3+\sqrt{3}}{\tau}I)]X=B$$

For more details, we refer to [4] and the references therein. Tables 1 and 2 report the numerical results.

From Table 1 we observe that the iteration counts of the Gl-GMRES and Gl-GMRES(20) methods increase rapidly with problem size, but that of Gl-PMHSS iteration method almost remains constant. Therefore, we can conclude the Gl-PMHSS iteration method is also almost independent of the problem size. Furthermore, as a solver, the Gl-PMHSS iteration method demonstrates the best convergence behavior than the Gl-GMRES and Gl-GMRES(20) methods in terms of iteration steps and CPU times.

Method	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-PMHSS	α_{exp}	1.09	1.50	1.52	1.31	1.48
	Its	22	22	23	23	23
	CPU	0.029	0.113	0.751	4.951	26.572
Gl-GMRES	Its	55	104	197	364	—
	CPU	0.279	5.701	43.309	1120.1	—
Gl-GMRES(20)	Its	75	259	713	2337	—
	CPU	0.226	3.041	16.647	413.08	—

Table 1: Its and CPU for Gl-PMHSS, Gl-GMRES and Gl-GMRES(20) methods for Example 1.

From Table 2 we see that the PMHSS preconditioner drastically reduces iteration steps and CPU times of the Gl-GMRES and Gl-GMRES(10) methods. Moreover, the iteration steps of the PMHSS preconditioner is almost constant, and then the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods demonstrate h-independent convergence behavior. Setting the iteration parameter to be 1, we see that iteration counts for the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods are almost identical to those obtained with the experimentally found optimal parameter α_{exp} . In addition, the PMHSS-preconditioned Gl-BiCGSTAB shows the same convergence properties described as above, and it requires less Its and CPU than the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods.

Example 2. In this example, we use the matrix which arises from Example 4.2 in [5], and the problem of (1) is of the form

$$[(-\omega^2 M+K)+i(\omega C_V+C_H)]X=B$$

For more details, we refer to [4] and the references therein. Tables 3 and 4 report the numerical results.

As observed from Table 3, we also see that the iteration counts of the Gl-GMRES and Gl-GMRES(20) methods increase rapidly with problem size, but that of Gl-PMHSS iteration method almost remains constant. Therefore, we can conclude the Gl-PMHSS iteration method is also almost independent of the problem size. Furthermore, as a solver, the Gl-PMHSS iteration method gives the best convergence results than the Gl-GMRES and Gl-GMRES(20) methods in terms of iteration steps and CPU times.

From Table 4 we observe that the PMHSS preconditioner shows high quality and drastically reduces iteration steps and CPU times of the Gl-GMRES and Gl-GMRES(10) methods. Again, the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods still demonstrate h-independent convergence behavior. Moreover, setting the iteration parameter α to be 1 can lead to nearly optimal numerical results obtained with the experimentally found optimal parameter α_{exp}. As before, the PMHSS-preconditioned Gl-BiCGSTAB shows the same convergence properties described above, and it can compete with or be superior to the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods.

Example 3. In this example, we use the matrix which arises from Example 4.3 in [5], and the problem of (1) is of the form

$$(W+iT)X=B$$

where $W=10(I\otimes V_c+V_c\otimes I)+9(e_1e_m^T+e_me_1^T)\otimes I$ $T=I\otimes$, $T=I\otimes$, $V_c=V-e_1e_m^T-e_me_1^T\in\mathbb{R}^{m\times m}$. For more details, we refer

Citation: Zhang J (2017) A New Global Method for Solving Complex Symmetric Linear Systems with Multiple Right-hand Sides. Int J Appl Exp Math 2: 118. doi: http://dx.doi.org/10.15344/2456-8155/2017/118

Method	preconditioner	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-GMRES	PMHSS	α_{exp}	1.47	1.04	0.69	0.93	1.41
		Its	6	6	6	7	7
		CPU	0.011	0.036	0.199	1.466	7.295
	PMHSS	α	1	1	1	1	1
		Its	6	6	7	7	7
		CPU	0.012	0.037	0.242	1.499	7.342
Gl-GMRES(10)	PMHSS	α_{exp}	0.67	0.76	0.74	0.56	1.28
		Its	6	6	6	7	7
		CPU	0.012	0.036	0.197	1.465	7.287
	PMHSS	α	1	1	1	1	1
		Its	6	6	7	7	7
		CPU	0.013	0.040	0.257	1.517	7.331
Gl-BICGSTAB	PMHSS	α_{exp}	0.54	0.86	1.67	0.76	1.02
		Its	3	4	4	4	4
		CPU	0.005	0.026	0.159	0.941	5.051
	PMHSS	α	1	1	1	1	1
		Its	3	4	4	4	4
		CPU	0.009	0.033	0.166	0.942	5.119

Table 2: Its and CPU for preconditioned Gl-GMRES, Gl-GMRES(10) and Gl-BiCGSTAB methods for Example 1.

Method	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-PMHSS	α_{exp}	1.09	1.50	1.52	1.31	1.48
	0.68	0.89	0.86	0.88	0.84	23
	Its	34	35	36	36	36
Gl-GMRES	CPU	0.044	0.187	1.262	7.129	41.664
	Its	44	82	158	306	—
Gl-GMRES(20)	CPU	0.157	3.144	42.299	800.20	—
	Its	56	160	525	1815	—
	CPU	0.189	1.337	19.987	327.389	—

Table 3: Its and CPU for Gl-PMHSS, Gl-GMRES and Gl-GMRES(20) methods for Example 2

Method	preconditioner	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-GMRES	PMHSS	α_{exp}	8.34	5.35	3.71	2.30	3.31
		Its	8	8	8	7	7
		CPU	0.014	0.050	0.294	1.476	7.049
	PMHSS	α	1	1	1	1	1
		Its	8	8	8	8	8
		CPU	0.018	0.054	0.331	1.786	8.772
Gl-GMRES(10)	PMHSS	α_{exp}	3.60	4.94	2.20	1.79	2.91
		Its	8	8	8	7	7
		CPU	0.016	0.051	0.284	1.491	7.260
	PMHSS	α	1	1	1	1	1
		Its	8	8	8	8	8
		CPU	0.019	0.057	0.378	1.727	8.771
Gl-BICGSTAB	PMHSS	α_{exp}	7.91	5.72	2.84	4.50	3.20
		Its	5	5	5	5	4
		CPU	0.010	0.033	0.196	1.067	5.070
	PMHSS	α	1	1	1	1	1
		Its	5	5	5	5	5
		CPU	0.016	0.037	0.207	1.094	6.495

Table 4: Its and CPU for preconditioned Gl-GMRES, Gl-GMRES(10) and Gl-BiCGSTAB methods for Example 2.

Method	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-PMHSS	α_{exp}	0.54	0.63	0.62	0.68	0.89
	0.68	32	32	33	34	36
	Its	0.046	0.217	1.639	11.177	60.287
Gl-GMRES	CPU	32	50	77	113	—
	Its	0.082	0.718	6.856	111.945	—
Gl-GMRES(20)	CPU	34	60	98	155	254
	Its	0.123	0.309	2.370	27.437	192.087
	CPU	0.189	1.337	19.987	327.389	—

Table 5: Its and CPU for Gl-PMHSS, Gl-GMRES and Gl-GMRES(20) methods for Example 3.

Method	preconditioner	m × m	16 × 16	32 × 32	64×64	128×128	256×256
Gl-GMRES	PMHSS	α_{exp}	6.35	7.06	4.10	2.62	2.80
		Its	6	7	8	10	12
		CPU	0.011	0.050	0.364	2.925	16.724
	PMHSS	α	1	1	1	1	1
		Its	6	7	9	10	12
		CPU	0.013	0.054	0.428	2.936	17.012
Gl-GMRES(10)	PMHSS	α_{exp}	7.44	6.51	4.85	2.24	3.20
		Its	6	7	8	10	12
		CPU	0.011	0.051	0.363	2.876	16.502
	PMHSS	α	1	1	1	1	1
		Its	6	7	9	10	13
		CPU	0.014	0.056	0.443	2.916	17.570
Gl-BICGSTAB	PMHSS	α_{exp}	6.37	7.08	4.63	2.43	1.86
		Its	3	4	5	6	7
		CPU	0.008	0.034	0.281	1.789	12.851
	PMHSS	α	1	1	1	1	1
		Its	4	4	5	7	7
		CPU	0.012	0.041	0.335	2.178	13.301

Table 6: Its and CPU for preconditioned Gl-GMRES, Gl-GMRES(10) and Gl-BiCGSTAB methods for Example 3.

Example	method	m × m	16 × 16	32 × 32	64×64	128×128	256×256
No.1	IGl-PMHSS	$IT_{int}(Gl\text{-}CG)$	10.5	17.0	26.7	39.6	59.9
		$IT_{int}(Gl\text{-}CG)$	7.3	11.8	16.7	21.3	32.2
	IGl-PMHSS	$IT_{int}(PGl\text{-}CG)$	3	3.3	2.6	3.2	4.3
		$IT_{int}(PGl\text{-}CG)$	2	2.1	2	2	3
No.2	IGl-PMHSS	$IT_{int}(Gl\text{-}CG)$	10.4	16.5	33.5	58.6	59.9
		$IT_{int}(Gl\text{-}CG)$	6.5	12.3	24.7	45.0	64.1
	IGl-PMHSS	$IT_{int}(PGl\text{-}CG)$	3.5	4.4	4.9	7.0	12.0
		$IT_{int}(PGl\text{-}CG)$	2	2.6	3.4	4.9	8.6
No.3	IGl-PMHSS	$IT_{int}(Gl\text{-}CG)$	17.2	33.4	57.8	90.3	120
		$IT_{int}(Gl\text{-}CG)$	15.2	28.0	48.5	90.6	119.2
	IGl-PMHSS	$IT_{int}(PGl\text{-}CG)$	4.5	5.4	7.6	16.1	35.3
		$IT_{int}(PGl\text{-}CG)$	4	5.3	6.2	10.1	14.3

Table 7: Its and CPU for preconditioned Gl-GMRES, Gl-GMRES(10) and Gl-BiCGSTAB methods for Example 3.

to [4] and the references therein. Tables 5 and 6 report the numerical results.

As observed from Table 5, we see that the Gl-PMHSS iteration method returns the best convergence results than the Gl-GMRES and Gl-GMRES(20) methods in terms of iteration steps and CPU times. From Table 6, we can get similar observations to the ones made for the other two examples. As before, the PMHSS-preconditioned Gl-BiCGSTAB shows the same convergence properties described above and performs better than the PMHSS-preconditioned Gl-GMRES and Gl-GMRES(10) methods in both iteration steps and CPU times.

Example 4. In this example, we consider the performances of the IGl-PMHSS iteration method on above three numerical examples, and the numerical results are given in Table 7. In Table 7, we provide the average number of inner Gl-CG or inner preconditioned Gl-CG (PGl-CG) for each of the two symmetric and positive systems of linear equations with $\alpha V + W$ and $\alpha V + T$. Let the stopping criterion for the inner iterations satisfy $\varepsilon_k = 0.1$ and $\eta_k = 0.1$.

Based on the datas in Table 7, we can conclude several observations. First, for the IGl-PMHSS employed inner Gl-CG iterations, the average number of inner iterations per outer iteration grows rapidly with problem size. Second, using inner PGl-CG iteration with incomplete Cholesky factorization with drop tolerance 0.001 (Cholinc(A, 1.e-3)) [17] preconditioner, the average number of inner iterations per outer iteration is small in most cases and the growth can be alleviated. However, for the case $m = 256$ in Example 3, the average number of inner iterations per outer iteration is still large. This problem can be overcome by using a suitable preconditioner or other preconditioned Krylov subspace methods.

Conclusion

We have established and analyzed the Gl-PMHSS iteration method and the corresponding inexact variants for solving a class of complex symmetric linear systems with multiple right-hand sides. Similar to convergence properties of the PMHSS iteration method and PMHSS preconditioner, numerical results have shown the feasibility and effectiveness of the Gl-PMHSS method, and taking the parameter α to be 1 can still yield nearly optimal numerical results. To reduce the computational cost, the IGl-PMHSS iteration method is also implemented and analyzed in detail. Choosing a tighter tolerance in the inner stopping criterion and some suitable flexible Krylov subspace methods to deal with practical problems are under investigation and will be given in the future.

Competing Interests

The author declares that he has no competing interests.

Funding

The work is supported by the University Natural Science Research key Project of Anhui Province under grant No. KJ2015A242.

Funding

This research was partially supported by the Faculty of Science, King Mongkut's University of Technology Thonburi.

References

1. Agullo E, Giraud L, Jing YF (2014) Block GMRES method with inexact breakdowns and deflated restarting. SIAM J Matrix Anal Appl 35: 1625-1651.
2. BaiZZ, Golub, GH, Ng MK (2003) Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems. SIAM J Matrix Anal Appl 24: 603-626.
3. BaiZZ,GolubGH, NgMK (2008) On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems. Linear Algebra Appl 428: 413-440.
4. Bai ZZ, Benzi M,Chen F (2010) Modified HSS iteration methods for a class of complex symmetric linear systems. Computing 87: 93-111.
5. BaiZZ, BenziM,Chen F (2011) On preconditioned MHSS iteration methods for complex symmetric linear systems. Numer Algor 56: 297-317.
6. Bai ZZ (2011) On Hermitian and Skew-Hermitian Splitting Iteration Methods for Continuous Sylvester equations. J Comput Math 29: 185-198.
7. BoyseWE, SeidlAA (1996) A block QMR method for computing multiple simultaneous solutions to complex symmetric systems. SIAM J Sci Comput 17: 263-274.
8. Calandra H, Gratton S, Langou J, Xavier P, Xavier V (2012) Flexible variants of block restarted GMRES methods with application to geophysics. SIAM J Sci Comput 34: A714-A736.
9. Chan T, Wang W (1997) Analysis of projection methods for solving linear systems with multiple right-hand sides. SIAM J Sci Comput 18: 1698-1721.
10. Darnell D, Morgan RB, Wilcox W (2008) Deflated GMRES for systems with multiple shifts and multiple right-hand sides. Linear Algebra Appl 429: 2415-2434.
11. Jbilou K, Messaoudi A, Sadok H (1999) Global FOM and GMRES algorithms for matrix equation. Appl Numer Math 31: 49-63.
12. Jbilou K, Sadok H, Tinzefte A (2005) Oblique projection methods for linear systems with multiple right-hand sides. Elect Trans Numer Anal 20:119-138.
13. Malhotra M, Freund RW, Pinsky PM (1997) Iterative solution of multiple radiation and scattering problems in structural acoustics using a block quasi-minimal residual algorithm. Comput Methods Appl Mech Engrg 146:173-196.
14. Morgan RB (2005) Restarted block-GMRES with deflation of eignvalues. Appl Numer Math 54: 222-236.
15. O'Leary D (1980) The block conjugate gradient algorithm and related methods. Linear Algebra Appl 29: 293-332.
16. Saad Y (1987) On the Lanczos method for solving symmetric linear systems with several right-hand sides. Math Comp 48: 651-662.
17. Saad Y (2003) Iterative Methods for Sparse Linear Systems. second ed., SIAM, Philadelphia, PA.
18. Salkuyeh DK (2006) CG-type algorithms to solve symmetric matrix equations. Appl Math Comput 172: 985-999.
19. Smith CF, Peterson AF, Mittra R (1989) A conjugate gradient algorithm for the treatment of multiple incident electromagnetic fields. IEEE Trans Antennas Propagation 37: 1490-1493.
20. Simoncini V, Gallopoulos E (1993) Iterative methods for complex symmetric systems with multiple right-hand sides. Technical Report 1322, Center for Supercomputing Research and Development.
21. Vital B (1990) Etude de quelques m'ethodes de r'esolution de probl'emes lin'eaires de grade taille sur multiprocesseur. Ph. D. Thesis, Universit'e de Rennes, France.
22. Zhang J, Dai H, Zhao J (2010) Generalized global conjugate gradient squared algorithm. Appl Math Comput 216: 3694-3706.
23. Zhang J, Dai H, ZhaoJ (2011) A new family of global methods for linear systems with multiple right-hand sides. J Comp Appl Math 236: 1562-1575.