Journal of Computational and Applied Mathematics 313 (2017) 343–354

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A splitting method for complex symmetric indefinite linear system[☆]

Shi-Liang Wu[*], Cui-Xia Li

School of Mathematics and Statistics, Anyang Normal University, Anyang, 455000, PR China

ARTICLE INFO

Article history:
Received 6 July 2015
Received in revised form 10 June 2016

MSC:
65F10
65F50

Keywords:
Complex symmetric indefinite linear system
Complex symmetric matrix
Positive-stable splitting
Negative-stable splitting
Matrix splitting

ABSTRACT

In this paper, not requiring that the Hermitian part of the complex symmetric linear system must be Hermitian positive definite, a class of splitting methods is established by the modified positive/negative-stable splitting (PNS) of the coefficient matrix and is called the MPNS method. Theoretical analysis shows that the MPNS method is absolutely convergent under proper conditions. Some useful properties of the corresponding MPNS-preconditioned matrix are obtained. Numerical experiments are reported to illustrate the efficiency of both the MPNS method and the MPNS preconditioner.

1. Introduction

Consider the iterative solution of the linear system of the form

$$[-M + K + iC]x = f, \tag{1.1}$$

where $M, K, C \in \mathbb{R}^{n\times n}$ are symmetric and positive definite matrices, and $i = \sqrt{-1}$ denotes the imaginary unit. At present, this class of complex symmetric linear systems attracts considerable attention because it comes from many actual problems in scientific computing and engineering applications, such as structural dynamics [1–3] and Helmholtz equations [4–7]. For more details about the practical backgrounds of this class of problems, one can refer to [8–10] and the references therein.

In recent years, some efficient iteration methods have been proposed to solve complex symmetric linear system, such as Conjugate Orthogonal Conjugate Gradient (COCG) [11], Complex Symmetric (CSYM) [12], Quasi-Minimal Residual (QMR) [13] and so on. Whereas, practices show that directly using complex arithmetics throughout the code may be wasteful [14]. Given this, people turn to use the preconditioned Krylov subspace methods for solving the complex linear system. It is noted that when the preconditioned Krylov subspace methods are adopted to solve the complex linear system, preconditioning techniques for Krylov subspace methods are almost always mandatory, and many standard preconditioners will have the undesirable effect of 'spreading' nonreal entries to most positions in the preconditioning matrix, such as standard incomplete factorization and sparse approximate inverse preconditioners [9,14].

☆ This research was supported by NSFC (No. 11301009) 17HASTIT012, Natural Science Foundations of Henan Province (No. 15A110007), Project of Young Core Instructor of Universities in Henan Province (No. 2015GGJS-003).

* Corresponding author.
E-mail addresses: wushiliang1999@126.com, slwu@aynu.edu.cn (S.-L. Wu).

http://dx.doi.org/10.1016/j.cam.2016.09.028
0377-0427/

When matrix $-M+K$ is symmetric positive definite, one can deal with one of the several $2n \times 2n$ equivalent real formulations to avoid solving the complex linear system. For example, the complex symmetric linear system (1.1) may be written as

$$\begin{bmatrix} -M+K & C \\ -C & -M+K \end{bmatrix} \begin{bmatrix} x_1 \\ -x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ -b_2 \end{bmatrix}, \quad x = x_1 + ix_2,\ b = b_1 + ib_2. \tag{1.2}$$

For other real equivalent formulations of the complex symmetric linear system (1.1), one can see [15,9,8,16] for more details. For developing the efficient and fast convergent iteration methods to solve (1.2), Krylov subspace methods (such as GMRES) have been tested for (1.2) in [1]. Practices show that Krylov subspace methods without preconditioner for solving this formulation are not ideal [16]. Further, in [8], numerical experiments show that Krylov subspace methods with standard incomplete LU (ILU) preconditioner for this formulation are efficient and feasible. Other types of block preconditioners for Krylov subspace methods to solve the corresponding real equivalent formulations have been proposed in [9]. Based on Hermitian and skew-Hermitian splitting (HSS in [17]) of the coefficient matrix in (1.2), a class of preconditioned MHSS (PMHSS) iteration methods for (1.2) has been constructed in [18]. Obviously, the PMHSS method belongs to the category of stationary matrix splitting iteration methods. Based on the following parameter-dependent formulation

$$\begin{bmatrix} -M+K-\alpha C & \sqrt{1+\alpha^2}C \\ \sqrt{1+\alpha^2}C & M-K-\alpha C \end{bmatrix} \begin{bmatrix} x_1 \\ z \end{bmatrix} = \begin{bmatrix} b_1 \\ d \end{bmatrix} \tag{1.3}$$

with a real non-zero parameter α, a class of complex-valued to real-valued (C-to-R) methods has been proposed in [15]. Essentially, the 'C-to-R' method is a preconditioned iteration method applied to a Schur-complement reduction of a bilaterally transformed variant of the block two-by-two linear system (1.3). In the implementations, the corresponding Schur complement linear system can be solved efficiently by using the preconditioner $-M+K+\alpha C$ when matrix C is symmetric positive semi-definite [10,19].

When matrix $-M+K$ is symmetric indefinite, the above-mentioned iteration methods (such as HSS [17], modified HSS (MHSS) [2], PMHSS [3,18] and 'C-to-R' [15]) may seriously challenge because matrix $\alpha I - M + K$ in [2,3,18,17,20,21] and $-M+K+\alpha C$ [15] may be indefinite or singular. For efficiently solving the complex symmetric linear system (1.1) with the symmetric indefinite matrix $-M+K$, a class of the CSS methods [14] has been established by the complex-symmetric and skew-Hermitian splitting (CSS) of the coefficient matrix from the classical state–space formulation of frequency analysis of the discrete dynamical system. Whereas, in theory, it only shows in [14] that the spectral radius of the iteration matrix of the CSS method is less than or equal to one. Recently, to overcome the symmetric indefinite matrix $-M+K$, by making use of the symmetric positive definite matrix $(-M+K)^2$, the Hermitian normal splitting (HNS) and simplified Hermitian normal splitting (SHNS) methods for solving the complex symmetric linear system (1.1) have been designed in [22]. Although the HNS and SHNS methods can be applied to solve the complex symmetric linear system (1.1), matrix $(-M+K)^2$ has serious effect on the convergence rate of the HNS and SHNS methods because the cost of the computation of matrix $(-M+K)^2$ may be high in the implementations.

In this paper, we will develop a class of the efficient iteration methods for solving the complex symmetric linear system (1.1). Not requiring that matrix $-M+K$ in (1.1) must be symmetric positive definite, the MPNS method can be established by the modified positive/negative-stable splitting (PNS) of the coefficient matrix of the complex symmetric linear system (1.1). The convergence properties of the MPNS method are discussed.

The remainder of the paper is organized as follows. In Section 2, we establish the MPNS method. In Section 3, the convergence properties of the MPNS method are discussed and the corresponding MPNS preconditioner is presented. In Section 4, the results of numerical experiments from the n-degree-of-freedom (n-DOF) linear system and the parabolic equations are reported. Finally, in Section 5 we give some conclusions to end the paper.

2. The MPNS method

To establish the MPNS method, we assume that

$$A = -M + K + iC.$$

Then the complex symmetric linear system (1.1) can be written as

$$Ax \equiv [-M+K+iC]x = f. \tag{2.1}$$

If the coefficient matrix A in (2.1) is expressed as

$$A = P + N \tag{2.2}$$

with

$$P = K \quad \text{and} \quad N = -M + iC,$$

then the matrix splitting (2.2) of the coefficient matrix A is a positive/negative-stable splitting (PNS) because matrices P and N, respectively, are positive-stable and negative-stable.

Based on the matrix splitting (2.2) of the coefficient matrix A, the complex symmetric linear system (1.1) can be rewritten into the following system of fixed-point equations

$$(\alpha I + P)x = (\alpha I - N)x + f \quad \text{or} \quad (\alpha I + K)x = (\alpha I + M - iC)x + f, \tag{2.3}$$

where α is a given positive parameter. Note that the complex symmetric linear system (1.1) is also equivalent to $-iAx = -if$, i.e.,

$$(iM + C - iK)x = -if,$$

which can be also rewritten into the following system of fixed-point equations

$$(\alpha I - iN)x = (\alpha I + iP)x - if \quad \text{or} \quad (\alpha I + C + iM)x = (\alpha I + iK)x - if. \tag{2.4}$$

Now, we can establish the following modified positive/negative-stable splitting (MPNS) iteration method for solving the complex symmetric linear system (1.1) by alternately iterating between two systems of fixed-point equations (2.3) and (2.4).

The MPNS method. Let $x^{(0)} \in \mathbb{C}^n$ be an arbitrary initial guess. For $k = 0, 1, 2, \ldots$ until the sequence of iterates $\{x^{(k)}\}_{k=0}^{\infty}$ converges, compute the next iterate $x^{(k+1)}$ according to the following procedure:

$$\begin{cases} (\alpha I + K)x^{\left(k+\frac{1}{2}\right)} = (\alpha I + M - iC)x^{(k)} + f, \\ (\alpha I + C + iM)x^{(k+1)} = (\alpha I + iK)x^{\left(k+\frac{1}{2}\right)} - if, \end{cases} \tag{2.5}$$

where α is a given positive constant and I is the identity matrix.

Evidently, each step of the MPNS iteration needs to alternate between the symmetric positive definite matrix $\alpha I + K$ and the positive stable matrix $\alpha I + C + iM$, that is to say, the MPNS method is needs to solve two linear sub-systems with $\alpha I + K$ and $\alpha I + C + iM$. Based on this fact, it is not difficult to find that the MPNS method no longer needs to keep matrix $-M + K$ symmetric positive definite. In other words, the MPNS method can avoid the symmetric indefinite matrix $-M + K$. In the actual computations, since the coefficient matrix, $\alpha I + K$, of linear systems is symmetric positive definite, there can be employed mostly real arithmetic (such as the Cholesky factorization, the conjugate gradient (CG) method) to gain its solution easily. Since the coefficient matrix, $\alpha I + C + iM$, of linear systems is positive stable, it can make use of Krylov subspace methods (such as GMRES [23]) to obtain its solution.

By eliminating the intermediate vector $x^{(k+\frac{1}{2})}$ in (2.5), one can obtain the following iteration in fixed-point form as

$$x^{(k+1)} = M_\alpha x^{(k)} + N_\alpha f, \quad k = 0, 1, 2, \ldots \tag{2.6}$$

where

$$M_\alpha = (\alpha I + C + iM)^{-1}(\alpha I + iK)(\alpha I + K)^{-1}(\alpha I + M - iC)$$

and

$$N_\alpha = (1 - i)\alpha(\alpha I + C + iM)^{-1}(\alpha I + K)^{-1}.$$

Note that M_α is the iteration matrix of the MPNS method.

In addition, if we introduce matrices

$$B_\alpha = \frac{1+i}{2\alpha}(\alpha I + K)(\alpha I + C + iM) \quad \text{and} \quad C_\alpha = \frac{1+i}{2\alpha}(\alpha I + iK)(\alpha I + M - iC),$$

then

$$A = B_\alpha - C_\alpha \quad \text{and} \quad M_\alpha = B_\alpha^{-1}C_\alpha. \tag{2.7}$$

Therefore, the MPNS method can be also induced by the matrix splitting $A = B_\alpha - C_\alpha$. It follows that the splitting matrix B_α can be viewed as a preconditioner used to solve the complex symmetric linear system (1.1). When the preconditioner B_α is applied to solve the complex symmetric linear system (1.1), it is not difficult to find that the multiplicative factor $\frac{1+i}{2\alpha}$ has no effect on the preconditioned system and thus it can be dropped. In practice, normally, matrix $B_\alpha = (\alpha I + K)(\alpha I + C + iM)$ can be referred to as the MPNS preconditioner. Clearly, matrix $B_\alpha = (\alpha I + K)(\alpha I + C + iM)$ is positive-stable.

3. Convergence analysis

As previously mentioned, the MPNS method belongs to the two-step splitting iteration framework. Thus, to study the convergence property of the MPNS method, the following lemmas are required.

Lemma 3.1 *([17]). Let $A \in \mathbb{C}^{n\times n}$, $A = M_i - N_i$ $(i = 1, 2)$ be two splittings of A, and $x^{(0)} \in \mathbb{C}^n$ be a given initial vector. If $\{x^{(k)}\}$ is a two-step iteration sequence defined by*

$$\begin{cases} M_1 x^{\left(k+\frac{1}{2}\right)} = N_1 x^{(k)} + f, \\ M_2 x^{(k+1)} = N_2 x^{\left(k+\frac{1}{2}\right)} + f, \end{cases} \quad k = 0, 1, \ldots,$$

then

$$x^{(k+1)} = M_2^{-1} N_2 M_1^{-1} N_1 x^{(k)} + M_2^{-1}(I + N_2 M_1^{-1}) f, \quad k = 0, 1, \ldots.$$

Moreover, if the spectral radius $\rho(M_2^{-1}N_2M_1^{-1}N_1) < 1$, then the iterative sequence $\{x^{(k)}\}$ converges to the unique solution $x^* \in \mathbb{C}^n$ of the system (1.1) for all initial vectors $x^{(0)} \in \mathbb{C}^n$.

Lemma 3.2. *Let $A = S + iT$, where $S = \frac{1}{2}(A + A^*)$ and $T = \frac{-i}{2}(A - A^*)$. If S is positive (semi-) definite and $S \pm T$ are positive semi-definite, then*

$$\|(\alpha I - iA)(\alpha I + A)^{-1}\|_2 \le 1 \quad \text{for } \forall\, \alpha > 0.$$

If, in addition, S is positive definite, and $S \pm T$ are positive definite, then

$$\|(\alpha I - iA)(\alpha I + A)^{-1}\|_2 < 1 \quad \text{for } \forall\, \alpha > 0.$$

Proof. Let

$$W = (\alpha I - iA)(\alpha I + A)^{-1}.$$

Then

$$\begin{aligned} \|W\|_2^2 &= \max_{x\neq 0} \frac{((\alpha I - iA)(\alpha I + A)^{-1}x, (\alpha I - iA)(\alpha I + A)^{-1}x)}{(x, x)} \\ &= \max_{y\neq 0} \frac{((\alpha I - iA)y, (\alpha I - iA)y)}{((\alpha I + A)y, (\alpha I + A)y)} \\ &= \max_{y\neq 0} \frac{(\alpha y - iAy)^*(\alpha y - iAy)}{(\alpha y + Ay)^*(\alpha y + Ay)} \\ &= \max_{y\neq 0} \frac{\alpha^2 y^*y - i\alpha y^*Ay + i\alpha y^*A^*y + y^*A^*Ay}{\alpha^2 y^*y + \alpha y^*Ay + \alpha y^*A^*y + y^*A^*Ay} \\ &= \max_{y\neq 0} \frac{\alpha^2 y^*y - i\alpha y^*(A - A^*)y + y^*A^*Ay}{\alpha^2 y^*y + \alpha y^*(A + A^*)y + y^*A^*Ay} \\ &= \max_{y\neq 0} \frac{\alpha^2 y^*y + 2\alpha y^*Ty + y^*A^*Ay}{\alpha^2 y^*y + 2\alpha y^*Sy + y^*A^*Ay}. \end{aligned}$$

When S is positive (semi-) definite and $S \pm T$ are positive semi-definite, $\|W\|_2^2 \le 1$ for $\alpha > 0$; when S is positive definite and $S \pm T$ are positive definite, $\|W\|_2^2 < 1$ for $\alpha > 0$. The proof of Lemma 3.2 is completed. □

Based on the results of Lemma 3.2, naturally, the following corollary is obtained.

Corollary 3.1. *Let $A = S + iT$, where S and T are real symmetric matrices. If S and T are positive (semi-) definite, and $S - T$ is positive semi-definite, then*

$$\|(\alpha I - iA)(\alpha I + A)^{-1}\|_2 \le 1 \quad \text{for } \forall\, \alpha > 0.$$

If, in addition, S and T are positive definite, and $S - T$ is positive definite, then

$$\|(\alpha I - iA)(\alpha I + A)^{-1}\|_2 < 1 \quad \text{for } \forall\, \alpha > 0.$$

Remark 3.1. The structure of Lemma 3.2 or Corollary 3.1 is similar to that of Kellogg's lemma [24,25], whereas, the results in Lemma 3.2 or Corollary 3.1 are different from that of Kellogg's lemma.

Concerning the convergence property of the MPNS method, based on Lemma 3.1 and Corollary 3.1, we have the following theorem.

Theorem 3.1. *Let $A = -M + K + iC$, where M, K and C are real symmetric and positive definite matrices, and let α be a positive constant. Then the iteration matrix M_α of the MPNS method is*

$$M_\alpha = (\alpha I + C + iM)^{-1}(\alpha I + iK)(\alpha I + K)^{-1}(\alpha I + M - iC). \tag{3.1}$$

If matrix $C - M$ is positive semi-definite, then the spectral radius $\rho(M_\alpha)$ of the MPNS iteration matrix is bounded by

$$\sigma(\alpha) \equiv \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i},$$

where $\lambda(K)$ denotes the spectrum of the matrix K. Therefore, it holds that

$$\rho(M_\alpha) \le \sigma(\alpha) < 1, \quad \forall\, \alpha > 0,$$

i.e., the MPNS iteration converges to the unique solution of the system of linear equations (1.1) for any initial guess.

Proof. Setting in Lemma 3.1

$$M_1 = \alpha I + K, \qquad N_1 = \alpha I + M - iC, \qquad M_2 = \alpha I + C + iM \quad \text{and} \quad N_2 = \alpha I + iK.$$

Since matrices $\alpha I + K$ and $\alpha I + C + iM$ are nonsingular for $\alpha > 0$, Eq. (3.1) holds.

By the similarity invariance of the matrix spectrum, the iteration matrix M_α is similar to

$$\bar{M}_\alpha = (\alpha I + iK)(\alpha I + K)^{-1}(\alpha I + M - iC)(\alpha I + C + iM)^{-1}.$$

Then

$$\begin{aligned}
\rho(M_\alpha) &= \rho((\alpha I + iK)(\alpha I + K)^{-1}(\alpha I + M - iC)(\alpha I + C + iM)^{-1}) \\
&\le \|(\alpha I + iK)(\alpha I + K)^{-1}(\alpha I + M - iC)(\alpha I + C + iM)^{-1}\|_2 \\
&\le \|(\alpha I + iK)(\alpha I + K)^{-1}\|_2 \|(\alpha I + M - iC)(\alpha I + C + iM)^{-1}\|_2 \\
&\le \|(\alpha I + iK)(\alpha I + K)^{-1}\|_2 \|(\alpha I - i(C + iM))(\alpha I + C + iM)^{-1}\|_2.
\end{aligned}$$

Let $U_\alpha = (\alpha I - i(C + iM))(\alpha I + C + iM)^{-1}$. Note that matrices C and M are symmetric positive definite and matrix $C - M$ is positive semi-definite, from Corollary 3.1, it is easy to see that $\|U_\alpha\|_2 \le 1$. Further,

$$\rho(M_\alpha) \le \|(\alpha I + iK)(\alpha I + K)^{-1}\|_2 = \max_{\lambda_i \in \lambda(K)} \left| \frac{\alpha + i\lambda_i}{\alpha + \lambda_i} \right| = \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i}. \tag{3.2}$$

Since $\lambda_i > 0$ $(i = 1, 2, \ldots, n)$ and $\alpha > 0$, it is easy to see that

$$\rho(M_\alpha) \le \sigma(\alpha) < 1,$$

i.e., the MPNS method can converge to the unique solution of the complex symmetric linear system (1.1) under proper conditions. □

Remark 3.2. In practice, the positive (semi-) definite matrix $C - M$ in Theorem 3.1 is often encountered, one can see [2,3,26,14,22] for more details. In other words, the assumed condition, the positive (semi-) definite matrix $C - M$, of Theorem 3.1 is reasonable and can be accepted.

Remark 3.3. Theorem 3.1 shows that the convergence speed of the MPNS iteration is indeed governed by $\sigma(\alpha)$ under certain conditions, which only depends on the spectrum of the symmetric positive definite matrix K, does not depend on the spectrum of the matrix $C + iM$, on the spectrum of the matrix $-M + K$, on the spectrum of the coefficient matrix A, or on the eigenvectors of the matrices K, $C + iM$ and A.

If matrices C and M satisfy $C = \beta M$ $(\beta \ge 1)$, then we have the following result.

Theorem 3.2. *Let $A = -M + K + iC$, where M, K and C are real symmetric and positive definite matrices, and let α be a positive constant. If $C = \beta M$ $(\beta \ge 1)$, then the spectral radius $\rho(M_\alpha)$ of the MPNS iteration matrix M_α is bounded by*

$$\sigma(\alpha) \equiv \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i},$$

where $\lambda(K)$ denotes the spectrum of the matrix K. Therefore, it holds that

$$\rho(M_\alpha) \le \sigma(\alpha) < 1, \quad \forall\, \alpha > 0,$$

i.e., the MPNS iteration converges to the unique solution of the system of linear equations (1.1) for any initial guess.

Proof. Based on the proof of Theorem 3.1, note that $C = \beta M$ $(\beta \geq 1)$, we have

$$\begin{aligned}\rho(M_\alpha) &\leq \|(\alpha I + iK)(\alpha I + K)^{-1}\|_2 \|(\alpha I - i(C + iM))(\alpha I + C + iM)^{-1}\|_2 \\ &\leq \|(\alpha I + iK)(\alpha I + K)^{-1}\|_2 \|(\alpha I - i(\beta + i)M)(\alpha I + (\beta + i)M)^{-1}\|_2.\end{aligned}$$

Since K and M are symmetric positive definite, there exist orthogonal matrices $S_1, S_2 \in \mathbb{R}^{n\times n}$ such that

$$S_1^T K S_1 = \Lambda_K, S_2^T M S_2 = \Lambda_M,$$

where $\Lambda_K = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_n)$ and $\Lambda_M = \operatorname{diag}(\mu_1, \mu_2, \ldots, \mu_n)$ with $\lambda_i > 0$ and $\mu_i > 0$ $(1 \leq i \leq n)$ being the eigenvalues of the matrices K and M, respectively.

By the simple calculations, for $\beta \geq 1$, we can get that

$$\begin{aligned}\rho(M_\alpha) &\leq \max_{\lambda_i \in \lambda(K)} \left|\frac{\alpha + i\lambda_i}{\alpha + \lambda_i}\right| \max_{\mu_i \in \mu(M)} \left|\frac{\alpha - i(\beta + i)\mu_i}{\alpha + (\beta + i)\mu_i}\right| \\ &= \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i} \max_{\mu_i \in \mu(M)} \sqrt{\frac{(\alpha + \mu_i)^2 + \beta^2\mu_i^2}{(\alpha + \beta\mu_i)^2 + \mu_i^2}}, \\ &= \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i} \max_{\mu_i \in \mu(M)} \sqrt{\frac{\alpha^2 + 2\alpha\mu_i + \mu_i^2 + \beta^2\mu_i^2}{\alpha^2 + 2\alpha\beta\mu_i + \beta^2\mu_i + \mu_i^2}} \\ &\leq \max_{\lambda_i \in \lambda(K)} \frac{\sqrt{\alpha^2 + \lambda_i^2}}{\alpha + \lambda_i}.\end{aligned}$$

The rest of proof is similar to the proof of Theorem 3.1, which is omitted. □

From the upper bound $\sigma(\alpha)$ of the spectral radius $\rho(M_\alpha)$ of the iteration matrix M_α, Theorems 3.1 and 3.2 provide some convergence conditions of the MPNS method for solving the system of linear equations (1.1). From Theorems 3.1 and 3.2, the convergent speed of the MPNS method may depend on two factors: (1) the spectrum of the symmetric positive definite matrix K; (2) the choice of the iteration parameter α. As the former is determined, the latter needs to be estimated. In practice, to improve the efficiency of the MPNS method, it is desirable to determine or find a good estimate of the optimal parameter α to minimize the convergence factor. Since the optimal parameter α minimizing the spectral radius $\rho(M_\alpha)$ is hardly obtained in general, we instead give the parameter α in the following corollary to minimize the upper bound $\sigma(\alpha)$ of the spectral radius $\rho(M_\alpha)$. More specifically, when the smallest and largest eigenvalues of the matrix K are known, the value of α can be obtained to minimize the upper bound $\sigma(\alpha)$. This fact is precisely stated as the following corollary, one can see [2] for more details.

Corollary 3.2. *Let the conditions of Theorem 3.1 or 3.2 be satisfied. Let $\lambda_{\min}$ and $\lambda_{\max}$ be the smallest and largest eigenvalues of the symmetric positive definite matrix K, respectively. Then*

$$\alpha^* = \arg\min_{\alpha}\left\{\max_{\lambda_{\min}\leq\lambda\leq\lambda_{\max}} \frac{\sqrt{\alpha^2 + \lambda^2}}{\alpha + \lambda}\right\} = \sqrt{\lambda_{\min}\lambda_{\max}}$$

and

$$\sigma(\alpha^*) = \frac{\sqrt{\lambda_{\max} + \lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}} = \frac{\sqrt{\kappa(K) + 1}}{\sqrt{\kappa(K)} + 1},$$

where $\kappa(K) = \frac{\lambda_{\max}}{\lambda_{\min}}$ is the spectral condition number of the matrix K.

Some remarks on Theorems 3.1–3.2 and Corollary 3.2 are given below.

- The optimal iteration parameter α^* given in Corollary 3.2 only minimizes the upper bound $\sigma(\alpha)$ of the spectral radius $\rho(M_\alpha)$ of the MPNS iteration matrix M_α, but not $\rho(M_\alpha)$ itself. The form of the optimal iteration parameter α^* is the same as that of the HSS method and its variants, one can also see [2,3,18,17,20] for details.
- Based on the inexact HSS (IHSS) method [17,27], we can establish the inexact MPNS (IMPNS) method, whose convergence conditions can be obtained as well. One can see [17,27,28] for more details.
- In practice, in order to improve the convergence rate of the MPNS method, a feasible strategy is to replace the identity matrix I in (2.5) with the symmetric positive definite matrix P. This leads to the preconditioned MPNS (PMPNS) method. It is easy to see that the convergence properties of the PMPNS method are similar to the MPNS method.
- Using another parameter $\bar{\alpha}$ instead of α of the first equation in (2.5), a class of the biparametric MPNS (BMPNS) methods can be established. Using this strategy to adjust and improve the convergence rate of the iteration methods, one can see [21,29] for more details.

Table 1
The optimal values α^* for MPNS.

$m \times m$	8×8	16×16	32×32
α^*	1.3681	0.735	0.3802

It is not difficult to find that all the eigenvalues of the MPNS-preconditioned matrix lie in the interior of the disk of radius 1 centered at the point $(1, 0)$ on the basis of the results in Theorem 3.1 or 3.2. Therefore, the MPNS-preconditioned matrix is positive definite. Further, when the spectral radius is reasonably small, all the eigenvalues of the MPNS-preconditioned matrix will gather into $(1, 0)$. Obviously, these are desirable properties when the preconditioner B_α is used to accelerate the convergence rate of Krylov subspace methods (such as GMRES). Further discussing about this point, one can see [21] for more details.

The following theorem describes the spectral distribution of the MPNS-preconditioned matrix $B_\alpha^{-1}A$, whose proof is omitted. One can see [14] for more details.

Theorem 3.3. *Let the conditions of Theorem 3.1 or 3.2 be satisfied. Then all the eigenvalues of the MPNS-preconditioned matrix $B_\alpha^{-1}A$ satisfy $|1 - |\lambda|| \le 1$.*

4. Numerical experiments

In this section, some numerical experiments are reported to illustrate the performance of the MPNS method and the MPNS preconditioner for the complex symmetric linear system (1.1). In addition, the efficiency of the MPNS and SHNS [22] methods for the complex symmetric linear system (1.1) has been investigated. All tests are started from the zero vector, performed in MATLAB 7.0 with machine precision 10^{-16}.

Example 1. Consider the equations of motion of an n-DOF linear system in matrix form as

$$M\ddot{q} + C\dot{q} + Kq = p, \tag{4.1}$$

where q is the configuration vector and p is the vector of generalized components of dynamic forces, matrices M, K and C are the inertia, stiffness and viscous-damping matrices, respectively. Complex harmonic excitation at the driving circular frequency ω, i.e., of the type $p(t) = fe^{i\omega t}$, admits the steady-state solution $q(t) = \tilde{q}(\omega)e^{i\omega t}$, where $\tilde{q}$ solves the linear system $E(\omega)\tilde{q}(\omega) = f$. Substituting $q(t) = \tilde{q}(\omega)e^{i\omega t}$ into (4.1) leads to the following complex-valued linear system

$$[-\omega^2 M + K + i\omega C]x = f, \tag{4.2}$$

where $M, K, C \in \mathbb{R}^{n\times n}$ are symmetric and positive definite matrices, one can see [3,1,2,14] for more details.

In the numerical experiments, the matrix K in (4.2) is of the tensor-product form

$$K = I \otimes V_m + V_m \otimes I \quad \text{with } V_m = h^{-2}\text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m\times m},$$

which is the five-point centered difference matrix approximating the negative Laplacian operator with homogeneous Dirichlet boundary conditions, on a uniform mesh in the unit square $[0, 1] \times [0, 1]$ with the mesh-size $h = \frac{1}{m+1}$. We take $C = \omega C_V + C_H$, where C_V and C_H, respectively, are the viscous and hysteretic damping matrices. The total number of variables is $n = m^2$.

In the implementations, to obtain the faster convergence speed for the MPNS method, based on Corollary 3.2, the optimal iteration parameters $\alpha^* = \sqrt{\lambda_{\min}\lambda_{\max}}$ are adopted. Specifically, see Table 1.

From Table 1, with the mesh-size increasing, the optimal iteration parameters α^* are decreasing for the MPNS method.

4.1. Results for MPNS iteration

In this subsection, the results of the numerical test for the MPNS method are listed. In [22], the following SHNS method has been established to solve the complex symmetric linear system (4.2) with the symmetric indefinite matrix $-\omega^2 M + K$, that is to say,

$$\begin{cases} (\alpha I + iW)x^{\left(k+\frac{1}{2}\right)} = (\alpha T - W^2)x^{(k)} + i\alpha b, \\ (\alpha T + W^2)x^{(k+1)} = (\alpha I - iW)x^{\left(k+\frac{1}{2}\right)} - i\alpha b, \end{cases} \qquad (\alpha > 0),$$

where $W = -\omega^2 M + K$ and $T = \omega C$. To illustrate the efficiency of the MPNS method, we compare the MPNS method with the HNS method. To this end, here, two aspects are considered to illustrate the performance of the MPNS and SHNS methods: one is $C = \omega C_V = \omega^2 \beta M$ $(\beta \ge 1)$, the other is $C = \omega C_V + C_H$ to keep matrix $C - \omega M$ symmetric positive semi-definite

Table 2
IT and CPU(s) for $C = C_V = \beta M$.

		β	1	2	4	6	8
8×8	MPNS	IT	82	82	82	83	88
		CPU(s)	0.016	0.016	0.016	0.018	0.021
	SHNS	IT	212	212	212	213	216
		CPU(s)	0.047	0.062	0.047	0.063	0.047
		$\bar{\alpha}$	143.6155	71.8078	35.9039	17.9519	8.976
16×16	MPNS	IT	138	138	139	140	145
		CPU(s)	0.038	0.0375	0.39	0.406	0.407
	SHNS	IT	–	–	–	–	–
		CPU(s)	–	–	–	–	–
		$\bar{\alpha}$	148.1277	74.0638	37.0319	18.516	9.258
32×32	MPNS	IT	240	240	241	243	250
		CPU(s)	15.765	15.672	15.704	15.922	16.328
	SHNS	IT	–	–	–	–	–
		CPU(s)	–	–	–	–	–
		$\bar{\alpha}$	149.4381	74.7191	37.3595	18.6798	9.3399

Table 3
IT and CPU(s) for $C = \omega C_V + C_H$ and ω.

		ω	3	2	1	0.5
8×8	MPNS	IT	163	102	82	79
		CPU(s)	0.031	0.016	0.015	0.015
	SHNS	IT	275	171	118	90
		CPU(s)	0.078	0.046	0.015	0.032
		$\bar{\alpha}$	3.913	7.8231	14.757	22.6668
16×16	MPNS	IT	258	168	139	133
		CPU(s)	0.75	0.485	0.407	0.375
	SHNS	IT	623	365	232	168
		CPU(s)	1.235	0.703	0.453	0.344
		$\bar{\alpha}$	2.7579	5.1543	8.9447	13.0306
32×32	MPNS	IT	434	289	241	232
		CPU(s)	28.516	19.128	15.781	15.188
	SHNS	IT	–	–	429	305
		CPU(s)	–	–	24.063	19
		$\bar{\alpha}$	1.5859	2.8698	4.8111	6.88

at least. The goal of the former is to confirm the results in Theorem 3.2 and the goal of the latter is to confirm the results in Theorem 3.1.

The MPNS and SHNS methods terminate if the relative residual error satisfies $\frac{\|r^{(k)}\|_2}{\|r^{(0)}\|_2} < 10^{-6}$. 'IT' denotes the number of iteration. 'CPU(s)' denotes the time (in seconds) required to solve a problem.

For $C = \omega C_V = \omega^2 \beta M$ $(\beta \geq 1)$, we take $M = I$, $\omega = 1$, $C_H = \mu K$ with a damping coefficient $\mu = 0.02$. The right-hand side vector b is to be adjusted such that $b = (1 + i)Ae$ $(e = (1, 1, \dots, 1)^T)$. In our numerical computations, we normalize the system by multiplying both sides through by h^2.

In Table 2, we present some iteration results to illustrate the convergence behavior of the MPNS and SHNS methods with $C = \omega C_V = \omega^2 \beta M$ $(\beta \geq 1)$. To compare the MPNS method with the SHNS method, $\bar{\alpha}$ in Tables 2–3 denotes the optimal iteration parameter for the SHNS method, the value of $\bar{\alpha}$ is $\frac{1}{\sqrt{\mu_{\min}\mu_{\max}}}$, where $\mu_{\min}$ and $\mu_{\max}$ are the extreme eigenvalues of the symmetric positive definite matrix $W^{-1}TW^{-1}$. In Tables 2–3, CPU times larger than 500 s or the iteration numbers larger than 500 steps are simply listed by the symbol '—'.

In Table 2, we find that under the same grid, with the value of parameter β increasing, the iteration numbers of the MPNS and SHNS methods are increasing; under the same parameter β, with the mesh-size increasing, the iteration numbers of the MPNS and SHNS methods are also increasing. From Table 2, the iteration numbers and CPU times of the MPNS method are less than that of the SHNS method under certain conditions. Based on the results in Table 2, we have also noticed that the optimal choice of the parameter β for MPNS may be one.

For $C = \omega C_V + C_H$, we take $M = I$, $C_V = 5I$, $C_H = \mu K$ with $\mu = 0.02$. As mentioned, the right-hand side vector b is to be adjusted such that $b = (1 + i)Ae$ $(e = (1, 1, \dots, 1)^T)$ and the system is normalized by multiplying both sides through by h^2.

Table 4
IT and CPU(s) of GMRES(20) with $C = C_V = M$.

		α	0.01	0.05	0.1	0.5	1
8×8	P_{MPNS}	IT	4	4	5	6	7
		CPU(s)	0.0123	0.0126	0.0152	0.0155	0.0155
	P_{SHNS}	IT	12	12	12	12	11
		CPU(s)	0.015	0.015	0.016	0.015	0.016
16×16	P_{MPNS}	IT	5	6	7	10	11
		CPU(s)	0.079	0.094	0.109	0.141	0.156
	P_{SHNS}	IT	36	37	37	35	30
		CPU(s)	0.511	0.515	0.516	0.5	0.437
32×32	P_{MPNS}	IT	6	8	10	15	19
		CPU(s)	7.922	8.141	9.718	13.797	17.875
	P_{SHNS}	IT	296	287	278	145	93
		CPU(s)	268.172	253.359	246.906	127.875	82.016

In the numerical experiments, some values of the driving circular frequency ω need to be selected to keep $C - \omega M$ symmetric positive definite. Specifically, see Table 3. We present some iteration results to illustrate the convergence behavior of the MPNS method with the symmetric positive definite matrix $C - \omega M$ in Table 3.

In Table 3, we find that under the same parameter ω, with the mesh-size increasing, the iteration numbers of the MPNS and SHNS methods are increasing; under the same grid, with the driving circular frequency ω decreasing, the iteration numbers of the MPNS and SHNS methods are decreasing. In terms of the iteration numbers and CPU times, the MPNS method outperforms the SHNS method under certain conditions.

No matter $C = \omega C_V = \omega^2 \beta M$ ($\beta \geq 1$) or $C = \omega C_V + C_H$, in our numerical experiments, when the MPNS and SHNS methods are used to solve the complex symmetric linear system (1.1), the MPNS method outperforms the SHNS method under certain conditions. Compared with the SHNS method, the MPNS method applied to solve the complex symmetric linear system (1.1) may be the top priority under certain conditions.

4.2. The MNPS preconditioner

For efficiently solving the complex symmetric linear system (1.1), Krylov subspace methods (such as GMRES(ℓ)) can be adopted because the corresponding coefficient matrix is not necessary to be positive definite. In general, the choice of the restart parameter ℓ in GMRES(ℓ) (ℓ is less than the order of the coefficient matrix A in (2.2)) is no general rule and mainly depends on a matter of experience in practice. In our numerical computations, for the sake of simplicity, we take $\ell = 20$. To investigate the performance of the MNPS preconditioner, we compare the MPNS preconditioner with the SHNS preconditioner, which is defined by

$$P_{SHNS} = (\alpha I + iW)(\alpha T + W^2)$$

with $W = -\omega^2 M + K$ and $T = \omega C$ in [22]. The purpose of these experiments is just to investigate the influence of the MPNS and SHNS preconditioners on the convergence behaviors of GMRES(20).

To easily compare the MPNS preconditioner with the SHNS preconditioner, the efficiency of both the MPNS preconditioner and the SHNS preconditioner has been investigated under the same iteration parameter α. In Tables 4– 5, the value of the parameter α is selected by the statement on the choice of the iteration parameter [25], that is to say, experience suggests that in most applications and for an appropriate scaling of the problem, a 'small' value of α (usually between 0.01 and 0.5) may give good results. In this case, some iteration results are presented in Tables 4–5 to illustrate the convergence behaviors of two preconditioners. With respect to the choice of α, one can see [25,30] about the choice of the iteration parameter α.

The GMRES(20) method terminates if the relative residual error satisfies $\frac{\|r^{(k)}\|_2}{\|r^{(0)}\|_2} < 10^{-6}$. In Tables 4–5, '$P_{MPNS}$' denotes the MPNS-preconditioned GMRES(ℓ) method, 'P_{SHNS}' denotes the SHNS-preconditioned GMRES(ℓ) method.

In Tables 4–5, we report the numerical results when GMRES(20) preconditioned with MPNS and SHNS are used to solve the complex symmetric linear system (1.1). From Tables 4–5, we find that under the same parameter α, with the mesh-size increasing, the iteration numbers and CPU times of the MPNS- and SHNS-preconditioners are increasing. From computational efficiency, when used as a preconditioner, MPNS performs much better than SHNS from the iteration numbers and CPU times under certain conditions. Compared with the MPNS preconditioner and the SHNS preconditioner, the MPNS preconditioner is quite competitive in terms of convergence rate, robustness and efficiency when Krylov subspace methods preconditioned with MPNS and SHNS are applied to solve the linear system (1.1) under certain conditions.

In Tables 4–5, some iteration results are listed for $\alpha = 1$ when the MPNS- and SHNS-preconditioned GMRES(20) are applied to solve the complex symmetric linear system (1.1). Compared with iteration results of Tables 4–5, $\alpha = 1$ may not be a good choice when the MPNS-preconditioner is applied to solve the complex symmetric linear system (1.1).

Table 5
IT and CPU(s) of GMRES(20) with $C = \omega C_V + C_H$ and $\omega = \pi$.

		α	0.01	0.05	0.1	0.5	1
8×8	P_{MPNS}	IT	6	6	6	6	7
		CPU(s)	0.0124	0.0124	0.0130	0.0128	0.133
	P_{SHNS}	IT	12	12	12	11	11
		CPU(s)	0.016	0.0162	0.0161	0.0157	0.162
16×16	P_{MPNS}	IT	11	9	8	9	11
		CPU(s)	0.172	0.172	0.141	0.14	0.156
	P_{SHNS}	IT	39	40	38	33	28
		CPU(s)	0.547	0.578	0.531	0.469	0.406
32×32	P_{MPNS}	IT	17	12	10	14	18
		CPU(s)	15.031	11.391	9.656	12.766	17
	P_{SHNS}	IT	332	327	268	119	66
		CPU(s)	287.516	278.172	231.343	111.313	60.735

Table 6
The optimal values α^* of MPNS for solving the linear system (4.5).

$m \times m$	8×8	16×16	32×32
α^*	1.0484	1.0137	1.0037

Table 7
IT and CPU(s) for the MPNS method with the different frequency ω.

m	ω	1	10	50	100
8×8	IT	22	22	27	57
	CPU(s)	0.015	0.016	0.018	0.023
16×16	IT	21	21	21	22
	CPU(s)	0.063	0.063	0.062	0.078
32×32	IT	21	21	21	21
	CPU(s)	1.437	1.454	1.375	1.391

4.3. An application for parabolic equations

In this subsection, the above results in Section 3 will be extended to solve the complex symmetric linear system from the following discrete parabolic equations

$$\frac{\partial v}{\partial t} - \triangle v = f(x,t), \quad t > 0, \tag{4.3}$$

where $\triangle$ is the Laplacian operator and the forcing function is periodic in time, $f(x,t) = f_0(x,t)e^{i\omega t}$ with ω being the frequency in [19]. Applying the Ansatz $v(x,t) = u(x)e^{i\omega t}$, where u and v are complex-valued functions, from (4.3) we have

$$\frac{\partial u}{\partial t} - \triangle u + i\omega u = f_0(x,t). \tag{4.4}$$

Using an implicit time integration method for (4.4), we require to solve a system of the form

$$(K + i\omega\tau M)u = b, \tag{4.5}$$

where $K = M + \tau L$, L is the discrete Laplace operator, M is the identity matrix and τ is the time-step.

By investigating the numerical results in Sections 4.1–4.2, the MPNS method outperforms the SHNS method, and the efficiency of MPNS-preconditioner also outperforms that of the SHNS-preconditioner under certain conditions. In this case, to solve the linear systems (4.5). we only consider the efficiency of the MPNS method and the MPNS-preconditioner.

In our numerical computations, based on Corollary 3.2, the optimal iteration parameter $\alpha^* = \sqrt{\lambda_{\min}\lambda_{\max}}$ for the MPNS method for solving the linear system (4.5) is adopted. Specifically, see Table 6.

Table 7 lists the numerical results of the MPNS method for the complex symmetric linear system (4.5) when using the optimal iteration parameters in Table 6. From Table 7, we find that under the same time step-size τ, with ω increasing, the change trend of the iteration number of the MPNS method is increasing.

In Table 8, we also report the numerical results for GMRES(20) with the MPNS-preconditioner for solving the linear system (4.5). From Table 8, the MPNS-preconditioner is quite competitive in terms of convergence rate, robustness and efficiency when some Krylov subspace methods combining with the MPNS-preconditioner are applied to solve the linear

Table 8
IT and CPU(s) of GMRES(20) with the MPNS-preconditioner and the different frequency ω.

		α	0.01	0.05	0.1	0.5	1
8×8	$\omega = 1$	IT	3	3	3	3	3
		CPU(s)	0.015	0.016	0.016	0.015	0.015
	$\omega = 10$	IT	3	3	3	3	3
		CPU(s)	0.015	0.015	0.015	0.015	0.16
	$\omega = 50$	IT	3	3	3	3	3
		CPU(s)	0.015	0.015	0.016	0.016	0.015
	$\omega = 100$	IT	3	3	3	3	3
		CPU(s)	0.015	0.015	0.016	0.015	0.015
16×16	$\omega = 1$	IT	2	2	2	2	2
		CPU(s)	0.016	0.016	0.016	0.016	0.15
	$\omega = 10$	IT	2	2	2	2	2
		CPU(s)	0.016	0.016	0.015	0.015	0.015
	$\omega = 50$	IT	2	2	2	2	2
		CPU(s)	0.016	0.015	0.016	0.015	0.15
	$\omega = 100$	IT	3	3	3	2	2
		CPU(s)	0.031	0.031	0.031	0.031	0.032
32×32	$\omega = 1$	IT	2	2	2	2	2
		CPU(s)	0.61	0.562	0.531	0.578	0.515
	$\omega = 10$	IT	2	2	2	2	2
		CPU(s)	0.532	0.531	0.688	0.672	0.61
	$\omega = 50$	IT	2	2	2	2	2
		CPU(s)	0.625	0.562	0.547	0.594	0.563
	$\omega = 100$	IT	2	2	2	2	2
		CPU(s)	0.578	0.516	0.609	0.579	0.656

system (4.5) under certain conditions. The numerical results in Table 8 imply that the MPNS-preconditioner may be suitable for the large sparse linear system (4.5) from the discrete parabolic equations.

5. Conclusion

In this paper, a class of splitting iteration methods, namely, the MPNS method, has been established on the basis of positive/negative-stable splitting (PNS) of the coefficient matrix of a class of complex symmetric linear system. From the structure of the MPNS method, the MPNS method seems to be not relevant to the original Hermitian part, $-M + K$, of the complex symmetric linear system. Theorem 3.1 or 3.2 further shows that the convergence properties of the MPNS method do not depend on the eigenvalues and eigenvectors of the Hermitian part, $-M + K$. In other words, the MPNS method successfully avoids this situation where the original Hermitian part, $-M + K$, of the complex symmetric linear system is symmetric indefinite.

Theoretical analysis shows that the MPNS method is absolutely convergent under certain conditions. Numerical experiments are to illustrate that both the MPNS method and the MPNS preconditioner are feasible and efficient. In particular, the resulting MPNS preconditioner leads to fast convergence when it is used to preconditioned Krylov subspace methods such as GMRES.

References

[1] A. Feriani, F. Perotti, V. Simoncini, Iterative system solvers for the frequency analysis of linear mechanical systems, Comput. Methods Appl. Mech. Engrg. 190 (2000) 1719–1739.
[2] Z.-Z. Bai, M. Benzi, F. Chen, Modified HSS iteration methods for a class of complex symmetric linear systems, Computing 87 (2010) 93–111.
[3] Z.-Z. Bai, M. Benzi, F. Chen, On preconditioned MHSS iteration methods for complex symmetric linear systems, Numer. Algorithms 56 (2011) 297–317.
[4] A. Bayliss, C.I. Goldstein, E. Turkel, An iterative method for Helmholtz equation, J. Comput. Phys. 49 (1983) 443–457.
[5] C.-H. Guo, Incomplete block factorization preconditioner for linear systems arising in the numerical solution of the Helmholtz equation, Appl. Numer. Math. 19 (1996) 495–508.
[6] S.-L. Wu, T.-Z. Huang, L. Li, L.-L. Xiong, Positive stable preconditioners for symmetric indefinite linear systems arising from Helmholtz equations, Phys. Lett. A 373 (2009) 2401–2407.
[7] S.-L. Wu, C.-X. Li, A modified SSOR preconditioning strategy for Helmholtz equations, J. Appl. Math. 2012 (2012) 9. Article ID 365124.
[8] D.D. Day, M.A. Heroux, Solving complex-valued linear systems via equivalent real formulations, SIAM J. Sci. Comput. 23 (2001) 480–498.
[9] M. Benzi, D. Bertaccini, Block preconditioning of real-valued iterative algorithms for complex linear systems, IMA J. Numer. Anal. 28 (2008) 598–618.
[10] Z.-Z. Bai, On preconditioned iteration methods for complex linear systems, J. Eng. Math. 93 (2015) 41–60.
[11] H.A. van der Vorst, J.B.M. Melissen, A Petrov–Galerkin type method for solving $Ax = b$, where A is symmetric complex, IEEE Trans. Magn. 26 (1990) 706–708.
[12] A. Bunse-Gerstner, R. Stöver, On a conjugate gradient-type method for solving complex symmetric linear systems, Linear Algbra Appl. 287 (1999) 105–123.
[13] R.W. Freund, Conjugate Gradient-type methods for linear systems with complex symmetric coefficient matrices, SIAM J. Sci. Stat. Comput. 13 (1992) 425–448.

[14] S.-L. Wu, C.-X. Li, A splitting iterative method for the discrete dynamic linear systems, J. Comput. Appl. Math. 267 (2014) 49–60.
[15] O. Axelsson, A. Kucherov, Real valued iterative methods for solving complex symmetric linear systems, Numer. Linear Algebra Appl. 7 (2000) 197–218.
[16] R.W. Freund, On Conjugate Gradient type methods and polynomial preconditioners for a class of complex non-Hermitian matrices, Numer. Math. 57 (1990) 285–312.
[17] Z.-Z. Bai, G.H. Golub, M.K. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, SIAM J. Matrix Anal. Appl. 24 (2003) 603–626.
[18] Z.-Z. Bai, M. Benzi, F. Chen, Z.-Q. Wang, Preconditioned MHSS iteration methods for a class of block two-by-two linear systems with applications to distributed control problems, IMA J. Numer. Anal. 33 (2013) 343–369.
[19] O. Axelsson, M. Neytcheva, B. Ahmad, A comparison of iterative methods to solve complex valued linear algebraic systems, Numer. Algorithms 66 (2014) 811–841.
[20] Z.-Z. Bai, G.H. Golub, J.-Y. Pan, Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems, Numer. Math. 98 (2004) 1–32.
[21] Z.-Z. Bai, G.H. Golub, Accelerated Hermitian and skew-Hermitian splitting iteration methods for saddle-point problems, IMA J. Numer. Anal. 27 (2007) 1–23.
[22] S.-L. Wu, Several variants of the Hermitian and skew-Hermitian splitting method for a class of complex symmetric linear systems, Numer. Linear Algebra Appl. 22 (2015) 338–356.
[23] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
[24] G.I. Marchuk, Methods of Numerical Mathematics, Springer-Verlag, New York, 1984.
[25] M. Benzi, A generalization of the Hermitian and skew-Hermitian splitting iteration, SIAM J. Matrix Anal. Appl. 31 (2009) 360–374.
[26] X. Li, A.-L. Yang, Y.-J. Wu, Lopsided PMHSS iteration method for a class of complex symmetric linear systems, Numer. Algorithms 66 (2014) 555–568.
[27] Z.-Z. Bai, G.H. Golub, M.K. Ng, On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, Linear Algebra Appl. 428 (2008) 413–440.
[28] C.-X. Li, S.-L. Wu, A modified GHSS method for non-Hermitian positive definite linear systems, Japan J. Indust. Appl. Math. 29 (2012) 253–268.
[29] W. Li, Y.-P. Liu, X.-F. Peng, The generalized HSS method for solving singular linear systems, J. Comput. Appl. Math. 236 (2012) 2338–2353.
[30] M. Benzi, G.H. Golub, A preconditioner for generalized saddle point problems, SIAM J. Matrix Anal. Appl. 26 (2004) 20–41.