Japan J. Indust. Appl. Math. (2014) 31:265–278
DOI 10.1007/s13160-014-0140-x

ORIGINAL PAPER **Area 2**

A parameterized splitting iteration method for complex symmetric linear systems

Guo-Feng Zhang · Zhong Zheng

Received: 21 September 2012 / Revised: 12 November 2012 / Published online: 29 January 2014

Abstract In this paper, we propose a parameterized splitting **(PS)** iteration method for solving complex symmetric linear systems. The convergence theory of the method is established and the spectral properties of the corresponding iteration matrix are analyzed. The explicit expression for the spectral radius of the iteration matrix is given. In addition, the optimal choice of the iteration parameter is discussed. It is shown that the eigenvalues of the preconditioned matrix are cluster at 1. Numerical experiments illustrate the theoretical results and also examine the numerical effectiveness of the new parameterized splitting iteration method served either as a preconditioner or as a solver.

Keywords Complex symmetric linear systems · PMHSS iteration method · GMRES · Spectral properties · Preconditioning

Mathematics Subject Classification (2000) 65F08 · 65F10 · 65F50 · 65N22

1 Introduction

Consider the iteration solution of the linear system

$$\mathscr{A}\mathbb{u} = \mathbb{b}, \qquad \mathscr{A} = \mathbf{W} + \imath\mathbf{T} \in \mathbb{C}^{n\times n}, \tag{1.1}$$

This work was supported by the National Natural Science Foundation (11271174).

G.-F. Zhang (✉) · Z. Zheng
School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000,
People's Republic of China
e-mail: gf_zhang@lzu.edu.cn

where $\mathbb{u}, \mathbb{b} \in \mathbb{C}^n$, and $\mathbf{W}, \mathbf{T} \in \mathbb{R}^{n\times n}$ are real, symmetric and positive semi-definite matrices with at least one of them, e.g., $\mathbf{W}$ being positive definite. Here and in the sequel, we use $\imath = \sqrt{-1}$ to denote the imaginary unit.

Complex symmetric linear systems of this kind arise in many problems in scientific computing and engineering applications, including diffuse optical tomography [1], FFT-based solution of certain time-dependent PDEs [11], quantum mechanics [15], molecular scattering [13], structural dynamics [12], and lattice quantum chromo dynamics [13] and so on. For more examples about the practical backgrounds of this class of problems, we refer to [8] and the references therein.

Let $\mathbb{u} = \mathbf{x}+\imath\mathbf{y}$, $\mathbb{b} = \mathbf{f}+\imath\mathbf{g}$, $\mathbf{x}, \mathbf{y}, \mathbf{f}, \mathbf{g} \in \mathbb{R}^n$. Then (1.1) can be equivalently written as

$$(\mathbf{W} + \imath\mathbf{T})(\mathbf{x} + \imath\mathbf{y}) = \mathbf{f} + \imath\mathbf{g},$$

or

$$(\mathbf{W}\mathbf{x} - \mathbf{T}\mathbf{y}) + \imath(\mathbf{W}\mathbf{y} + \mathbf{T}\mathbf{x}) = \mathbf{f} + \imath\mathbf{g}.$$

That is to say, $\mathbf{W}\mathbf{x} - \mathbf{T}\mathbf{y} = \mathbf{f}$ and $\mathbf{T}\mathbf{x} + \mathbf{W}\mathbf{y} = \mathbf{g}$. Therefore the complex symmetric linear systems (1.1) can be equivalently transformed into the following two-by-two block form

$$\mathbf{A}\mathbf{u} \equiv \begin{pmatrix} \mathbf{W} & -\mathbf{T} \\ \mathbf{T} & \mathbf{W} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix} \equiv \mathbf{b}; \tag{1.2}$$

see [7].

Linear system (1.2) can be formally regarded as a special case of the generalined saddle-point problem. In many cases $\mathbf{W}$ and $\mathbf{T}$ are large sparse matrices and iterative techeniques are preferable for solving (1.2). In order to solve (1.2) more effectively using the iteration methods, usually efficient splitting of the coefficient matrix $\mathbf{A}$ determines the asymptotic convergence rate of the corresponding iteration method. Many efficient iterative methods have been proposed in the literature, see [3–7,9,10,17]. One can see [8] for a comprehensive survey.

By making use of the special structure of the coefficient matrix $\mathscr{A} \in \mathbb{C}^{n\times n}$, Bai et al. [4] proposed the modified Hermitian and skew-Hermitian splitting (MHSS) iteration method:

MHSS Iteration Method: Given an initial guess $\mathbf{u}^{(0)}$, for $k = 0, 1, 2, \cdots$ until $\mathbf{u}^{(k)}$ converges, compute

$$\begin{cases} (\alpha\mathbf{I} + \mathbf{W})\mathbb{u}^{\left(k+\frac{1}{2}\right)} = (\alpha\mathbf{I} - \imath\mathbf{T})\mathbb{u}^{(k)} + \mathbb{b}, \\ (\alpha\mathbf{I} + \mathbf{T})\mathbb{u}^{(k+1)} = (\alpha\mathbf{I} + \imath\mathbf{W})\mathbb{u}^{\left(k+\frac{1}{2}\right)} - \imath\mathbb{b}, \end{cases}$$

α is a given positive constant and $\mathbf{I} \in \mathbb{R}^{n\times n}$ is a unit matrix. Recently, Bai et al. [5] further presented preconditioned MHSS **(PMHSS) iteration method**:

$$\begin{cases} (\alpha \mathbf{V} + \mathbf{W})\mathbf{u}^{\left(k+\frac{1}{2}\right)} = (\alpha \mathbf{V} - \imath \mathbf{T})\mathbf{u}^{(k)} + \mathbf{b}, \\ (\alpha \mathbf{V} + \mathbf{T})\mathbf{u}^{(k+1)} = (\alpha \mathbf{V} + \imath \mathbf{W})\mathbf{u}^{\left(k+\frac{1}{2}\right)} - \imath \mathbf{b}, \end{cases} \tag{1.3}$$

and PMHSS iteration method (Real Version [7]).

In this paper, by making use of the special structure of the coefficient matrix $\mathscr{A} \in \mathbb{C}^{n\times n}$, a new parameterized splitting (PS) iteration method is presented and some of its basic properties is studied. The explicit expression for the spectral radius of the iteration matrix is given and the optimal choice of the iteration parameter is discussed. An advantage of the PS method is that the solution of the linear system with complex coefficient matrix is avoided and only two linear sub-systems with real and symmetric positive definite coefficient matrix need to be solved at each step.

The remainder of the paper is organized as follows. In Sect. 2, the PS method is described. The convergence theory of the PS method is established. The spectral properties of the corresponding iteration matrix are discussed, and the explicit expression for the spectral radius of the iteration matrix is given. In addition, the optimal choice of the iteration parameter is discussed. In Sect. 3, some implementation aspects are briefly discussed. Some properties of the preconditioned matrix are analyzed. It is shown that the eigenvalues of the preconditioned matrix are cluster at 1. Results of the numerical experiments on a few model problems are given in Sect. 4. Finally, in Sect. 5 we offer brief concluding remarks to end the paper.

2 The PS iteration method

We split the coefficient matrix $\mathbf{A}$ in (1.2) into the sum of the following matrices:

$$\mathbf{A} = \mathbf{M}(\alpha) - \mathbf{N}(\alpha), \tag{2.1}$$

where

$$\mathbf{M}(\alpha) := \begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix}, \quad \mathbf{N}(\alpha) := \begin{pmatrix} 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & \mathbf{T} \\ -\alpha\mathbf{T} & -\frac{1}{2}\mathbf{W} \end{pmatrix}. \tag{2.2}$$

Based on this splitting, we establish the following parameterized splitting (PS) iteration method for the linear systems (1.1):

The PS Iteration Method: Let $\mathbf{u}^{(0)} = \left(\mathbf{x}^{(0)^T}, \mathbf{y}^{(0)^T}\right)^T \in \mathbb{R}^{2n}$ be an arbitrary initial guess. For $k = 0, 1, 2, \cdots$ until the iteration sequence $\left\{\left(\mathbf{x}^{(k)^T}, \mathbf{y}^{(k)^T}\right)^T\right\}_{k=0}^{\infty}$ converge, compute the next iteration $\left(\mathbf{x}^{(k+1)^T}, \mathbf{y}^{(k+1)^T}\right)^T$ according to the following procedure:

$$\begin{cases} (\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\mathbf{x}^{(k+1)} = \mathbf{f} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{x}^{(k)} + \mathbf{T}\mathbf{y}^{(k)}, \\ \mathbf{y}^{(k+1)} = -\mathbf{y}^{(k)} + 2\mathbf{W}^{-1}\left(\mathbf{g} - \alpha\mathbf{T}\mathbf{x}^{(k)} - (1-\alpha)\mathbf{T}\mathbf{x}^{(k+1)}\right). \end{cases} \tag{2.3}$$

where α is a given constant.

The PS iteration method (2.3) can be rewritten in matrix-vector form as

$$\mathbf{u}^{(k+1)} = \mathbf{L}(\alpha)\mathbf{u}^{(k)} + \mathbf{M}^{-1}(\alpha)\mathbf{b}, \qquad k = 0, 1, 2, \cdots \tag{2.4}$$

where $\mathbf{L}(\alpha) = \mathbf{M}^{-1}(\alpha)\mathbf{N}(\alpha)$, $\mathbf{u}^{(k)} = \left(\mathbf{x}^{(k)^T}, \mathbf{y}^{(k)^T}\right)^T$ and α is a given positive constant, or equivalently,

$$\begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \mathbf{x}^{(k+1)} \\ \mathbf{y}^{(k+1)} \end{pmatrix} = \begin{pmatrix} 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & \mathbf{T} \\ -\alpha\mathbf{T} & -\frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \mathbf{x}^{(k)} \\ \mathbf{y}^{(k)} \end{pmatrix} + \begin{pmatrix} \mathbf{f} \\ \mathbf{g} \end{pmatrix}.$$

Obviously,

$$\mathbf{L}(\alpha) = \begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}T & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix}^{-1} \begin{pmatrix} 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & \mathbf{T} \\ -\alpha\mathbf{T} & -\frac{1}{2}\mathbf{W} \end{pmatrix} \tag{2.5}$$

is the iteration matrix of the PS iteration method (2.3). Therefore, the PS iteration scheme is induced by the matrix splitting $\mathbf{A} = \mathbf{M}(\alpha) - \mathbf{N}(\alpha)$, $\mathbf{M}(\alpha)$ and $\mathbf{N}(\alpha)$ defined in (2.2). It follows that the splitting matrix $\mathbf{M}(\alpha)$ can be used as a preconditioner for the coefficient matrix $\mathbf{A} \in \mathbb{R}^{2n\times 2n}$ in (1.2).

The following Lemma 2.1 give out the condition that the coefficient matrix $\mathbf{A}$ is nonsingular.

Lemma 2.1 **([7])** *Let $\mathbf{A} \in \mathbb{R}^{2n\times 2n}$ be the block two-by-two matrix defined as in (1.2), with both $\mathbf{W} \in \mathbb{R}^{n\times n}$ and $\mathbf{T} \in \mathbb{R}^{n\times n}$ being symmetric positive semi-definite matrices. Then $\mathbf{A}$ is nonsingular if and only if null($\mathbf{W}$)$\cap$ null($\mathbf{T}$) $= \{0\}$.*

Lemma 2.2 **([16])** *Consider the real quadratic equation $x^2 - \delta_1 x + \delta_2 = 0$, where δ_1 and δ_2 are real numbers. Both roots of the equation are less than one in modules if and only if $|\delta_2| < 1$ and $|\delta_1| < 1 + \delta_2$.*

Next, we discuss the convergence of the PS iteration method.

Lemma 2.3 *Let both $\mathbf{W} \in \mathbb{R}^{n\times n}$ and $\mathbf{T} \in \mathbb{R}^{n\times n}$ be symmetric and positive definite matrices. Assume that λ is an eigenvalue of the iteration matrix $\mathbf{L}(\alpha)$ defined as in (2.5), and $\left(\mathbf{v}_1^T, \mathbf{v}_2^T\right)^T$ is the corresponding eigenvector. If $\alpha > \frac{1}{2}$, then $\lambda \neq -1$ and $\mathbf{v}_1 \neq 0$. Moreover, if $\mathbf{v}_2 = 0$, then $|\lambda| < 1$.*

Proof Since $\left(\lambda, \left(\mathbf{v}_1^T, \mathbf{v}_2^T\right)^T\right)$ is an eigenpair of the matrix $\mathbf{L}(\alpha)$. Then we have

$$\begin{pmatrix} 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & \mathbf{T} \\ -\alpha\mathbf{T} & -\frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix},$$

or equivalently

$$\begin{cases} 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{v}_1 + \mathbf{T}\mathbf{v}_2 = \lambda(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\mathbf{v}_1, \\ (\lambda+1)\mathbf{W}\mathbf{v}_2 = 2(\lambda\alpha - \lambda - \alpha)\mathbf{T}\mathbf{v}_1. \end{cases} \tag{2.6}$$

If $\lambda = -1$, from (2.6) we obtain

$$\begin{cases} (\mathbf{W} + 4\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\mathbf{v}_1 + \mathbf{T}\mathbf{v}_2 = 0, \\ (1-2\alpha)\mathbf{T}\mathbf{v}_1 = 0. \end{cases} \tag{2.7}$$

Since $\alpha > \frac{1}{2}$, $\mathbf{W}$ and $\mathbf{T}$ are symmetric and positive definite matrices, from (2.7) it follows that $\mathbf{v}_1 = 0$ and $\mathbf{v}_2 = 0$, which contradicts with the assumption that $\left(\mathbf{v}_1^T, \mathbf{v}_2^T\right)^T$ is the eigenvector of $\mathbf{L}(\alpha)$.

If $\mathbf{v}_1 = 0, \lambda \neq -1$, from (2.6) we have $(\lambda+1)\mathbf{W}\mathbf{v}_2 = 0$. It follows that $\mathbf{v}_2 = 0$, which is a contradiction.

If $\mathbf{v}_2 = 0$. Define

$$\sigma_1 := \frac{\mathbf{v}_1^T\mathbf{W}\mathbf{v}_1}{\mathbf{v}_1^T\mathbf{v}_1}, \qquad \sigma_2 := \frac{\mathbf{v}_1^T\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{v}_1}{\mathbf{v}_1^T\mathbf{v}_1}. \tag{2.8}$$

We easily see that $\sigma_1 > 0$ and $\sigma_2 > 0$.

From (2.6) we obtain

$$2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{v}_1 = \lambda(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\mathbf{v}_1. \tag{2.9}$$

By left-multiplying $\frac{\mathbf{v}_1^T}{\mathbf{v}_1^T\mathbf{v}_1}$ on both sides in (2.9) we have

$$2\alpha\sigma_2 = \lambda(\sigma_1 + 2\alpha\sigma_2).$$

Hence, $\lambda = \frac{2\alpha\sigma_2}{\sigma_1 + 2\alpha\sigma_2}$. Therefore we see that $|\lambda| < 1$ holds when $\alpha > 0$.

Applying Lemmas 2.2 and 2.3, we can obtain the convergence result in the following theorem.

Theorem 2.4 *Let $\mathbf{A} \in \mathbb{R}^{2n\times 2n}$ be the block two-by-two matrix defined as in (1.2), with both $\mathbf{W} \in \mathbb{R}^{n\times n}$ and $\mathbf{T} \in \mathbb{R}^{n\times n}$ being symmetric positive definite. If $\alpha > \frac{1}{2}$, then the PS iteration method (2.3) is convergent. Moreover, the spectral radius $\rho(\mathbf{L}(\alpha))$ of the PS iteration matrix has the following explicit expression:*

$$\rho(\mathbf{L}(\alpha)) = \left|\frac{2\alpha\sigma_2 - 2\sigma_2 - \sigma_1}{\sigma_1 + 2\alpha\sigma_2}\right|,$$

where σ_1 and σ_2 defined as in (2.8)

Proof Let λ be an eigenvalue of the iteration matrix $\mathbf{L}(\alpha)$ and $(\mathbf{v}_1^T, \mathbf{v}_2^T)^T$ be the corresponding eigenvector. From Lemma 2.2 we know that $\lambda \neq -1$. Since $\mathbf{W}$ is a symmetric positive definite matrix, we see that $(\lambda+1)\mathbf{W}$ is nonsingular. Hence, from the second equation in (2.6) we obtain

$$\mathbf{v}_2 = \frac{2(\lambda\alpha - \lambda - \alpha)}{\lambda + 1}\mathbf{W}^{-1}\mathbf{T}.$$

From the first equation in (2.6) we have

$$2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{v}_1 + \frac{2(\lambda\alpha - \lambda - \alpha)}{\lambda + 1}\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\mathbf{v}_1 = \lambda(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\mathbf{v}_1, \quad (2.10)$$

which, left multiplication $\dfrac{\mathbf{v}_1^T}{\mathbf{v}_1^T\mathbf{v}_1}$ and together with the definition of σ_1 and σ_2, leads to

$$\lambda^2(\sigma_1 + 2\alpha\sigma_2) - \lambda(2\alpha\sigma_2 - 2\sigma_2 - \sigma_1) = 0. \quad (2.11)$$

Both roots of the the quadratic equation (2.11) are $\lambda_1 = 0$ and $\lambda_2 = \dfrac{2\alpha\sigma_2 - 2\sigma_2 - \sigma_1}{\sigma_1 + 2\alpha\sigma_2}$. Noting that $\alpha > \dfrac{1}{2}$, $\sigma_1 > 0$, and $\sigma_2 > 0$, by simple computations we obtain

$$\rho(\mathbf{L}(\alpha)) = \left|\frac{2\alpha\sigma_2 - 2\sigma_2 - \sigma_1}{\sigma_1 + 2\alpha\sigma_2}\right| < 1,$$

i.e., the PS iteration method converges to the unique solution $\mathbb{u}_\star = \mathbf{x}_\star + \imath\mathbf{y}_\star \in \mathbb{C}^n$ of the complex symmetric linear systems (1.1) for any initial guess.

Theorem 2.4 gives the explicit expression for the spectral radius of the iteration matrix $L(\alpha)$. It shows that the eigenvalues of the iteration matrix of the PS iteration method are real and nonnegative.

Next, we discuss how to choose the iteration parameter α. For the optimal choice α_{opt} of the parameter α, we have the following result.

Theorem 2.5 *Suppose that the conditions of Theorem 2.4 are satisfied. Let a_1, a_n and b_1, b_n be the largest and the smallest eigenvalues of the matrix $\mathbf{W}$ and $\mathbf{T}\mathbf{W}^{-1}\mathbf{T}$, respectively. Then if we choose*

$$\alpha = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{(b_n + a_1)(b_1 + a_n)}{b_1 b_n}},$$

then

$$\rho(L(\alpha)) = \frac{b_1 a_1 - b_n a_n}{\left(\sqrt{b_1 b_n + b_1 a_1} + \sqrt{b_1 b_n + b_1 a_n}\right)^2} = min.$$

That is to say, the optimal parameter $\alpha_{opt} = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{(b_n + a_1)(b_1 + a_n)}{b_1 b_n}}$.

Proof Let $\sigma := \frac{\sigma_2}{\sigma_1}$. Denote by $b_1 \geq b_2 \geq \cdots \geq b_n > 0$ and $a_1 \geq a_2 \geq \cdots \geq a_n > 0$ the eigenvalues of the matrice $\mathbf{TW}^{-1}\mathbf{T}$ and $\mathbf{W}$, respectively. By using of the Rayleigh quotient principle we can know that $0 < \frac{b_n}{a_1} \leq \sigma \leq \frac{b_1}{a_n}$.

Define

$$f(\alpha, \sigma) := \frac{2\alpha\sigma - 2\sigma - 1}{1 + 2\alpha\sigma},$$

where $\alpha > \frac{1}{2}$ and $\sigma \in \left[\frac{b_n}{a_1}, \frac{b_1}{a_n}\right]$. By straightforward computations we have

$$\frac{\partial f}{\partial \sigma} = \frac{(2\alpha - 2)(1 + 2\alpha\sigma) - 2\alpha(2\alpha\sigma - 2\sigma - 1)}{(1 + 2\alpha\sigma)^2} = \frac{2(2\alpha - 1)}{(1 + 2\alpha\sigma)^2}.$$

As $\alpha > \frac{1}{2}$, we know that $f_\sigma(\alpha, \sigma) > 0$. It means that $f(\alpha, \sigma)$ is a strictly increase function for σ when $\alpha > \frac{1}{2}$. Therefore, if the condition $\left|f\left(\alpha, \frac{b_n}{a_1}\right)\right| = \left|f\left(\alpha, \frac{b_1}{a_n}\right)\right|$ holds, then we minimize the spectral radius of the iteration matrix $L(\alpha)$. If $\left|f\left(\alpha, \frac{b_n}{a_1}\right)\right| = \left|f\left(\alpha, \frac{b_1}{a_n}\right)\right|$, we obtain

$$f\left(\alpha, \frac{b_n}{a_1}\right) = -f\left(\alpha, \frac{b_1}{a_n}\right).$$

After straightforward computations we get

$$4\alpha^2 \frac{b_1 b_n}{a_1 a_n} - 4\alpha \frac{b_1 b_n}{a_1 a_n} = \frac{b_1}{a_n} + \frac{b_n}{a_1} + 1.$$

It follows that

$$\alpha_{opt} = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{(b_n + a_1)(b_1 + a_n)}{b_1 b_n}}.$$

Therefore, it holds that

$$\rho(L(\alpha_{opt})) = \frac{b_1 a_1 - b_n a_n}{\left(\sqrt{b_1 b_n + b_1 a_1} + \sqrt{b_1 b_n + b_1 a_n}\right)^2}.$$

Evidently,

$$\rho(L(\alpha_{opt})) < \left(\frac{\sqrt{a_1}}{\sqrt{b_n + a_1} + \sqrt{b_1 + a_n}}\right)^2 < 1.$$

3 Krylov subspace acceleration

Even with the optimal choice α_{opt} of the parameter α, the convergence of the stationary iteration is typically too slow for the method to compete. For this reason, we propose using the preconditioned Krylov subspace method like GMRES or its restarted version $GMRES(\natural)$ to accelerate the convergence of the PS iteration method.

The PS method may also result from the matrix splitting $\mathbf{A} = \mathbf{M}(\alpha) - \mathbf{N}(\alpha)$, with $\mathbf{A}$, $\mathbf{M}(\alpha)$ and $\mathbf{N}(\alpha)$ being defined as in (2.1)–(2.2). It then follows that the splitting matrix $\mathbf{M}(\alpha)$ can be used as a preconditioner $\mathbf{A}$. The PS iteration method can be rewritten in correct form as:

$$\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \mathbf{M}(\alpha)^{-1}\mathbf{r}^{(k)}, \qquad \mathbf{r}^{(k)} := \mathbf{b} - \mathbf{A}\mathbf{u}^{(k)}.$$

Then the linear systems (1.2) are equivalent to the following linear systems

$$(\mathbf{I} - \mathbf{L}(\alpha))\mathbf{u} = \mathbf{M}^{-1}(\alpha)\mathbf{A}\mathbf{u} = \mathbf{c},$$

where $\mathbf{c} = \mathbf{M}(\alpha)^{-1}\mathbf{b}$. This equivalent (left-preconditioned) system can be solved with GMRES. Hence, the matrix $\mathbf{M}(\alpha)$ can be seen as a preconditioner for GMRES method. Equivalently, we can say that GMRES method is used to accelerate the convergence of the splitting iteration method applied to $\mathscr{A}\mathbb{u} = \mathbb{b}$.

Application of the preconditioner $\mathbf{M}(\alpha)$ with GMRES requires solving a linear system of the form

$$\mathbf{M}(\alpha)\overline{\mathbf{u}} = \begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \overline{\mathbf{x}} \\ \overline{\mathbf{y}} \end{pmatrix} = \begin{pmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \end{pmatrix},$$

at each iteration step, where $\mathbf{r} = \left(\mathbf{r}_1^T, \mathbf{r}_2^T\right)^T$ represents the current residual vector and $\overline{\mathbf{u}} = \left(\overline{\mathbf{x}}^T, \overline{\mathbf{y}}^T\right)^T$ represents the generalized residual vector. Note that the matrix $(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})$ is symmetric positive definite. This can be done by first solving

$$(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\overline{\mathbf{x}} = \mathbf{r}_1$$

with the conjugate gradient(**CG**) or preconditioned **CG** (**PCG**) obtain $\overline{\mathbf{x}}^T$, then we get $\overline{\mathbf{y}}^T$ from the following equality

$$\overline{\mathbf{y}} = 2\mathbf{W}^{-1}(\mathbf{r}_2 - (1-\alpha)\mathbf{T}\overline{\mathbf{x}}) = 2\mathbf{W}^{-1}\mathbf{r}_2 - 2(1-\alpha)\mathbf{W}^{-1}\mathbf{T}\overline{\mathbf{x}}.$$

The spectral properties of the preconditioning matrix $\mathbf{M}(\alpha)^{-1}\mathbf{A}$ are established in the following theorem.

Theorem 3.1 *Suppose that the conditions of Theorem 2.5 are satisfied.* $\mathbf{M}(\alpha)$ *is the preconditioner defined as in (2.2). Let* λ *be the eigenvalue of the matrix* $\mathbf{M}(\alpha)^{-1}\mathbf{A}$ *and* $(\tilde{\mathbf{u}}^T, \tilde{\mathbf{v}}^{\mathbf{T}})^T$ *be the corresponding eigenvector. Then the eigenvalues of the preconditioned matrix* $\mathbf{M}(\alpha)^{-1}\mathbf{A}$ *are* 1 *(with algebraic multiplicity n) and* $\lambda = \dfrac{2\sigma_3 + 2\sigma_4}{\sigma_3 + 2\alpha\sigma_4}$, *where*

$$\sigma_3 = \frac{\tilde{\mathbf{u}}^T\mathbf{W}\tilde{\mathbf{u}}}{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}}}, \qquad \sigma_4 = \frac{\tilde{\mathbf{u}}^T\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\tilde{\mathbf{u}}}{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}}}.$$

Proof Since λ is an eigenvalue of the matrix $\mathbf{M}(\alpha)^{-1}\mathbf{A}$ and $(\tilde{\mathbf{u}}^T, \tilde{\mathbf{v}}^{\mathbf{T}})^T$ is the corresponding eigenvector. Then we have

$$\begin{pmatrix} \mathbf{W} & -\mathbf{T} \\ \mathbf{T} & \mathbf{W} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{v}} \end{pmatrix} = \lambda \begin{pmatrix} \mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T} & 0 \\ (1-\alpha)\mathbf{T} & \frac{1}{2}\mathbf{W} \end{pmatrix} \begin{pmatrix} \tilde{\mathbf{u}} \\ \tilde{\mathbf{v}} \end{pmatrix},$$

or equivalently

$$\begin{cases} \mathbf{W}\tilde{\mathbf{u}} - \mathbf{T}\tilde{\mathbf{v}} = \lambda(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\tilde{\mathbf{u}}, \\ \mathbf{T}\tilde{\mathbf{u}} + \mathbf{W}\tilde{\mathbf{v}} = \lambda(1-\alpha)\mathbf{T}\tilde{\mathbf{u}} + \frac{1}{2}\lambda\mathbf{W}\tilde{\mathbf{v}}. \end{cases} \tag{3.1}$$

From the second equality in 3.1 we have

$$\tilde{\mathbf{v}} = \frac{2(1-\lambda+\lambda\alpha)}{\lambda-2}\mathbf{W}^{-1}\mathbf{T}\tilde{\mathbf{u}}. \tag{3.2}$$

Substituting (3.2) into the first equality in (3.1) we obtain

$$\mathbf{W}\tilde{\mathbf{u}} - \frac{2(1-\lambda+\lambda\alpha)}{\lambda-2}\mathbf{T}\mathbf{W}^{-1}\mathbf{T}\tilde{\mathbf{u}} = \lambda(\mathbf{W} + 2\alpha\mathbf{T}\mathbf{W}^{-1}\mathbf{T})\tilde{\mathbf{u}}. \tag{3.3}$$

Which, left multiplication $\dfrac{\tilde{\mathbf{u}}^T}{\tilde{\mathbf{u}}^T\tilde{\mathbf{u}}}$ and together with the definition of σ_3 and σ_4, leads to

$$\lambda^2(\sigma - 3 + 2\alpha\sigma - 4) - \lambda(2\alpha\sigma - 4 + 3\sigma_3 + 2\sigma_4) + (2\sigma_4 + 2\sigma_3) = 0. \tag{3.4}$$

Both roots of the quadratic equation (3.4) are $\lambda = 1$ and $\lambda = \dfrac{2\sigma_3 + 2\sigma_4}{\sigma_3 + 2\alpha\sigma_4}$. This completes the proof.

From Theorem 3.1 we can see that if we choose $\alpha = 1 + \dfrac{\sigma_3}{2\sigma_4}$, then the eigenvalues of the preconditioned matrix $\mathbf{M}(\alpha)^{-1}\mathbf{A}$ are cluster at 1.

4 Numerical experiments

In this section we use two numerical examples from [2–5] to assess the feasibility and effectiveness of the PS and PS-preconditioned method in terms of the iteration count (denoted as IT), the relative residual error (denoted as RES) and the computing time (in seconds, denoted as CPU), when it is employed either as a solver or as a preconditioner for solving the system of linear equations (1.1). Besides comparing the efficiency of the PS with the PMHSS methods which refer to (1.2), we also examine its numerical behavior as preconditioners for the (full) GMRES method and its restarted variants, say, GMRES(♯). All runs are started from the initial vector $\left(\mathbf{x}^{(0)^T}, \mathbf{y}^{(0)^T}\right)^T = \mathbf{0}$, and the iteration is terminated once the current iteration satisfying

$$\text{RES} = \frac{\sqrt{\|\mathbf{f} - \mathbf{W}\mathbf{x}^{(k)} + \mathbf{T}\mathbf{y}^{(k)}\|^2 + \|\mathbf{g} - \mathbf{T}\mathbf{x}^{(k)} - \mathbf{W}\mathbf{y}^{(k)}\|^2}}{\sqrt{\|\mathbf{f}\|^2 + \|\mathbf{g}\|^2}} \leq 10^{-6}.$$

Example 4.1 ([2–5]) The system of linear equations (1.1) is of the form

$$\left[\left(\mathbf{K} + \frac{3-\sqrt{3}}{\tau}\mathbf{I}\right) + \imath\left(\mathbf{K} + \frac{3+\sqrt{3}}{\tau}\mathbf{I}\right)\right]\mathbb{u} = \mathbb{b},$$

where τ is the time step-size and $\mathbf{K}$ is the five-point centered difference matrix approximating the negative Laplacian operator $L = -\Delta$ with homogenous Dirichlet boundary conditions, on a uniform mesh in the unit square $[0, 1] \times [0, 1]$ with the mesh-size $h = \frac{1}{m+1}$. The matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ possesses the tensor-product form $\mathbf{K} = \mathbf{I} \otimes \mathbf{B}_m + \mathbf{B}_m \otimes \mathbf{I}$, with $\mathbf{B}_m = h^{-2}tridiag(-1, 2, -1) \in \mathbb{R}^{m\times m}$. Hence, $\mathbf{K}$ is an $n \times n$ block-tridiagonal matrix, with $n = m^2$. We take

$$\mathbf{W} = \mathbf{K} + \frac{3-\sqrt{3}}{\tau}\mathbf{I}, \qquad \mathbf{T} = \mathbf{K} + \frac{3+\sqrt{3}}{\tau}\mathbf{I},$$

$\tau = h$ and the right-hand side vector $\mathbb{b}$ with its jth entry $[\mathbb{b}]_j$ being given by

$$[\mathbb{b}]_j = \frac{(1-\imath)j}{\tau(j+1)^2}, \qquad j = 1, 2, \cdots, n.$$

This complex symmetric system of linear equations arises in centered difference discretizations of the $R_{22} - P\acute{a}de$ approximations in the time integration of parabolic partial differential equations [2]. For more details, we refer to [2].

Here and the next tables the iteration parameters α_{exp} used in PMHSS iteration method as well as the corresponding PMHSS preconditioner are chosen according to the strategy described in [5]. For Example 1, the optimial iteration parameter $\alpha_{opt} = \frac{1}{2} + \frac{1}{2}\sqrt{\frac{(b_n + a_1)(b_1 + a_n)}{b_1 b_n}} = 1.23$ for PS iteration method. However, taking $\alpha = \alpha_{exp}$ in the neighborhood of α_{opt} is normally a very good choice in practice.

Table 1 IT, CPU and ERR(1e-7) for PS, PMHSS, GMRES and GMRES(20) methods for Example 4.1

Method	m	$m = 16$	$m = 24$	$m = 32$	$m = 48$	$m = 64$
PS	$\alpha_{\exp}$	1.25	1.25	1.25	1.25	1.25
	IT	9	10	10	10	11
	CPU	0.0036	0.0205	0.0599	0.2756	0.9595
	ERR	7.3	1.7	2.3	3.1	7.8
PMHSS	$\alpha_{\exp}$	1.09	1.28	1.36	1.45	1.35
	IT	21	21	21	21	21
	CPU	0.0187	0.0460	0.1441	0.6850	2.0045
	ERR	7.7	7.0	7.6	7.0	7.5
GMRES	IT	25	27	29	37	48
	CPU	0.0794	0.1812	0.4676	1.2869	3.3239
	ERR	9.7	9.9	9.9	9.9	9.9
GMRES(20)	IT	47	71	101	138	167
	CPU	0.0697	0.1720	0.4209	1.1440	2.1975
	ERR	9.9	9.8	9.9	9.9	9.9

Numerical results for Example 4.1 are listed in Tables 1 and 2. In Table 1, we show the numbers of iteration steps (IT), the computing times (CPU) and the relative residual errors (RES) for the PS method, the PMHSS method, the GMRES and the GMRES(20), when the problem sizes are equal to $16 \times 16, 24 \times 24, 32 \times 32, 48 \times 48, 64 \times 64$, respectively. In Table 2, we list the results for PS- and PMHSS-preconditioned GMRES and GMRES(10) methods, respectively.

From Table 1 we see that the iteration counts of the GMRES and GMRES(20) methods grow rapidly with problem size, while that of PS and PMHSS methods almost remain constantly. In other words the PS and the PMHSS methods are almost independent of the problem sizes. Moreover, as a solver, PS method costs much less iteration step and CPU time than PMHSS method, GMRES and GMRES(20).

In Table 2 we report the numerical results for preconditioned GMRES and GMRES(10) with PS and PMHSS. From the Table we observe that when $\mathbf{M}(\alpha)$ used as a preconditioner, PS performs better than PMHSS both in iteration counts and CPU times. Moreover, the iteration steps of PS-preconditioned keep a constant.

Example 4.2 ([4]) The systems of linear equations (1.1) is of the form

$$(\mathbf{W} + i\mathbf{T})\mathbb{u} = \mathbb{b},$$

with

$$\mathbf{T} = \mathbf{I} \otimes \mathbf{B} + \mathbf{B} \otimes \mathbf{I} \quad \text{and} \quad \mathbf{W} = 10(\mathbf{I} \otimes \mathbf{B}_c + \mathbf{B}_c \otimes \mathbf{I}) + 9\left(\mathbf{e}_1\mathbf{e}_m^T + \mathbf{e}_m\mathbf{e}_1^T\right) \otimes \mathbf{I},$$

where $\mathbf{B} = tridiag(-1, 2, -1) \in \mathbb{R}^{m\times m}$, $\mathbf{B}_c = \mathbf{B} - \mathbf{e}_1\mathbf{e}_m^T - \mathbf{e}_m\mathbf{e}_1^T \in \mathbb{R}^{m\times m}$, and $\mathbf{e}_1$ and $\mathbf{e}_m$ are the first and the mth unit basis vectors in $\mathbb{R}^m$, respectively. We take the right-

Table 2 IT, CPU and ERR(1e-7) for preconditioned GMRES and GMRES(10) for Example 4.1

Method	Preconditioner	m	$m = 16$	$m = 24$	$m = 32$	$m = 48$	$m = 64$
GMRES	PS	α_{exp}	0.515	0.515	0.515	0.515	0.515
		IT	2	2	2	2	2
		CPU	0.0071	0.0441	0.1807	0.6384	1.2370
		ERR	1.2	1.4	8.4	3.3	2.5
GMRES	PMHSS	α_{exp}	0.52	1.05	1.84	1.55	1.50
		IT	6	7	7	8	8
		CPU	0.0175	0.0482	0.1524	0.7400	1.8935
		ERR	7.0	6.7	7.2	5.4	6.6
GMRES(10)	PS	α_{exp}	0.508	0.508	0.508	0.508	0.508
		IT	3	3	3	3	3
		CPU	0.0056	0.0375	0.1648	0.6228	1.0170
		ERR	5.3	7.7	9.3	1.0	3.4
GMRES(10)	PMHSS	α_{exp}	0.56	0.90	1.64	1.84	3.67
		IT	6	7	7	7	8
		CPU	0.0175	0.0484	0.1528	0.7426	1.8842
		ERR	7.1	6.7	7.4	6.4	7.9

hand side vector $\mathbb{b}$ to be the form $\mathbb{b} = (1 + \imath)\mathbf{A1}$, with $\mathbf{1}$ being the vector of all entries equal to 1. Here $\mathbf{T}$ and $\mathbf{W}$ correspond to the five-point centered difference matrices approximating the negative Laplacian operator with homogeneous Dirichlet boundary conditions and periodic boundary conditions, respectively, on a uniform mesh in the unit square $[0, 1] \times [0, 1]$ with the mesh-size $h = 1/(m+1)$. Although this problem is an artificially constructed one, it is quite challenging for iterative solvers and therefore we include it in our tests.

Numerical results for Example 4.2 are listed in Tables 3 and 4. Table 3 shows the numbers of iteration steps (IT), the computing times (CPU) and the computing residual error (RES) with respect to the PS method, the PMHSS method, and GMRES and GMRES(20) methods. In Table 4 we show the results obtained with GMRES and GMRES(10) preconditioned with the PS- and PMHSS-preconditionees, respectively.

From Table 3 we see that the PS iteration method performs much better than GMRES and GMRES(20) both in terms of iteration counts and in terms of CPU times. Compared with PMHSS iteration method, the PS method has an advantage in CPU times.

From Table 4 we see that the iteration counts with the PMHSS and PS-Preconditioned GMRES(20) methods almost remain constantly. In addition, we observe that when $\mathbf{M}(\alpha)$ used as a preconditioner, PS performs better than PMHSS in iteration steps and CPU times.

Table 3 IT, CPU and ERR(1e-7) for PMHSS, PS, GMRES and GMRES(20) methods for Example 4.2

Method	m	$m = 16$	$m = 24$	$m = 32$	$m = 48$	$m = 64$
PS	$\alpha_{\exp}$	7.0	7.0	7.0	7.0	6.5
	IT	52	54	60	70	78
	CPU	0.0074	0.0558	0.1945	1.1251	2.0926
	ERR	8.6	9.9	8.5	9.9	9.2
PMHSS	$\alpha_{\exp}$	0.61	0.66	0.42	0.49	0.57
	IT	33	33	34	34	34
	CPU	0.0291	0.1411	0.3336	1.1517	2.6327
	ERR	7.6	9.6	7.7	8.0	8.1
GMRES	IT	18	34	43	76	90
	CPU	0.0454	0.0892	0.2047	1.2103	2.7746
	ERR	9.5	9.5	9.9	9.9	9.9
GMRES(20)	IT	61	94	126	189	253
	CPU	0.0435	0.1235	0.3046	1.3168	2.7341
	ERR	7.6	8.2	9.5	9.3	9.9

Table 4 IT, CPU and ERR(1e-7) for preconditioned GMRES and GMRES(10) for Example 4.2

Method	Preconditioner	m	$m = 16$	$m = 24$	$m = 32$	$m = 48$	$m = 64$
GMRES	PS	$\alpha_{\exp}$	0.605	0.605	0.605	0.605	0.605
		IT	3	3	3	3	4
		CPU	0.0056	0.0677	0.1645	0.9947	2.5324
		ERR	1.3	3.3	6.0	4.1	5.8
GMRES	PMHSS	$\alpha_{\exp}$	4.37	5.24	7.06	4.85	2.71
		IT	5	6	6	6	7
		CPU	0.0134	0.0852	0.2437	1.1045	2.6355
		ERR	7.8	9.2	8.7	9.0	6.7
GMRES(10)	PS	$\alpha_{\exp}$	0.564	0.564	0.564	0.564	0.564
		IT	4	5	5	5	5
		CPU	0.0056	0.0863	0.1682	1.0024	2.0847
		ERR	5.1	3.3	9.1	3.3	4.5
GMRES(10)	PMHSS	$\alpha_{\exp}$	6.62	4.27	6.82	5.91	8.67
		IT	5	5	5	5	7
		CPU	0.0133	0.0844	0.2511	1.0217	2.6523
		ERR	7.1	6.7	7.4	6.4	7.9

5 Conclusions

In this paper, we have established and analyzed a new parameterized splitting iteration method for solving an important class of complex symmetric linear systems based on a parameterized splitting of the coefficient matrix. Numerical experiments have shown

that the new splitting iteration method served either as a preconditioner or as a solver may yield satisfactory results when applied to the complex symmetric linear systems of practical interest.

References

1. Arridge, SR: Optical tomography in medical imaging. Inverse Probl. **15**, 41–93 (1999)
2. Axelsson, O, Kucherov, A: Real valued iteration methods for solving complex symmetric linear syetems. Numer. Linear Algebra Appl. **7**, 197–218 (2000)
3. Bai, Z-Z: Structured preconditioners for nonsingular matrices of block two-by-two structures. Math. Comput **75**, 791–815 (2006)
4. Bai, Z-Z, Benzi, M, Chen, F: Modified HSS iteration methods for complex symmetric linear systems. Computing **87**, 93–111 (2010)
5. Bai, Z-Z, Benzi, M, Chen, F: On preconditioned MHSS iteration methods for complex symmetric linear systems. Numer. Algor. **56**, 297–317 (2011)
6. Bai, Z-Z, Golub, GH, Ng, MK: Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems. SIAM J. Matrix Anal. Appl. **24**, 603–626 (2003)
7. Bai, Z-Z., Benzi, M., Chen, F., Wang, Z.-Q.: Preconditioned MHSS iteration methods for a class of block two-by-two linear systems with application to distributed control problems. IMA J. Numer. Anal. 33(1):343–369 (2013)
8. Benzi, M, Golub, GH, Liesen, J: Numerical solution of saddle point problems. Acta Numer. **14**, 1–137 (2005)
9. Benzi, M, Bertaccini, D: Block preconditioning of real-valued iterative algorithms for complex linear systems. IMA J. Numer. Anal. **28**, 598–618 (2008)
10. Bertaccini, D: Efficient solvers for sequences of complex symmetric linear systems. Electr. Trans. Numer. Anal. **18**, 49–64 (2004)
11. Betts, J.T.: Practical Methods For Optimal Control Using Nonlinear Programming. SIAM, Philadelphia, (2001)
12. Feriani, A, Perotti, F, Simoncini, V: Iterative system solvers for the frequency analysis of linear mechanical systems. Comput. Methods Appl. Mech. Eng. **190**, 1719–1739 (2000)
13. Frommer, A., Lippert, T., Medeke, B., Schilling, K.: Numerical challenges in lattice quantum chromodynamics, Lecture notes in computational science and engineering. Springer, Heidelberg, **15** (2000)
14. Poirier, B: Efficient preconditioning scheme for block partitioned matrices with structured sparsity. Numer. Linear Algebra Appl. **7**, 715–726 (2000)
15. van Dijk, W., Toyama, F.M.: Accurate numerical solutions of the time-dependent Schrodinger equation. Phys. Rev. E, **75**, 036707 (2007)
16. Young, D.M.: Iterative Solution of Large Linear Systems. Academic Press, New York, (1971)
17. Zhou, Y-Y, Zhang, G-F: A generalization of parameterized inexact Uzawa method for generalized saddle point problems. Appl. Math. Comput. **215**, 599–607 (2009)