Numer Algor (2016) 73:927–955
DOI 10.1007/s11075-016-0123-x

ORIGINAL PAPER

# A new iterative method for solving a class of complex symmetric system of linear equations

Davod Hezari[1] · Davod Khojasteh Salkuyeh[1] ·
Vahid Edalatpour[1]

Received: 25 April 2015 / Accepted: 4 March 2016 / Published online: 22 March 2016

**Abstract** We present a new stationary iterative method, called Scale-Splitting (SCSP) method, and investigate its convergence properties. The SCSP method naturally results in a simple matrix splitting preconditioner, called SCSP-preconditioner, for the original linear system. Some numerical comparisons are presented between the SCSP-preconditioner and several available block preconditioners, such as PGSOR (Hezari et al. Numer. Linear Algebra Appl. **22**, 761–776, 2015) and rotate block triangular preconditioners (Bai Sci. China Math. **56**, 2523–2538, 2013), when they are applied to expedite the convergence rate of Krylov subspace iteration methods for solving the original complex system and its block real formulation, respectively. Numerical experiments show that the SCSP-preconditioner can compete with PGSOR-preconditioner and even more effective than the rotate block triangular preconditioners.

**Keywords** Complex symmetric linear systems · Symmetric positive definite · Preconditioner

**Mathematics Subject Classification (2010)** 65F10 · 65F15

✉ Davod Hezari
davodhezari@phd.guilan.ac.ir; d.hezari@gmail.com

Davod Khojasteh Salkuyeh
khojasteh@guilan.ac.ir; salkuyeh@gmail.com

Vahid Edalatpour
vedalat.math@gmail.com

[1] Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran

## 1 Introduction

We consider the nonsingular system of linear equations of the form

$$Az = b, \quad \text{with} \quad A = W + iT, \quad z = x + iy \quad \text{and} \quad b = f + ig, \tag{1}$$

where $i = \sqrt{-1}$ is the imaginary unit, the $n \times n$ matrices $W$ and $T$ are real and the vectors $x$, $y$, $f$ and $g$ are all in $\mathbb{R}^n$. Note that $A$ is nonsingular if and only if $null(W) \cap null(T) = \{0\}$ and $i$ is not a generalized eigenvalue of the matrix pair $(W, T)$ (i.e., $Tx \neq iWx$ for any $x \neq 0$), where $null(\cdot)$ denotes the null space of the corresponding matrix.

Such systems appear in a variety of scientific computing and engineering applications such as computational electrodynamic [1, 2], FFT-based solution of certain time-dependent PDEs [3], structural dynamics [4], diffuse optical tomography [5], quantum mechanics [6] and molecular scattering [7]. For more applications of this class of complex symmetric systems see [8] and references therein.

Bai et al. in [9] described the Hermitian and skew-Hermitian splitting (HSS) method to solve non-Hermitian positive definite system of linear equations. By modifying and preconditioning the HSS iteration method, Bai et al. [10, 11] presented a preconditioned and modified HSS (PMHSS) iteration method for solving the complex linear system (1). Moreover, Lopsided version of the PMHSS method has been presented by Li et al. [12]. Additionly, there are also other solution techniques for solving complex symmetric linear system (1) which are extensions of Krylov subspace methods. For example, the well-known conjugate orthogonal conjugate gradient (COCG) [13], quasi-minimal residual (QMR) method [14], conjugate $A$-orthogonal conjugate residual (COCR) [15], symmetric complex bi-conjugate gradient (SCBiCG) method [16] and Quasi-minimal residual variants of the COCG and COCR methods [17].

To avoid the complex arithmetic and constructing efficient preconditioners, one can deal with the $2n \times 2n$ equivalent real formulation

$$\mathcal{R}u \equiv \begin{pmatrix} W & -T \\ T & W \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix} \equiv d; \tag{2}$$

see [8, 18, 19]. In [18], the foregoing PMHSS method is accommodated to solve the block two-by-two linear system (2) using real arithmetics. The PMHSS methods naturally lead to matrix splitting preconditioners, called PMHSS preconditioners, for coefficient matrices of (1) and (2) which have recently gained more attention; see [20, 21]. When the matrices $W$ and $T$ are symmetric positive semi-definite satisfying $null(W) \cap null(T) = \{0\}$, the convergence analysis of these PMHSS methods and the spectral properties of the preconditioned matrix have been presented for linear systems (1) and (2); see [11, 18]. Based on the PMHSS preconditioning matrix, Bai in [22] proposed three rotated block triangular preconditioning matrices for block two-by-two linear system (2) as follows

$$
\begin{aligned}
L(\omega) &= \frac{1}{\sqrt{\omega^2+1}} G \begin{pmatrix} \omega W + T & 0 \\ \omega T - W & \omega W + T \end{pmatrix}, \\
U(\omega) &= \frac{1}{\sqrt{\omega^2+1}} G \begin{pmatrix} \omega W + T & \omega W - T \\ 0 & \omega W + T \end{pmatrix},
\end{aligned}
\tag{3}
$$

and

$$
\begin{aligned}
P(\omega) = & \frac{1}{\sqrt{\omega^2+1}} G \begin{pmatrix} \omega W + T & 0 \\ \omega T - W & \omega W + T \end{pmatrix} \begin{pmatrix} \omega W + T & 0 \\ 0 & \omega W + T \end{pmatrix}^{-1} \\
& \times \begin{pmatrix} \omega W + T & \omega W - T \\ 0 & \omega W + T \end{pmatrix},
\end{aligned}
$$

where $G = \frac{1}{2} \begin{pmatrix} I & -I \\ I & I \end{pmatrix}$ is a block Givens rotation, $I$ is the identity matrix and $\omega$ is a real constant. The matrices $L(\omega)$, $U(\omega)$ and $P(\omega)$ are called the rotated block lower triangular (RBLT), rotated block upper triangular (RBUT) and rotated block triangular product (RBTP) preconditioners, respectively. The rotated block triangular preconditioners can be applied not only to the case that the matrices $W$ and $T$ are symmetric positive semi-definite but also to case that either $W$ or $T$ is nonsymmetric. At each step of the rotated block triangular preconditioning processes, it is required to invert the matrix $\omega W + T$ exactly. To avoid this problem, an inexact variant of the rotated block triangular preconditioners have been proposed in [23], which use a nonsingular approximation of $\omega W + T$, for example, incomplete Cholesky factorization or incomplete LU factorization when the matrix $\omega W + T$ is symmetric positive definite or nonsymmetric, respectively.

Based on the successive overrelaxation (SOR) method, recently Salkuyeh et al. [19] applied the generalized SOR (GSOR) to efficiently solve the block two-by-two linear system (2). After that, they proposed a preconditioned variant of the GSOR (PGSOR) method [24] and established conditions such that the PGSOR iteration method to be more effective than the GSOR iteration method.

In this paper, a new stationary matrix splitting iteration method, called Scale-Splitting (SCSP), is presented to solve the original complex system (1) and convergence theory together with the spectral properties of the corresponding preconditioned matrix are established. The authors in [24] have established the convergence analysis of the PGSOR method when both of the matrices $W$ and $T$ are symmetric positive semi-definite and at leat one of them is positive definite. Here, we weaken these conditions to the requirement that both of the matrices $W$ and $T$ are symmetric positive semi-definite satisfying $null(W) \cap null(T) = \{0\}$. Also, the SCSP and PGSOR iteration methods naturally leads to a preconditioning matrix for the original complex linear system (1) and the block two-by-two linear system (2), respectively. Applications of these two preconditioners involve solving linear subsystem with the coefficient matrix $\omega W + T$ that depends on the structures of matrices $W$ and $T$, and can be exactly solved by the Cholesky factorization or LU factorization. Moreover, similar to the inexact rotated block triangular preconditioners, an inexact variant of the SCSP and PGSOR preconditioners can be used by applying a nonsingular approximation of the matrix $\omega W + T$, for example, incomplete Cholesky factorization or incomplete LU factorization when the matrix $\omega W + T$ is symmetric positive definite or nonsymmetric, respectively. Numerical experiments show that the Krylov subspace methods such as GMRES [25] or its restarted variant GMRES($\ell$) [25] incorporated with the exact or inexact SCSP and PGSOR preconditioners lead to rapid convergence and tend to outperform the exact or inexact rotated block triangular preconditioners.

Throughout this paper, we use $(\cdot)^T$ and $\|\cdot\|$ to show the transpose and Euclidean norm of either a vector or a matrix, and denote by $\rho(\cdot)$ the spectral radius of the corresponding matrix.

The outline of this paper is as follows. In Section 2 we describe the SCSP iteration method for solving the original complex system (1) and investigate its convergence properties as well as extending the analysis to the preconditioned GMRES method. In Section 3 we present and analyze the PGSOR iteration method for solving the block two-by-two linear system (2) and discuss the spectral properties of the corresponding preconditioned matrix. Section 4 presents some numerical examples to show the effectiveness of the SCSP iteration method and to compare the effectiveness of the SCSP, PGSOR and rotate block triangular preconditioners to accelerate Krylov subspace iteration methods such as GMRES or its restarted variant GMRES($\ell$).

## 2 Scale-Splitting iteration method

In this section, we present the SCSP iteration method for solving the original complex system (1). Let $\omega$ be a real positive constant and the matrix $\omega W + T$ be nonsingular. By multiplying the complex number $(\omega - i)$ through both sides of the complex system (1) we obtain the following equivalent system

$$A_\omega z = (\omega - i)b, \tag{4}$$

where $A_\omega = (\omega W + T) + i(\omega T - W)$ in which $i = \sqrt{-1}$. By rewriting it as the system of fixed-point equation

$$(\omega W + T)z = i(W - \omega T)z + (\omega - i)b,$$

we obtain the SCSP iteration method, for solving equivalent complex system (4), as follows.

SCSP iteration method

*Given an initial guess $z^{(0)} \in \mathbb{C}^n$ and positive constant $\omega$, for $k = 0, 1, 2 \ldots$, until $\{z^{(k)}\}$ converges, solve*

$$(\omega W + T)z^{(k+1)} = i(W - \omega T)z^{(k)} + (\omega - i)b. \tag{5}$$

*for $z^{(k+1)}$.*

The SCSP iteration method can be equivalently rewritten as

$$z^{(k+1)} = G_\omega z^{(k)} + c_\omega, \quad k = 0, 1, 2, \dots \tag{6}$$

where

$$G_\omega = i(\omega W + T)^{-1}(W - \omega T) \quad \text{and} \quad c_\omega = (\omega - i)(\omega W + T)^{-1} b. \tag{7}$$

Here, $G_\omega$ is the iteration matrix of the SCSP method. In fact, (6) is also obtained from the splitting

$$A = M_\omega - N_\omega \tag{8}$$

of the coefficient matrix $A$, with

$$M_\omega = \frac{\omega + i}{\omega^2 + 1}(\omega W + T) \quad \text{and} \quad N_\omega = \frac{-1 + i\omega}{\omega^2 + 1}(W - \omega T).$$

Note that

$$G_\omega = M_\omega^{-1} N_\omega,$$

and $M_\omega$ can be served as a preconditioner to the system (1). Therefore, the preconditioned system takes the following form

$$M_\omega^{-1} A z = M_\omega^{-1} b. \tag{9}$$

At each iteration of SCSP method, it is required to solve a linear system with coefficient matrix $\omega W + T$. In particular, if $W$ and $T$ are symmetric positive semi-definite matrices satisfying $null(W) \cap null(T) = \{0\}$, then $\omega W + T$ is symmetric positive definite. Hence, the SCSP method is well defined and the linear system with coefficient matrix $\omega W + T$ can be exactly solved by the Cholesky factorization or inexactly by the CG algorithm.

In continuation, we discuss the spectral radius of the iteration matrix $G_\omega$ and, based on the results obtained, investigate the convergence properties of the method and determine the optimal value of the parameter $\omega$.

**Theorem 1** *Let $W, T \in \mathbb{R}^{n \times n}$ be symmetric positive semi-definite matrices satisfying $null(W) \cap null(T) = \{0\}$, $\omega$ be a positive constant, and $G_\omega$ be the iteration matrix (7) of the SCSP method. Then the following statements hold true:*

(i) *$\lambda$ is an eigenvalue of $G_\omega$ if*

$$\lambda = i\frac{1 - \omega\mu}{\omega + \mu},$$

*where $\mu$ is a generalized eigenvalue of the matrix pair $(W, T)$ and $i$ is imaginary unit;*

(ii) *the spectral radius $\rho(G_\omega)$ of the iteration matrix $G_\omega$ satisfies*

$$\rho(G_\omega) = \max\left\{\frac{1 - \omega\mu_{\min}}{\omega + \mu_{\min}}, \frac{\omega\mu_{\max} - 1}{\omega + \mu_{\max}}\right\}, \tag{10}$$

*furthermore, if*

$$\begin{cases} \dfrac{1-\mu_{\min}}{1+\mu_{\min}} < \omega < \dfrac{1+\mu_{\max}}{\mu_{\max}-1} & \textit{for } \mu_{\max} > 1 \\ \dfrac{1-\mu_{\min}}{1+\mu_{\min}} < \omega & \textit{for } \mu_{\max} \le 1, \end{cases}$$

*then $\rho(G_\omega) < 1$, where $\mu_{\min}$ and $\mu_{\max}$ are the smallest and largest generalized eigenvalues of the matrix pair $(W, T)$, respectively;*

(iii) *the optimal parameter $\omega$ which minimizes $\rho(G_\omega)$ is given by*

$$\omega^* = \frac{1-\mu_{\min}\mu_{\max}+\sqrt{(1+\mu_{\min}^2)(1+\mu_{\max}^2)}}{\mu_{\min}+\mu_{\max}}, \tag{11}$$

*and the corresponding optimal convergence factor of the method is given by*

$$\rho(G_{\omega^*}) = \frac{1-\omega^*\mu_{\min}}{\omega^*+\mu_{\min}} \left( = \frac{\omega^*\mu_{\max}-1}{\omega^*+\mu_{\max}} \right). \tag{12}$$

*Proof* To prove (i), let $0 \ne x \in \mathbb{C}^n$ be a eigenvector associated with a generalized eigenvalue $\mu$ of the matrix pair $(W, T)$. We have

$$Tx = \mu W x, \tag{13}$$

which implies $(\omega W + T)x = (\omega + \mu)Wx$. Since $\omega > 0$, and $W$ and $T$ are symmetric positive semi-definite matrices satisfying $null(W) \cap null(T) = \{0\}$, it is easy to see that $\mu \ge 0$ and $\omega W + T$ is symmetric positive definite. Thus, we can write

$$(\omega W + T)^{-1} W x = \frac{1}{\omega+\mu} x. \tag{14}$$

By applying (13) and (14), we have

$$\begin{aligned} G_\omega x &= i(\omega W + T)^{-1}(W - \omega T)x \\ &= i(1-\omega\mu)(\omega W + T)^{-1} W x \\ &= i\frac{1-\omega\mu}{\omega+\mu} x. \end{aligned}$$

We now turn to prove (ii). From (i) we can see that

$$\rho(G_\omega) = \max_{\mu\in\sigma(W,T)} \left| \frac{1-\omega\mu}{\omega+\mu} \right|, \tag{15}$$

where $\sigma(W, T)$ is the set of generalized eigenvalues of the matrix pair $(W, T)$. Invoking that $\mu \ge 0$ and noting that

$$\frac{1-\omega\mu}{\omega+\mu}$$

is a decreasing function with respect to $\mu$, it is easy to obtain (10). From (15), we find that $\rho(G_\omega) < 1$ if and only if

$$-1 < \frac{1-\omega\mu}{\omega+\mu} < 1. \tag{16}$$

The right inequality of (16) holds if and only if

$$\frac{1-\mu}{1+\mu} < \omega \quad \forall \mu \in \sigma(W,T). \tag{17}$$

Let us define the function $f(\mu)$ by

$$f(\mu) = \frac{1-\mu}{1+\mu}.$$

Then, the inequality (17) holds if and only if

$$\frac{1-\mu_{\min}}{1+\mu_{\min}} < \omega,$$

that follows from the fact that $f(\mu)$ is a decreasing function. The left inequality of (16) is equivalent to

$$\omega(\mu-1) < 1+\mu. \tag{18}$$

If $\mu_{\max} \le 1$, then the inequality (18) holds for every $\mu \in \sigma(W,T)$ and if $\mu_{\max} > 1$, it is easy to find that the inequality (18) holds if and only if

$$\omega < \frac{1+\mu_{\max}}{\mu_{\max}-1},$$

which completes the proof of the part (ii).

Finally, to prove (iii), let us define the functions $f_1(\omega)$ and $f_2(\omega)$ by

$$f_1(\omega) = \frac{1-\omega\mu_{\min}}{\omega+\mu_{\min}} \quad \text{and} \quad f_2(\omega) = \frac{\omega\mu_{\max}-1}{\omega+\mu_{\max}}.$$

From relation (10), and the fact that $f_1(\omega)$ and $f_2(\omega)$ are respectively decreasing and increasing finctions of $\omega$ (see Fig. 1), we deduce that $\rho(G_\omega)$ attains its minimum when $\omega$ satisfies

$$\frac{1-\omega\mu_{\min}}{\omega+\mu_{\min}} = \frac{\omega\mu_{\max}-1}{\omega+\mu_{\max}}.$$

Or, equivalently, at

$$\omega^* = \frac{1-\mu_{\min}\mu_{\max}+\sqrt{(1+\mu_{\min}^2)(1+\mu_{\max}^2)}}{\mu_{\min}+\mu_{\max}},$$

at which we have (12). □

When $W$ and $T$ are symmetric positive semi-definite matrices satisfying $null(W) \cap null(T) = \{0\}$, we are ready to describe the spectral properties of the preconditioned matrix $M_{\omega^*}^{-1}A$ where $\omega^*$ is the optimal parameter introduced in Theorem 1. From (8), we have

$$M_{\omega^*}^{-1}A = I - G_{\omega^*},$$

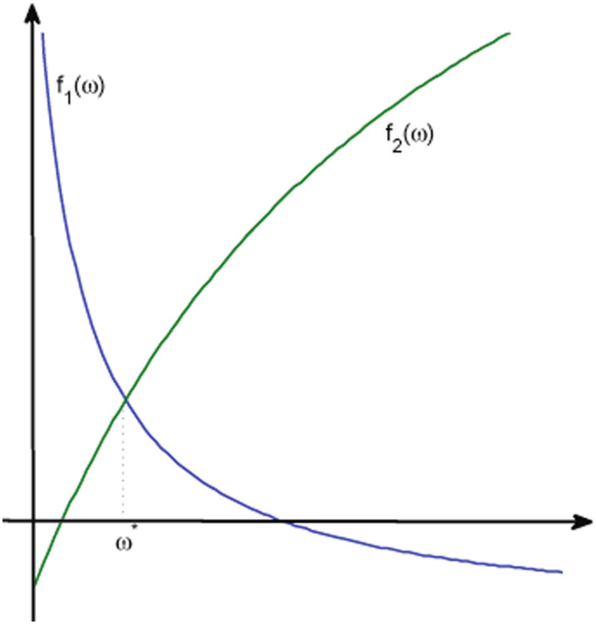

**Fig. 1** Graph of $f_1(\omega)$ and $f_2(\omega)$

where $G_{\omega^*} = M_{\omega^*}^{-1} N_{\omega^*}$ is the same as the iteration matrix of the SCSP method. Then, the eigenvalues of the preconditioned matrix $M_{\omega^*}^{-1} A$ are $1 - \lambda$ where $\lambda$ is an eigenvalue of the matrix $G_{\omega^*}$. From part (i) of Theorem 1, we see that the eigenvalues of $M_{\omega^*}^{-1} A$ are as follows

$$1 - i\frac{1 - \omega^* \mu}{\omega^* + \mu},$$

where $\mu$ is a generalized eigenvalue of the matrix pair $(W, T)$. Since $\mu \geq 0$, the eigenvalues of the preconditioned matrix $M_{\omega^*}^{-1} A$ are located on a straight line orthogonal to the real axis at the point $(1, 0)$, which their imaginary parts are smaller than 1. Therefore, it is expected that the preconditioned equation (9) would lead to considerable improvement over the unpreconditioned one.

*Remark 1* If the smallest generalized eigenvalue of the matrix pair $(W, T)$ is positive then, from part (ii) of Theorem 1, we can see that spectral radius of iteration matrix of the SCSP method, with $\omega = 1$, is smaller than 1 and this means that SCSP iteration method, with $\omega = 1$, is convergent. Therefore, similar to the above discussion, we can conclude that the eigenvalues of the corresponding preconditioned matrix are located on a straight line orthogonal to the real axis at the point $(1, 0)$, which their imaginary parts are smaller than 1.

## 3 PGSOR iteration method

The PGSOR iteration method was presented in [24] for solving the block two-by-two linear system (2) is summarized as follows.

The PGSOR iteration method

*Given an initial guess* $(x^{(0)^T}, y^{(0)^T})^T \in \mathbb{R}^{2n}$ *and positive constants* $\alpha$ *and* $\omega$*, for* $k = 0, 1, 2\ldots,$ *until* $\{(x^{(k)^T}, y^{(k)^T})^T\}$ *converges, compute*

$$\begin{cases} (\omega W + T)x^{(k+1)} = (1-\alpha)(\omega W + T)x^{(k)} + \alpha(\omega T - W)y^{(k)} + \alpha(\omega f + g), \\ (\omega W + T)y^{(k+1)} = -\alpha(\omega T - W)x^{(k+1)} + (1-\alpha)(\omega W + T)y^{(k)} + \alpha(\omega g - f). \end{cases} \tag{19}$$

In the matrix-vector form, the PGSOR iteration method can be equivalently rewritten as

$$\begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \mathcal{L}(\omega, \alpha) \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \alpha \mathcal{C}(\omega, \alpha) \begin{pmatrix} \omega f + g \\ \omega g - f \end{pmatrix}, \tag{20}$$

where

$$\mathcal{L}(\omega, \alpha) = \begin{pmatrix} \omega W + T & 0 \\ \alpha(\omega T - W) & \omega W + T \end{pmatrix}^{-1} \begin{pmatrix} (1-\alpha)(\omega W + T) & \alpha(\omega T - W) \\ 0 & (1-\alpha)(\omega W + T) \end{pmatrix}, \tag{21}$$

is iteration matrix, and

$$\mathcal{C}(\omega, \alpha) = \begin{pmatrix} \omega W + T & 0 \\ \alpha(\omega T - W) & \omega W + T \end{pmatrix}^{-1}.$$

The PGSOR method is well defined when matrix $\omega W + T$ is nonsingular. In particular, authors in [24] investigated the convergence conditions of the PGSOR iteration method when both of the matrices $W$ and $T$ are symmetric positive semidefinite, with at least one of them, e.g., $W$, being positive definite, and obtained the optimal values of the parameters $\alpha$ and $\omega$. Here, in the following theorem, we establish the convergence conditions of the PGSOR iteration method under weaker conditions that $W$ and $T$ are symmetric positive semidefinite matrices satisfying $null(W) \cap null(T) = \{0\}$.

**Theorem 2** *Let* $W, T \in \mathbb{R}^{n\times n}$ *be symmetric positive semidefinite satisfying* $null(W) \cap null(T) = \{0\}$*. Also, let* $\mu_{\min}$ *and* $\mu_{\max}$ *be the smallest and largest generalized eigenvalues of the matrix pair* $(W, T)$*, respectively. Then*

(i) *for every positive parameter* $\omega$*, the PGSOR iteration method is convergent if and only if*

$$0 < \alpha < \frac{2}{1 + \rho(\tilde{S}(\omega))}, \tag{22}$$

*where* $\tilde{S}(\omega) = (\omega W + T)^{-1}(\omega T - W)$ *and*

$$\rho(\tilde{S}(\omega)) = \max\left\{\frac{1-\omega\mu_{\min}}{\omega+\mu_{\min}}, \frac{\omega\mu_{\max}-1}{\omega+\mu_{\max}}\right\}. \tag{23}$$

*Moreover, the optimal value of relaxation parameter* $\alpha$ *and corresponding optimal convergence factor for the PGSOR iteration method are respectively given by*

$$\alpha^* = \frac{2}{1+\sqrt{1+\rho(\tilde{S}(\omega))^2}} \quad \textit{and} \quad \rho(\mathcal{L}(\omega,\alpha^*)) = 1-\alpha^*$$
$$= 1 - \frac{2}{1+\sqrt{1+\rho(\tilde{S}(\omega))^2}}; \tag{24}$$

(ii) *the optimal value of the parameter* $\omega$*, which minimizes the spectral radius* $\rho(\mathcal{L}(\omega,\alpha^*))$ *is given by*

$$\omega^* = \frac{1-\mu_{\min}\mu_{\max}+\sqrt{(1+\mu_{\min}^2)(1+\mu_{\max}^2)}}{\mu_{\min}+\mu_{\max}}, \tag{25}$$

*at which we have*

$$\alpha^* = \frac{2}{1+\sqrt{1+\xi^2}},$$

*where*

$$\xi = \frac{1-\omega^*\mu_{\min}}{\omega^*+\mu_{\min}} \left(= \frac{\omega^*\mu_{\max}-1}{\omega^*+\mu_{\max}}\right).$$

*Moreover, we have* $\rho(\mathcal{L}(\omega^*,\alpha^*)) < \frac{\sqrt{2}-1}{\sqrt{2}+1}$.

*Proof* The proof of relation (23) is rather similar to that of relation (10) in Theorem 1 and is omitted. Since the PGSOR iteration method is the same as the GSOR iteration method for solving block two-by-two linear system obtained from (4), then, from Theorem 1 and 2 in [19], we can find that (i) holds.

The proof of (ii) is quite similar to that of Theorem 2.4 and Corollary 2.2 in [24] and is omitted. □

*Remark 2* Note that the optimal value of parameter $\omega$ for the PGSOR iteration method is equal to that of the SCSP iteration method.

Considering matrices

$$\mathcal{P}(\omega,\alpha) = \frac{1}{\omega^2+1}\mathcal{Q}\begin{pmatrix} \omega W + T & 0 \\ \alpha(\omega T - W) & \omega W + T \end{pmatrix} \tag{26}$$

and

$$\mathcal{F}(\omega,\alpha) = \frac{1}{\omega^2+1}\mathcal{Q}\begin{pmatrix} (1-\alpha)(\omega W + T) & \alpha(\omega T - W) \\ 0 & (1-\alpha)(\omega W + T) \end{pmatrix}$$

with $\mathcal{Q} = \begin{pmatrix} \omega I & -I \\ I & \omega I \end{pmatrix}$, we see that $\mathcal{R} = \mathcal{P}(\omega, \alpha) - \mathcal{F}(\omega, \alpha)$ and $\mathcal{L}(\omega, \alpha) = \mathcal{P}^{-1}(\omega, \alpha)\mathcal{F}(\omega, \alpha)$. Therefore, the PGSOR method is a stationary iterative method induced by the previous matrix splitting. Hence, we deduce that the $\mathcal{P}(\omega, \alpha)$ can be used as a preconditioner for the block two-by-two system (2), which will be referred to as the PGSOR preconditioner. In the implementation of the preconditioner $\mathcal{P}(\omega, \alpha)$, we need to solve $\mathcal{P}(\omega, \alpha)v = r$ for the generalized residual vector $v = (v_a^T, v_b^T)^T$, where $r = (r_a^T, r_b^T)^T$ is the current residual vector. Due to the multiplicative structure of the preconditioner $\mathcal{P}(\omega, \alpha)$, computing of the generalized residual vector can be accomplished through the following procedure which involves solving two linear subsystems of the same coefficient matrix $\omega W + T$:

- Set $\tilde{r}_a = \omega f + g$ and $\tilde{r}_b = \omega g - f$;
- Solve $(\omega W + T)v_a = \tilde{r}_a$ for $v_a$;
- Solve $(\omega W + T)v_b = \alpha(W - \omega T)v_a + \tilde{r}_b$ for $v_b$.

Comparing with the rotated block triangular preconditioners in [22], we see that the action costs of the PGSOR preconditioner are essentially the same as those of RBLT and RBUT preconditioners and less than that of RBTP preconditioner.

*Remark 3* If, in the PGSOR preconditioner (26), $\omega = 1$ and $\alpha = 1$ then $\mathcal{P}(\omega, \alpha)$ is a scaled product of the RBLT preconditiner with $\omega = 1$.

*Remark 4* By writing the real form of the SCSP iteration method (5), it is easy to see that the resulting iteration method is the same as block Jacobi iteration method for solving the block two-by-two linear system (2) and the SCSP preconditioner $M_\omega$ have the following real form

$$\mathcal{H}_\omega = \begin{pmatrix} \omega W + T & 0 \\ 0 & \omega W + T \end{pmatrix},$$

which can be used for the block two-by-two linear system (2). It is noteworthy that the PGSOR iteration method and corresponding preconditioner have no complex counterpart.

In the following, we describe the spectral properties of the preconditioned matrix $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$ for some values of $\alpha$ and $\omega$.

*Remark 5* Let $\mathcal{P}(\omega, \alpha)$ be the PGSOR preconditioning matrix and $\mathcal{L}(\omega, \alpha)$ be the iteration matrix of the PGSOR method defined in (26) and (21), respectively. Then, according to Theorem 2, the following statements hold true.

(i) Since $\rho(\mathcal{L}(\omega^*, \alpha^*)) < \frac{\sqrt{2}-1}{\sqrt{2}+1} \approx 0.17$, the eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ are contained within the complex disk centered at 1 with radius $\frac{\sqrt{2}-1}{\sqrt{2}+1}$ due to $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R} = I - \mathcal{L}(\omega^*, \alpha^*)$.

(ii) If $\omega = 1$, then the PGSOR iteration method is convergent for every $0 < \alpha < 1$, since $\rho(\tilde{S}(1)) < 1$. Also, from part (i) of Theorem 2 we see that the optimal parameter $\tilde{\alpha}^*$ corresponding to $\omega = 1$ belongs to the interval $(\frac{2}{\sqrt{2}+1}, 1)$ and

$$\rho(\mathcal{L}(\omega^*, \alpha^*)) \leq \rho(\mathcal{L}(1, \tilde{\alpha}^*)) < \frac{\sqrt{2}-1}{\sqrt{2}+1}.$$

And similar to the case (i), the eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}(1, \tilde{\alpha}^*)\mathcal{R}$ are also contained within the complex disk centered at 1 with radius $\frac{\sqrt{2}-1}{\sqrt{2}+1}$.

Since it may turn out to be difficult to find the optimal values of the parameters $\omega$ and $\alpha$, Hezari et al. in [24] used the values $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$, and presented some reasons for using these values. Numerical experiments in [24] show that performance of the PGSOR iteration method with $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is close to those of the optimal parameters. In the following theorem, we exactly describe the spectral radius of the iteration matrix $\mathcal{L}(\omega, \alpha)$ of the PGSOR iteration method and the spectral distribution of the preconditioned matrix $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$ with those values that is a strong reason why the corresponding PGSOR iteration method and the PGSOR preconditioner are of high-performance.

**Theorem 3** *Let $\mathcal{R}$ be the block two-by-two matrix defined in (2), with $W$ and $T$ being symmetric positive semidefinite matrices satisfying $null(W) \cap null(T) = \{0\}$, and let $\mathcal{L}(\omega, \alpha)$ be the iteration matrix of the PGSOR method and $\mathcal{P}(\omega, \alpha)$ be the PGSOR preconditioning matrix defined in (21) and (26), respectively. If $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ then*

$$\rho\left(\mathcal{L}\left(1, \frac{2}{\sqrt{2}+1}\right)\right) = \frac{\sqrt{2}-1}{\sqrt{2}+1} \tag{27}$$

*and the eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}\left(1, \frac{2}{\sqrt{2}+1}\right)\mathcal{R}$ are contained within the complex disk centered at 1 with radius $\frac{\sqrt{2}-1}{\sqrt{2}+1}$*

*Proof* If $\omega = 1$ it follows from part (i) of Theorem 2, that the corresponding optimal parameter $\alpha^*$ is given by

$$\alpha^* = \frac{2}{1 + \sqrt{1 + \rho(\tilde{S}(1))^2}},$$

where $\tilde{S}(1) = (W+T)^{-1}(T-W)$. From (23) and the fact the generalized eigenvalues of matrix pair $(W, T)$ is nonnegative, we have $\rho(\tilde{S}(1)) < 1$ and this implies that $\frac{2}{\sqrt{2}+1} < \alpha^*$. Note that the PGSOR iteration method with $\omega = 1$ is the same as the GSOR iteration method for solving block two-by-two linear system obtained from (4) and therefore, $\mathcal{L}(1, \alpha)$ and $\alpha^*$ are the same as the corresponding iteration matrix and the optimal relaxation parameter of the GSOR method, respectively. Following the proof of finding the optimal value of relaxation parameter for the GSOR method [19, Theorem 2], we find that for every $\alpha < \alpha^*$, it holds that $\rho(\mathcal{L}(1, \alpha)) = 1 - \alpha$.

Therefore, here for $\alpha = \frac{2}{\sqrt{2}+1} < \alpha^*$, we can conclude (27). The second part of the theorem is derived from the fact that $\mathcal{P}^{-1}\left(1, \frac{2}{\sqrt{2}+1}\right)\mathcal{R} = I - \mathcal{L}\left(1, \frac{2}{\sqrt{2}+1}\right)$. □

Now, we consider weaker conditions for the matrices $W$ and $T$, and similar to the [22, Theorem 3.1], we estimate region for the eigenvalues of the preconditioned matrix $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$.

**Theorem 4** *Let $\omega$ be a real constant and $\alpha \in (0, 1]$. Also, let $\mathcal{R}$ be the nonsingular block two-by-two matrix defined in (2), with $W$ and $T$ being real square matrices such that $\omega W + T$ is nonsingular. Then, the eigenvalues of $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$ are located within the complex disk centered at 1 with radius $\delta(\omega) = \|V(\omega)\|(\sqrt{1 + \alpha^2\|V(\omega)\|^2} + 1 - \alpha)$ where $V(\omega) = (\omega W + T)^{-1}(W - \omega T)$.*

*Proof* Let

$$Y(\omega) = \begin{pmatrix} 0 & V(\omega) \\ 0 & \alpha V^2(\omega) \end{pmatrix} \quad \text{and} \quad Z(\omega) = \begin{pmatrix} 0 & 0 \\ (\alpha - 1)V(\omega) & 0 \end{pmatrix}.$$

Then,

$$\begin{aligned} \mathcal{P}^{-1}(\omega, \alpha)\mathcal{R} &= \begin{pmatrix} \omega W + T & 0 \\ \alpha(\omega T - W) & \omega W + T \end{pmatrix}^{-1} \begin{pmatrix} \omega W + T & W - \omega T \\ \omega T - W & \omega W + T \end{pmatrix} \\ &= \begin{pmatrix} (\omega W + T)^{-1} & 0 \\ \alpha V(\omega)(\omega W + T)^{-1} & (\omega W + T)^{-1} \end{pmatrix} \begin{pmatrix} \omega W + T & W - \omega T \\ \omega T - W & \omega W + T \end{pmatrix} \\ &= \begin{pmatrix} I & V(\omega) \\ (\alpha - 1)V(\omega) & I + \alpha V^2(\omega) \end{pmatrix} \\ &= I + Y(\omega) + Z(\omega). \end{aligned}$$

From straightforward computations we can further obtain $\|Y(\omega) + Z(\omega)\| \leq \|Y(\omega)\| + \|Z(\omega)\| \leq \|V(\omega)\|(\sqrt{1 + \alpha^2\|V(\omega)\|^2} + 1 - \alpha)$. Now, by making use of the Lemma 3.2 in [26], we deduce that the eigenvalues of $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$ are located within the complex disk centered at 1 with radius $\delta(\omega)$. □

Now, when $\alpha = 1$, we can obtain deeper properties for the preconditioned matrix $\mathcal{P}^{-1}(1, \alpha)\mathcal{R}$.

**Theorem 5** *Let $\omega$ be a real constant and $\mathcal{R}$ be the nonsingular block two-by-two matrix defined in (2), with $W$ and $T$ being real square matrices such that $\omega W + T$ is nonsingular. Then, for the PGSOR preconditioning matrix $\mathcal{P}(\omega, 1)$, it holds that*

(i) *the eigenvalues of $\mathcal{P}^{-1}(\omega, 1)\mathcal{R}$ are 1 and $1 + \mu^2$, with $\mu$ being the eigenvalue of the matrix $V(\omega)$, so they are contained within the complex disk centered at 1 with radius $\|V(\omega)\|^2$;*

(ii) *let $\ell_0$ be the degree of the minimal polynomial of the matrix $V(\omega)$. Then the degree of the minimal polynomial of the matrix $\mathcal{P}^{-1}(\omega, 1)\mathcal{R}$ is, at most, $\ell_0 + 1$.*

*Proof* Following the proof of Theorem 4, we know that

$$\mathcal{P}^{-1}(\omega, 1)\mathcal{R} = \begin{pmatrix} I & V(\omega) \\ 0 & I + V^2(\omega) \end{pmatrix},$$

which immediately results in (i). Moreover, following the proof of the Proposition 2.1 in [27] and using the structure of the matrix $\mathcal{P}^{-1}(\omega, 1)\mathcal{R}$, we find that (ii) holds true. □

From Theorem 5 we find that when a Krylov subspace iteration method with an optimal or Galerkin property, like GMRES, is used to the preconditioned linear system $\mathcal{P}^{-1}(\omega, 1)\mathcal{R}$, it will converge to the exact solution in $\ell_0 + 1$ or fewer number of iteration steps, in exact arithmetic.

When the matrix $V(\omega)$ is diagonalizable, similar to the theorem 3.3 in [22], we can give the analytical expression for the eigenvalues and eigenvectors of the preconditioned matrix $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$.

**Theorem 6** *Let $\omega$ and $\alpha$ be real constants, and $\mathcal{R}$ be the nonsingular block two-by-two matrix defined in (2), with $W$ and $T$ being real square matrices such that $\omega W + T$ is nonsingular. Let $\Psi(\omega)$ be a nonsingular matrix and $\Upsilon(\omega)$ be a diagonal matrix such that $V(\omega)\Psi(\omega) = \Psi(\omega)\Upsilon(\omega)$, with $\psi_j := \psi_j(\omega)$ the j-th column of $\Psi(\omega)$ and $\upsilon_j := \upsilon_j(\omega)$ the j-th diagonal element of $\Upsilon(\omega)$. Denote by $s_j = sign(\upsilon_j)$ the sign function of $\upsilon_j$. Then, the eigenvalues $\lambda_j$, $j = 1, 2, \ldots, 2m$, of the matrix $\mathcal{P}^{-1}(\omega, \alpha)\mathcal{R}$ are $\lambda_j = \lambda_j^+$ and $\lambda_{m+j} = \lambda_j^-$, $j = 1, 2, \ldots, m$, where*

$$\lambda_j^{\pm} = \frac{1}{2}\left(2 + \alpha\upsilon_j^2 \pm \sqrt{\alpha^2\upsilon_j^4 + 4(1-\alpha)\upsilon_j^2}\right),$$

*and the corresponding eigenvectors are*

$$x_j = \begin{pmatrix} \frac{1}{\sqrt{1+\beta_j^2}}\psi_j \\ \frac{\beta_j}{\sqrt{1+\beta_j^2}}\psi_j \end{pmatrix}, \quad x_{m+j} = \begin{pmatrix} \frac{1}{\sqrt{1+\gamma_j^2}}\psi_j \\ -\frac{\gamma_j}{\sqrt{1+\gamma_j^2}}\psi_j \end{pmatrix}, \quad j = 1, 2, \ldots, m,$$

*where*

$$\begin{cases} \beta_j = \frac{1}{2}\left(\alpha\upsilon_j + s_j\sqrt{\alpha^2\upsilon_j^2 + 4(1-\alpha)}\right), \\ \gamma_j = \frac{1}{2}\left(-\alpha\upsilon_j + s_j\sqrt{\alpha^2\upsilon_j^2 + 4(1-\alpha)}\right). \end{cases}$$

*Proof* For convenience, let

$$\tilde{\Psi}(\omega) = \begin{pmatrix} \Psi(\omega) & 0 \\ 0 & \Psi(\omega) \end{pmatrix}, \quad K(\omega, \alpha) = \begin{pmatrix} I & \Upsilon(\omega) \\ (\alpha-1)\Upsilon(\omega) & I + \alpha\Upsilon^2(\omega) \end{pmatrix}.$$

From the proof of Theorem 4, we observe that

$$\begin{aligned}
\mathcal{P}^{-1}(\omega,\alpha)\mathcal{R} &= \begin{pmatrix} I & V(\omega) \\ (\alpha-1)V(\omega) & I+\alpha V^2(\omega) \end{pmatrix} \\
&= \begin{pmatrix} \Psi(\omega) & 0 \\ 0 & \Psi(\omega) \end{pmatrix} \begin{pmatrix} I & \Upsilon(\omega) \\ (\alpha-1)\Upsilon(\omega) & I+\alpha\Upsilon^2(\omega) \end{pmatrix} \begin{pmatrix} \Psi(\omega) & 0 \\ 0 & \Psi(\omega) \end{pmatrix}^{-1} \\
&= \tilde{\Psi}(\omega)K(\omega,\alpha)\tilde{\Psi}^{-1}(\omega).
\end{aligned}$$

Since the sub-blocks of $K(\omega,\alpha)$ are square diagonal matrices, then by following the proof of [22, Theorem 3.3], we can obtain the spectral decomposition of $K(\omega,\alpha)$, and after straightforward computations, the above eigenvalues and eigenvectors of the matrix $\mathcal{P}^{-1}(\omega,\alpha)\mathcal{R}$ are derived. □

## 4 Numerical experiments

In this section, we use three examples which are of the complex linear system of the form (1) with real matrices $W$ and $T$ that are symmetric positive definite. By using these examples, we illustrate the feasibility and effectiveness of the SCSP iteration method when it is employed as a solver to solve the complex symmetric system (1). We compare the performance of the SCSP iteration method with that of the PGSOR iteration method, which is applied for the equivalent block two-by-two linear system (2), from the point of view of both the number of iterations (denoted by IT) and the total computing times (in seconds, denoted by CPU). Beside these three complex linear systems, one [22] is given which is a nonsymmetric variant of the complex linear system (1). Using these examples, we examine the numerical behavior of the SCSP, PGSOR and RBLT preconditioning matrices and the corresponding preconditioned GMRES method, termed briefly as SCSP-GMRES, PGSOR-GMRES and RBLT-GMRES. For a comprehensive comparison we also solve the original complex valued system (1) via Matlab's sparse direct solver "\" and by un-preconditioned or ILU-preconditioned GMRES. Note that, since the numerical experiments in [22] show that the performance of the RBLT, RBUT and RBPT preconditioning matrices are almost the same, and the structure of RBLT preconditioning matrix is similar to that of the PGSOR preconditioning matrix, the comparison is done only with the RBLT preconditioning matrix. In the SCSP, PGSOR and RBLT preconditioning processes, it is required to solve linear subsystems with the coefficient matrix $\omega W + T$ and depending on the structure of the matrices $W$ and $T$ it can be done exactly by making use of the Cholesky or LU factorization.

Moreover, similar to the inexact RBLT preconditioning matrix (IRBLT) in [23], an inexact variant of the SCSP and PGSOR preconditioning matrices and the corresponding preconditioned restarted GMRES($\ell$) method, termed briefly as ISCSP-GMRES($\ell$), IPGSOR-GMRES($\ell$) and IRBLT-GMRES($\ell$), can be used by applying a nonsingular approximation of the matrix $\omega W + T$. Depending on structure of matrices $W$ and $T$, we use an incomplete Cholesky or LU factorization of $\omega W + T$ with dropping tolerance 0.1. The reported CPU times are the sum of the CPU time for the convergence of the method and the CPU time for computing the (complete or incomplete) Cholesky or LU factorization. It is necessary to mention that to solve symmetric positive definite and nonsymmetric system of linear equations we have used the sparse Cholesky and LU factorization incorporated with the symmetric and column approximate minimum degree reordering, respectively. To do so we have used the `symamd.m` and `colamd.m` commands of MATLAB. It should be noted that in all the tests, "IT" for the preconditioned or non-preconditioned GMRES($\ell$) stands for the number of restarts.

All tests are performed in MATLAB 8.0.0.783 (64-bit) on a laptop ASUS, Intel Core i7, 1.8 GHz with 6GB RAM. We use a null vector as an initial guess and the stopping criterion $\|b - Ax^{(k)}\| < 10^{-6}\|b\|$. For Example 1, 2 and 4, the optimal values of the parameters $\omega$ and $\alpha$ for the PGSOR method are the ones determined by the formula given in Theorem 2 and, according to Remark 2, the optimal value of the parameter $\omega$ for SCSP iteration method is the same as that of the PGSOR iteration method. Since Example 3 is a nonsymmetric variant of the complex linear system (1), the parameters $\omega$ and $\alpha$ adopted in exact preconditioners are the experimentally found optimal ones that minimize the total numbers of iteration steps of the GMRES iteration process and listed in Table 1 for different values of $m$ (denoted by $\alpha^*$ and $\omega^*$). These optimal parameters is also used in inexact preconditioners and corresponding restarted GMRES($\ell$) method. If the optimal parameters $\omega^*$ and $\alpha^*$ are not single points, we choose the $\omega^*$ as the one closest to 1.0 and the $\alpha^*$ as the smallest one. Moreover, we use an incomplete LU factorization of $\omega W + T$ with dropping tolerance 0.1 when applying ISCSP-GMRES($\ell$), IPGSOR-GMRES($\ell$) and IRBLT-GMRES($\ell$) for this example.

*Example 1* [24] Consider the linear system of equations (1) of the form

$$\left[\left(K + \frac{3-\sqrt{3}}{\tau} I\right) + i\left(K + \frac{3+\sqrt{3}}{\tau} I\right)\right] x = b, \tag{28}$$

where $\tau$ is the time step-size and $K$ is the five-point centered difference matrix approximating the negative Laplacian operator $L = -\Delta$ with homogeneous Dirichlet boundary conditions, on a uniform mesh in the unit square $[0, 1] \times [0, 1]$ with the mesh-size $h = 1/(m + 1)$. The matrix $K \in \mathbb{R}^{n\times n}$ possesses the tensor-product form

**Table 1** The experimental optimal parameters for preconditioned GMRES for Example 3 by minimizing iteration steps

| Method | | $m^2$ | | | | |
|---|---|---|---|---|---|---|
| | | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | $\omega^*$ | [0.10, 0.90] | [0.10, 1.20] | [0.50, 1.04] | [0.70, 0.99] | [0.70, 1.10] |
| PGSOR-GMRES | $\omega^*$ | [0.2, 0.48] | [0.4, 0.80] | [0.74, 0.91] | [0.70, 1.10] | [0.85, 1.10] |
| | $\alpha^*$ | [0.92, 1.02] | [0.90, 1.03] | [0.90, 1.02] | [0.80, 1.05] | [0.80, 1.05] |
| RBLT-GMRES | $\omega^*$ | [0.82, 1.04] | [0.83, 1.06] | [0.95, 1.03] | [0.97, 1.03] | [0.98, 1.02] |

$K = I \otimes V_m + V_m \otimes I$, with $V_m = h^{-2}\text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m\times m}$. Hence, $K$ is an $n \times n$ block-tridiagonal matrix, with $n = m^2$. We take

$$W = K + \frac{3-\sqrt{3}}{\tau} I \quad \text{and} \quad T = K + \frac{3+\sqrt{3}}{\tau} I,$$

and the right-hand side vector $b$ with its $j$th entry $b_j$ being given by

$$b_j = \frac{(1-i)j}{\tau(j+1)^2}, \quad j = 1, 2, \ldots, n.$$

In our tests, we take $\tau = h$. Furthermore, we normalize the coefficient matrix and the right-hand side by multiplying both by $h^2$.

*Example 2* [24] Consider the linear system of equations (1) as following

$$\left[(-\theta^2 M + K) + i(\theta C_V + C_H)\right] x = b,$$

where $M$ and $K$ are the inertia and the stiffness matrices, $C_V$ and $C_H$ are the viscous and the hysteretic damping matrices, respectively, and $\theta$ is the driving circular frequency. We take $C_H = \mu K$ with $\mu$ a damping coefficient, $M = I$, $C_V = 10I$, and $K$ 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 $h = 1/(m + 1)$. The matrix $K \in \mathbb{R}^{n\times n}$ possesses the tensor-product form $K = I \otimes V_m + V_m \otimes I$, with $V_m = h^{-2}\text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m\times m}$. Hence, $K$ is an $n \times n$ block-tridiagonal matrix, with $n = m^2$. In addition, we set $\theta = \pi$, $\mu = 0.02$, and the right-hand side vector $b$ to be $b = (1 + i)A\mathbf{1}$, with $\mathbf{1}$ being the vector of all entries equal to 1. As before, we normalize the system by multiplying both sides through by $h^2$.

The number of iteration steps and the CPU times for the PGSOR and SCSP methods for the examples 1, 2 and 4 with respect to different values of the problem size $m$, are listed in Tables 2, 3, 4 and 5, where the CPU times are shown in parentheses. Tables 2 and 4 show ITs and CPUs when the optimal parameters $\omega$ and $\alpha$ are adopted, while Tables 3 and 5 show ITs and CPUs when the iteration parameters $\omega$ and $\alpha$ are set to be 1 and $\frac{2}{\sqrt{2}+1}$, respectively. Note that the SCSP method is employed

**Table 2** IT and CPU for SCSP and PGSOR methods for Examples 1 and 2 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Example | Method | $m^2$ | | | | |
|---|---|---|---|---|---|---|
| | | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| No.1 | SCSP | 10 (0.02) | 10 (0.094) | 11 (0.393) | 11 (1.917) | 11 (9.822) |
| | PGSOR | 5 (0.024) | 5 (0.075) | 5 (0.310) | 5 (1.624) | 5 (8.286) |
| No.2 | SCSP | 42 (0.045) | 42 (0.190) | 43 (0.875) | 43 (4.675) | 43 (21.423) |
| | PGSOR | 8 (0.025) | 8 (0.086) | 8 (0.380) | 8 (1.884) | 8 (8.913) |

**Table 3** IT and CPU for SCSP and PGSOR methods for Examples 1 and 2 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Example | Method | $m^2$ | | | | |
|---|---|---|---|---|---|---|
| | | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| No.1 | SCSP | 18 (0.031) | 18 (0.114) | 18 (0.470) | 17 (2.392) | 16 (11.349) |
| | PGSOR | 8 (0.027) | 8 (0.093) | 8 (0.380) | 8 (1.863) | 9 (9.190) |
| No.2 | SCSP | 300 (0.212) | 329 (1.095) | 340 (5.402) | 344 (29.721) | 345 (132) |
| | PGSOR | 9 (0.027) | 9 (0.093) | 9 (0.423) | 9 (1.997) | 9 (9.288) |

for solving the original complex system (1), while PGSOR method is employed for solving equivalent block two-by-two system (2). At each step of the PGSOR method, it is required to solve two subsystems with the coefficient matrix $\omega W + T$, while in SCSP method one subsystem with the same coefficient matrix is required. In Table 2 we can see that the number of iterations of the SCSP method for Example 1 are almost twice as much as that of the PGSOR method, but the their computing times are close together. As seen, the PGSOR iteration method outperforms the SCSP iteration method.

In addition, when the results of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ are compared with those of the optimal values, for each $m$, we see that the ITs and CPUs of the PGSOR and SCSP methods in Tables 3 and 5 are larger than those in Tables 2 and 4, respectively. However, for Example 1, it can be mentioned that the performance of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is acceptable and close to those of the optimal values and also, for Examples 2 and 4, the performance of the PGSOR method with $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is acceptable and close to those of the optimal values, while performance of the SCSP method with $\omega = 1$ is far away those of the optimal values.

Tables 6, 7 and 8 present results for Example 1 where the CPU times are shown in parentheses. Table 7 shows ITs and CPUs when the optimal parameters $\omega$ and $\alpha$ are adopted, while Table 8 shows ITs and CPUs when the iteration parameters $\omega$ and $\alpha$ are set to be 1 and $\frac{2}{\sqrt{2}+1}$, respectively. The symbol † indicates that the computing time has lasted more than one hour. As seen, the direct method is slower than PGSOR-GMRES (except for $m = 64$) and also slower than SCSP-GMRES ( except for $m = 64, 128$). Both SCSP-GMRES and PGSOR-GMRES are very robust in all cases and exhibit mesh-independent convergence.

**Table 4** IT and CPU for SCSP and PGSOR methods for Example 4 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Method | $m^3$ | | | | |
|---|---|---|---|---|---|
| | $30^3$ | $35^3$ | $40^3$ | $45^3$ | $50^3$ |
| SCSP | 41 (1.557) | 54 (3.898) | 69 (8.536) | 86 (20.031) | 104 (36.293) |
| PGSOR | 8 (0.806) | 8 (1.737) | 8 (3.569) | 8 (7.455) | 9 (13.277) |

**Table 5** IT and CPU for SCSP and PGSOR methods for Example 4 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Method | $m^3$ | | | | |
|---|---|---|---|---|---|
| | $30^3$ | $35^3$ | $40^3$ | $45^3$ | $50^3$ |
| SCSP | 390 (10.424) | 389 (19.871) | 379 (35.794) | 375 (68.478) | 371 (103) |
| PGSOR | 9 (0.841) | 9 (1.799) | 9 (3.608) | 9 (7.641) | 9 (13.331) |

From Table 7, we see that among exact preconditioned GMRES methods and among inexact preconditioned GMRES($\ell$) methods, the PGSOR-preconditioned GMRES and ISCSP-preconditioned GMRES($\ell$) need the least iteration step and CPU time, respectively.

In addition, when the results of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ are compared with those of the optimal values, for each $m$, we see that the ITs of the SCSP-GMRES method in Table 8 are the same as those in Table 7 and the CPUs are smaller than those in Table 7. The ITs and CPUs of the RBLT-GMRES and IRBLT-GMRES(20) in Table 8 are exactly the same as those in Table 7. Moreover, the ITs and CPUs of the PGSOR-GMRES in Table 8 are larger than those in Table 7, and also the ITs of the IPGSOR-GMRES(20) and ISCSP-GMRES(20) in Table 8 are exactly the same as those in Table 7 (except for IPGSOR-GMRES(20) when $m = 1024$,). Thus, it can be said that the performance of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is acceptable and close to those of the optimal values.

Tables 9, 10 and 11 present results for Example 2 where the CPU times are shown in parentheses. Table 10 shows ITs and CPUs when the optimal parameters $\omega$ and $\alpha$ are adopted, while Table 11 shows ITs and CPUs when the iteration parameters $\omega$ and $\alpha$ are set to be 1 and $\frac{2}{\sqrt{2}+1}$, respectively. The symbol † indicates that the computing time has lasted more than one hour. As seen, the direct method is slower than SCSP-GMRES (except for $m = 64$) and slower than PGSOR-GMRES for $m = 512, 1024$. Both SCSP-GMRES and PGSOR-GMRES are very robust in all cases and exhibit mesh-independent convergence.

**Table 6** IT and CPU for GMRES, GMRES(20), ILU-GMRES, ILU-GMRES(20) and $A\backslash b$ for Example 1

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| $A\backslash b$ | (0.024) | (0.130) | (0.630) | (3.475) | (20.716) |
| GMRES | 81 (0.404) | 112 (2.449) | 155 (16.921) | 211 (171) | † |
| GMRES(20) | 5 (0.095) | 8 (2.474) | 11 (2.422) | 21 (27.472) | 34 (180) |
| ILU-GMRES | 25 (0.058) | 35 (0.375) | 48 (2.363) | 65 (20.622) | 87 (134) |
| ILU-GMRES(20) | 2 (0.053) | 2 (0.171) | 3 (0.980) | 4 (7.425) | 5 (35.607) |

**Table 7** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 1 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 8 (0.042) | 8 (0.139 ) | 8 (0.618) | 8 (2.828) | 8 (13.378) |
| PGSOR-GMRES | 4 (0.037) | 4 (0.116) | 4 (0.480) | 4 (2.541) | 5 (11.961) |
| RBLT-GMRES | 6 (0.043) | 6 (0.140) | 6 (0.599 ) | 6 (2.929) | 6 (13.127) |
| ISCSP-GMRES(20) | 2 (0.052) | 2 (0.162) | 3 (0.915) | 4 (7.226) | 5 (35.458) |
| IPGSOR-GMRES(20) | 2 (0.062) | 3 (0.275) | 4 (1.667) | 5 (9.910) | 7 (54.964) |
| IRBLT-GMRES(20) | 2 (0.060) | 3 (0.263) | 4 (1.613) | 5 (9.903) | 8 (61.820) |

From Table 10, we see that among exact preconditioned GMRES methods, the performance of SCSP- and PGSOR-GMRES is better than RBLT-GMRES from point of view of both the iteration number and the CPU time. The ITs of the SCSP-GMRES method show one more iteration step than those of the PGSOR-GMRES method except for $m = 64$, in which case the IT of the SCSP-GMRES method is exactly the same as that of the PGSOR-GMRES method. It is important to note that, among exact preconditioned GMRES methods, the computing time of the SCSP-GMRES is the least. Among inexact preconditioned GMRES($\ell$) methods, we see that the ITs and CPUs of ISCSP-GMRES(20) are the least.

In addition, when the results of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ are compared with those of the optimal values, for each $m$, we see that the ITs of the SCSP-, and RBLT-GMRES and SCSP-, and IRBLT-GMRES(20) method in Table 11 are exactly the same as those in Table 10. The ITs and CPUs of the IPGSOR-GMRES(20) in Table 11 are larger than those in Table 10 except for $m = 64$. It is important to note that among the exact the preconditioned GMRES and the inexact preconditioned GMRES($\ell$) methods in Table 11, ITs and CPUs of the SCSP-GMRES and ISCSP-GMRES(20) are the least, respectively.

**Table 8** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 1 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 8 (0.042) | 8 (0.133) | 8 (0.586) | 8 (2.828) | 8 (13.298) |
| PGSOR-GMRES | 7 (0.044) | 7 (0.151) | 7 (0.676) | 7 (3.265) | 7 (14.268) |
| RBLT-GMRES | 6 (0.043) | 6 (0.140) | 6 (0.599 ) | 6 (2.929) | 6 (13.127) |
| ISCSP-GMRES(20) | 2 (0.052) | 2 (0.162) | 3 (0.910) | 4 (7.481) | 5 (35.263) |
| IPGSOR-GMRES(20) | 2 (0.055) | 3 (0.270) | 4 (1.623) | 5 (9.912) | 8 (61.032) |
| IRBLT-GMRES(20) | 2 (0.060) | 3 (0.263) | 4 (1.613) | 5 (9.903) | 8 (61.820) |

**Table 9** IT and CPU for GMRES, GMRES(20), ILU-GMRES, ILU-GMRES(20) and $A\backslash b$ for Example 2

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| $A\backslash b$ | (0.024) | (0.131) | (0.611) | (3.366) | (19.893) |
| GMRES | 102 (0.622) | 196 (6.060) | 379 (96.904) | >500 | † |
| GMRES(20) | 15 (0.281) | 51 (2.967) | 167 (36.359) | >500 | † |
| ILU-GMRES | 38 (0.110) | 68 (1.092) | 115 (10.961) | 219 (192) | † |
| ILU-GMRES(20) | 3 (0.078) | 7 (0.586) | 23 (7.202) | 70 (126) | 228 (1603) |

*Example 3* [22] (Nonsymmetric BBC problem) Consider the linear system of equations $(W + iT)x = b$, with

$$W = I \otimes L + L \otimes I \quad \text{and} \quad T = 10(I \otimes L_c + L_c \otimes I) + 9(e_1 e_m^T + e_m e_1^T) \otimes I,$$

where $L = \text{tridiag}(-1-\theta, 2, -1+\theta) \in \mathbb{R}^{m\times m}$, $\theta = \frac{1}{2(m+1)}$, $L_c = L - e_1 e_m^T - e_m e_1^T \in \mathbb{R}^{m\times m}$ and $e_1$ and $e_m$ are the first and last unit vectors in $\mathbb{R}^m$, respectively. We take the right-hand side vector $b$ to be $b = (1+i)A\mathbf{1}$, with $\mathbf{1}$ being the vector of all entries equal to 1.

*Example 4* [28] Consider the 3-D Helmholtz equation

$$\begin{cases} -\Delta u - k^2 u + i\sigma_2 u = f(x, y, z), & (x, y, z) \in \Omega = [0, 1] \times [0, 1] \times [0, 1], \\ u|_\Gamma = g(x, y, z), & (x, y, z) \in \Gamma, \end{cases}$$

where $k$ and $\Gamma$ denote the wave number and the boundary of $\Omega = [0, 1] \times [0, 1] \times [0, 1]$, respectively. Let $\sigma_2$ and $k$ be 0.1 and 1, respectively. Using the center-difference scheme to the above 3-D Helmholtz equation with the mesh-size $h = \frac{1}{m+1}$, we have

$$A = W + iT,$$

**Table 10** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 2 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 7 (0.040) | 7 (0.124) | 7 (0.529) | 7 (2.637) | 7 (12.128) |
| PGSOR-GMRES | 7 (0.046) | 6 (0.145) | 6 (0.681) | 6 (2.831) | 6 (13.279) |
| RBLT-GMRES | 8 (0.049) | 8 (0.162) | 8 (0.730) | 8 (3.574) | 8 (15.938) |
| ISCSP-GMRES(20) | 3 (0.071) | 7 (0.525) | 23 (7.036) | 70 (125) | 228 (1579) |
| IPGSOR-GMRES(20) | 9 (0.205) | 20 (1.723) | 58 (23.424) | 190 (366) | >500 |
| IRBLT-GMRES(20) | 9 (0.202) | 23 (1.943) | 73 (29.034) | 242 (505) | >500 |

**Table 11** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 2 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 7 (0.039) | 7 (0.124) | 7 (0.523) | 7 (2.591) | 7 (12.099) |
| PGSOR-GMRES | 8 (0.049) | 8 (0.163) | 8 (0.720) | 8 (3.516) | 8 (15.876) |
| RBLT-GMRES | 8 (0.049) | 8 (0.162) | 8 (0.730) | 8 (3.574) | 8 (15.938) |
| ISCSP-GMRES(20) | 3 (0.071) | 7 (0.525) | 23 (7.210) | 70 (124) | 228 (1577) |
| IPGSOR-GMRES(20) | 9 (0.199) | 23 (1.943) | 70 (27.887) | 234 (454) | >500 |
| IRBLT-GMRES(20) | 9 (0.202) | 23 (1.943) | 73 (29.034) | 242 (505) | >500 |

where $W = T_m \otimes I_m \otimes I_m + I_m \otimes T_m \otimes I_m + I_m \otimes I_m \otimes T_m - k^2h^2(I_m \otimes I_m \otimes I_m)$ and $T_m = \sigma_2(I_m \otimes I_m \otimes I_m)$. Moreover, $I_m \in \mathbb{R}^{m\times m}$ is the identity matrix and $T_m$ is the tridiagonal matrix defined by $T_m = \text{tridiag}(-1, 2, -1)$. We take the right-hand side vector $b$ to be $b = (1+i)A\mathbf{1}$, with $\mathbf{1}$ being the vector of all entries equal to 1.

Tables 12, 13 and 14 present results for Example 3 where the CPU times are shown in parentheses. Table 13 shows ITs and CPUs when the optimal parameters $\omega$ and $\alpha$ are adopted, while Table 14 shows ITs and CPUs when the iteration parameters $\omega$ and $\alpha$ are set to be 1 and $\frac{2}{\sqrt{2}+1}$, respectively. The symbol † indicates that the computing time has lasted more than one hour. As seen, the direct method is sometimes slightly faster and is slower than SCSP-GMRES and PGSOR-GMRES for $m = 512, 1024$.

From Table 13, we see that among exact preconditioned GMRES methods, the performance of SCSP- and PGSOR-GMRES is better than RBLT-GMRES from point of view of both the iteration number and the CPU time. It is important to note that, among exact preconditioned GMRES methods, the computing time of the PGSOR-GMRES, for large $m$, is the least. Among inexact preconditioned GMRES($\ell$) methods, we see that the ITs and CPUs of ISCSP-GMRES(20) are the least.

**Table 12** IT and CPU for GMRES, GMRES(20), ILU-GMRES, ILU-GMRES(20) and $A\backslash b$ for Example 3

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| $A\backslash b$ | (0.034) | (0.175) | (0.910) | (5.341) | (32.357) |
| GMRES | 192 (2.172) | 350 (23.004) | >500 | >500 | † |
| GMRES(20) | 31 (0.529) | 70 (4.062) | 245 (53.350) | >500 | † |
| ILU-GMRES | 60 (0.254) | 108 (2.505) | 185 (25.256) | 325 (407) | † |
| ILU-GMRES(20) | 5 (0.126) | 11 (0.869) | 23 (7.182) | 64 (118) | 183 (1304) |

**Table 13** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 3 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 9 (0.047) | 11 (0.190) | 12 (0.952) | 13 (4.945) | 14 (26.137) |
| PGSOR-GMRES | 7 (0.050) | 8 (0.201) | 8 (0.916) | 9 (4.877) | 9 (25.205) |
| RBLT-GMRES | 10 (0.029) | 9 (0.213) | 9 (0.998) | 9 (5.245) | 9 (26.209) |
| ISCSP-GMRES(20) | 5 (0.108) | 11 (0.824) | 23 (7.341) | 63 (112) | 172 (1229) |
| IPGSOR-GMRES(20) | 7 (0.171) | 26 (2.180) | 92 (35.982) | 216 (485) | † |
| IRBLT-GMRES(20) | 13 (0.275) | 34 (2.718) | 108 (41.105) | 368 (742) | † |

In addition, when the results of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ are compared with those of the optimal values, for each $m$, we see that the ITs and CPUs of the RBLT-GMRES and IRBLT-GMRES(10) methods in Table 14 are exactly the same as those in Table 13. Also, the ITs and CPUs of the SCSP-GMRES method in Table 14 are exactly the same as those in Table 13 except for $m = 64, 512$, in which cases the difference of IT is only one. The results of the PGSOR-GMRES method in Table 14 are close to those in Table 13, and for some cases the results are the same. The ITs of the ISCSP-GMRES(20) method in Table 14 are the same as those in Table 13 except for $m = 512$, in which case the difference of IT is only one. Thus, it can be said that the performance of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is acceptable and close to those of the optimal values, especially for the SCSP preconditioner.

Tables 15, 16 and 17 present results for Example 4 where the CPU times are shown in parentheses. Table 16 shows ITs and CPUs when the optimal parameters $\omega$ and $\alpha$ are adopted, while Table 17 shows ITs and CPUs when the iteration parameters $\omega$ and $\alpha$ are set to be 1 and $\frac{2}{\sqrt{2}+1}$, respectively. The symbol † indicates that the computing time has lasted more than one hour. As seen, the direct method is slower than SCSP-GMRES and PGSOR-GMRES methods.

**Table 14** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 3 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Method | $m^2$ | | | | |
|---|---|---|---|---|---|
| | $64^2$ | $128^2$ | $256^2$ | $512^2$ | $1024^2$ |
| SCSP-GMRES | 10 (0.049) | 11 (0.190) | 12 (0.952) | 14 (5.519) | 14 (26.209) |
| PGSOR-GMRES | 11 (0.050) | 10 (0.231) | 9 (0.978) | 9 (4.832) | 9 (25.008) |
| RBLT-GMRES | 10 (0.029) | 9 (0.213) | 9 (0.998) | 9 (4.945) | 9 (26.137) |
| ISCSP-GMRES(20) | 5 (0.108) | 11 (0.824) | 23 (7.341) | 64 (115) | 172 (1206) |
| IPGSOR-GMRES(20) | 13 (0.275) | 33 (2.683) | 105 (40.189) | 356 (682) | † |
| IRBLT-GMRES(20) | 13 (0.275) | 34 (2.718) | 108 (41.105) | 368 (742) | † |

**Table 15** IT and CPU for GMRES, GMRES(20), ILU-GMRES, ILU-GMRES(20) and $A\backslash b$ for Example 4

| Method | $m^3$ | | | | |
|---|---|---|---|---|---|
| | $30^3$ | $35^3$ | $40^3$ | $45^3$ | $50^3$ |
| $A\backslash b$ | (2.992) | (6.608) | (14.676) | (185) | † |
| GMRES | 57 (1.121) | 64 (2.108) | 70 (3.643) | 76 (6.195) | 81 (9.794) |
| GMRES(20) | 5 (0.485) | 6 (0.915) | 6 (1.350) | 6 (2.005) | 6 (2.736) |
| ILU-GMRES | 22 (0.301) | 24 (0.565) | 26 (0.875) | 28 (1.492) | 29 (2.241) |
| ILU-GMRES(20) | 2 (0.281) | 2 (0.448) | 2 (0.668) | 2 (0.980) | 2 (1.599) |

From Table 16, we see that among exact preconditioned GMRES methods, the performance of SCSP- and PGSOR-GMRES is better than RBLT-GMRES from point of view of both the iteration number and the CPU time. It is important to note that, among the exact preconditioned GMRES methods, the computing time of the SCSP-GMRES is the least, except for $m = 50$ in which case that of the PGSOR-GMRES is the least. Among inexact preconditioned GMRES($\ell$) methods, we see that the ITs and CPUs of ISCSP-GMRES(20) are the least.

In addition, when the results of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ are compared with those of the optimal values, for each $m$, we see that, among exact preconditioned GMRES methods, the ITs of the RBLT-GMRES and SCSP-GMRES in Table 17 are exactly the same as those in Table 16. Also, among inexact preconditioned GMRES($\ell$) methods, the ITs of all of them in Table 17 are exactly the same as those in Table 16. Thus, it can be said that the performance of $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$ is acceptable and close to those of the optimal values.

In Figs. 2, 3, 4, 5, 6, 7 and 8 we depict the eigenvalues distribution of the coefficient matrix $A$ and SCSP-preconditioned matrices $M_\omega^{-1}A$, when $\omega = \omega^*$ and $\omega = 1$, and also the eigenvalues distribution of the block two-by-two coefficient matrix $\mathcal{R}$ and PGSOR-preconditioned matrices $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ and $\mathcal{P}^{-1}(1, \frac{2}{\sqrt{2}+1})\mathcal{R}$

**Table 16** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 4 when $\omega = \omega^*$ and $\alpha = \alpha^*$

| Method | $m^3$ | | | | |
|---|---|---|---|---|---|
| | $30^3$ | $35^3$ | $40^3$ | $45^3$ | $50^3$ |
| SCSP-GMRES | 9 (1.134) | 10 (2.437) | 10 (4.691) | 11 (10.074) | 12 (16.867) |
| PGSOR-GMRES | 8 (1.279) | 7 (2.456) | 8 (4.865) | 8 (10.086) | 8 (16.510) |
| RBLT-GMRES | 9 (1390) | 9 (2.852) | 9 (5.238) | 9 (10.660) | 9 (17.462) |
| ISCSP-GMRES(20) | 2 (0.263) | 2 (0.417) | 2 (0.627) | 2 (0.932) | 2 (1.329) |
| IPGSOR-GMRES(20) | 3 (0.441) | 3 (0.707) | 3 (1.063) | 3 (1.972) | 3 (2.539) |
| IRBLT-GMRES(20) | 3 (0.441) | 3 (0.709) | 3 (1.069) | 3 (1.779) | 3 (2.542) |

**Table 17** IT and CPU for exact SCSP-, PGSOR- and RBLT-preconditioned GMRES and inexact SCSP-, PGSOR- and RBLT-preconditioned GMRES($\ell$) methods for Example 4 when $\omega = 1$ and $\alpha = \frac{2}{\sqrt{2}+1}$

| Method | $m^3$ | | | | |
|---|---|---|---|---|---|
| | $30^3$ | $35^3$ | $40^3$ | $45^3$ | $50^3$ |
| SCSP-GMRES | 9 (1.170) | 10 (2.430) | 10 (4.943) | 11 (9.851) | 12 (17.053) |
| PGSOR-GMRES | 9 (1.384) | 9 (2.746) | 9 (5.199) | 9 (10.605) | 9 (17.396) |
| RBLT-GMRES | 9 (1390) | 9 (2.852) | 9 (5.238) | 9 (10.660) | 9 (17.462) |
| ISCSP-GMRES(20) | 2 (0.262) | 2 (0.419) | 2 (0.630) | 2 (0.906) | 2 (1.357) |
| IPGSOR-GMRES(20) | 3 (0.445) | 3 (0.707) | 3 (1.064) | 3 (1.825) | 3 (2.544) |
| IRBLT-GMRES(20) | 3 (0.44) | 3 (0.709) | 3 (1.069) | 3 (1.779) | 3 (2.542) |

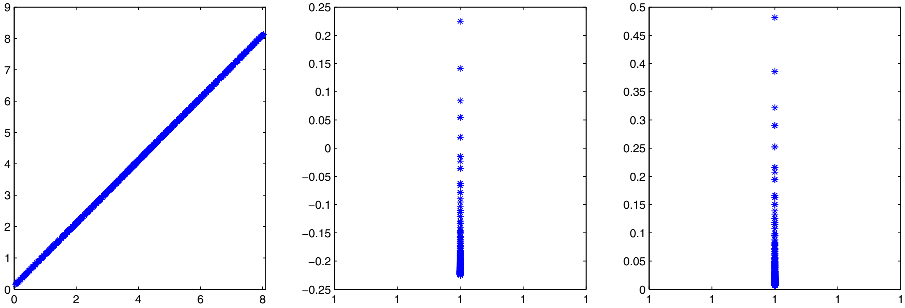

**Fig. 2** Eigenvalues distribution of the matrix $A$ (*left*) and the preconditioned matrix, $M_{\omega}^{-1}A$ where $\omega = \omega^*$ (*middle*) and $M_{\omega}^{-1}A$ where $\omega = 1$ (*right*) for Example 1 with $m = 32$

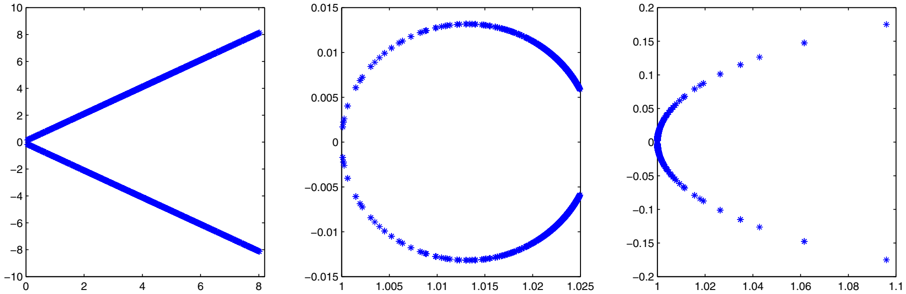

**Fig. 3** Eigenvalues distribution of the block two-by-two matrix $\mathcal{R}$ (*left*), the preconditioned matrix $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ (*middle*) and the preconditioned matrix $\mathcal{P}^{-1}(1, \frac{2}{\sqrt{2}+1})\mathcal{R}$ (*right*) for Example 1 with $m = 32$

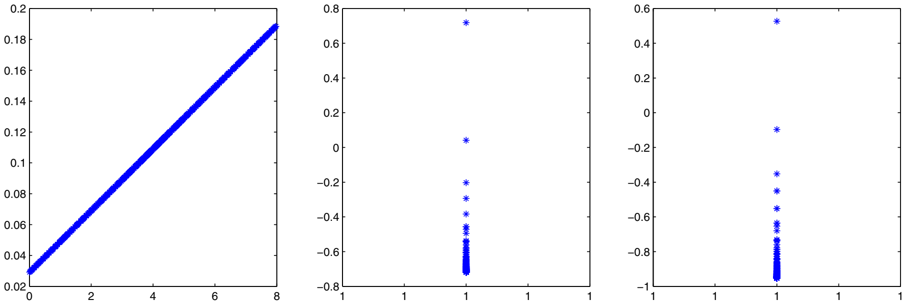

**Fig. 4** Eigenvalues distribution of the matrix $A$ (*left*) and the preconditioned matrix, $M_\omega^{-1}A$ where $\omega = \omega^*$ (*middle*) and $M_\omega^{-1}A$ where $\omega = 1$ (*right*) for Example 2 with $m = 32$

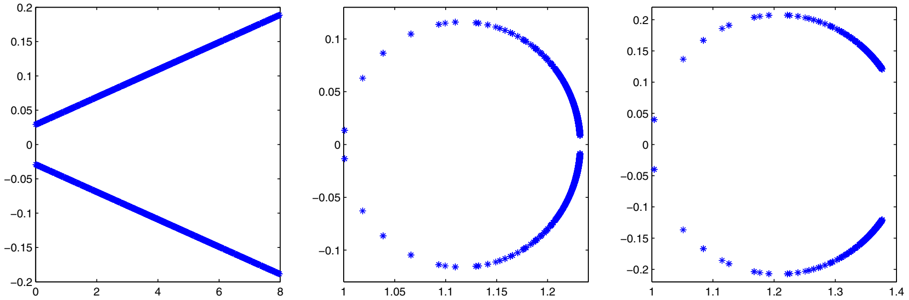

**Fig. 5** Eigenvalues distribution of the block two-by-two matrix $\mathcal{R}$ (*left*), the preconditioned matrix $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ (*middle*) and the preconditioned matrix $\mathcal{P}^{-1}(1, \frac{2}{\sqrt{2}+1})\mathcal{R}$ (*right*) for Example 2 with $m = 32$

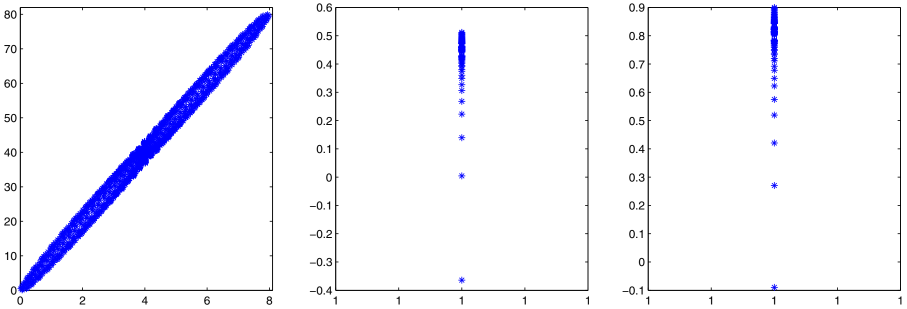

**Fig. 6** Eigenvalues distribution of the matrix $A$ (*left*) and the preconditioned matrix, $M_\omega^{-1}A$ where $\omega = \omega^*$ (*middle*) and $M_\omega^{-1}A$ where $\omega = 1$ (*right*) for Example 3 with $m = 32$

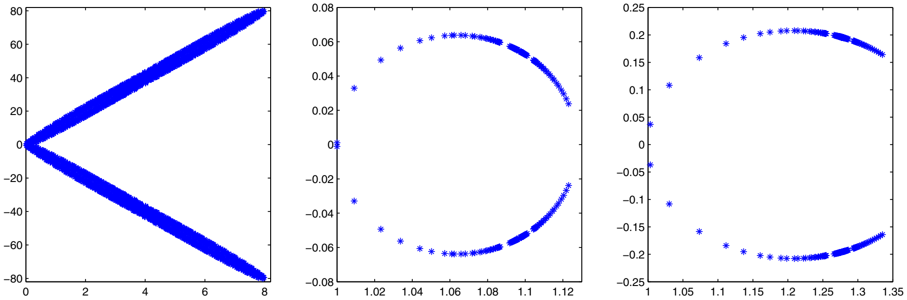

**Fig. 7** Eigenvalues distribution of the block two-by-two matrix $\mathcal{R}$ (*left*), the preconditioned matrix $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ (*middle*) and the preconditioned matrix $\mathcal{P}^{-1}(1, \frac{2}{\sqrt{2}+1})\mathcal{R}$ (*right*) for Example 2 with $m = 32$

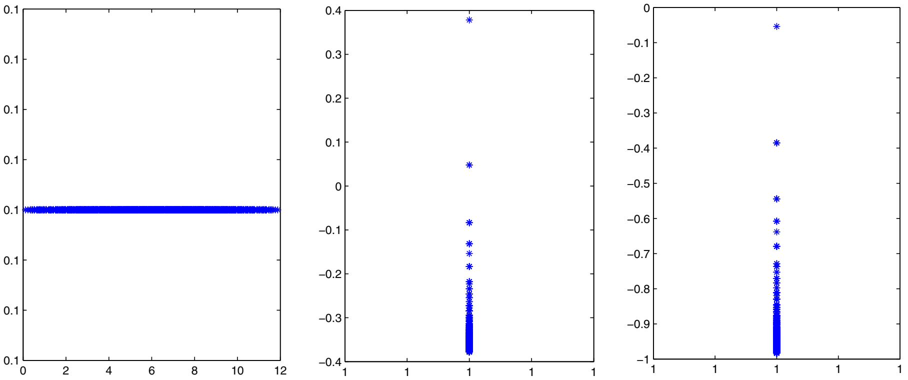

**Fig. 8** Eigenvalues distribution of the block two-by-two matrix $\mathcal{R}$ (*left*), the preconditioned matrix $\mathcal{P}^{-1}(\omega^*, \alpha^*)\mathcal{R}$ (*middle*) and the preconditioned matrix $\mathcal{P}^{-1}(1, \frac{2}{\sqrt{2}+1})\mathcal{R}$ (*right*) for Example 4 with $m = 15$

with $m = 32$ for Examples 1–3. It is evident that systems which are preconditioned by SCSP and PGSOR method are of a well-clustered spectrum around $(1, 0)$. These observations imply that when SCSP and GSOR are applied as a preconditioner for GMRES, the rate of convergence can be improved considerably. This fact is further confirmed by the numerical results presented in Tables 7–14 (Fig. 8).

## 5 Conclusion

In this paper we have presented a new stationary iterative method, called Scale-Splitting (SCSP) method, for solving original complex linear system (1). Convergence properties of the method have been also investigated. We have further explored algebraic and convergence properties of the PGSOR iteration method for solving the block two-by-two equivalent linear system (2) obtained from (1). We also analyzed the eigenvalue properties of the SCSP and PGSOR preconditioned matrices. Both theoretical analysis and numerical performances have shown that the SCSP and PGSOR iteration methods, and the SCSP and PGSOR preconditioners at accelerating the convergence rates of Krylov subspace iteration methods such as GMRES for solving the linear system (1) and (2), respectively, are competitive, and they are advantageous over rotated block triangular preconditioners. Moreover, we presented the ISCSP and IPGSOR preconditioning matrices to precondition linear system (1) and (2), respectively, based on the SCSP and PGSOR preconditioners and showed the effectiveness of these preconditioners at accelerating the convergence rates of Krylov subspace iteration methods such as GMRES on the basis of numerical results.

**Acknowledgments** The authors would like to thank the anonymous referees for their valuable comments and suggestions which substantially improved the quality of the paper. The work of the second author is partially supported by University of Guilan.

## References

1. Hiptmair, R.: Finite elements in computational electromagnetism. Acta Numer. **11**, 237–339 (2002)
2. van Rienen, U.: Numerical Methods in Computational Electrodynamics: Linear Systems in Practical Applications. Springer, Berlin (2001)
3. Bertaccini, D.: Efficient solvers for sequences of complex symmetric linear systems. Electr. Trans. Numer. Anal. **18**, 49–64 (2004)
4. 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)
5. Arridge, S.R.: Optical tomography in medical imaging. Inverse Probl. **15**, 41–93 (1999)
6. Dijk, W.V., Toyama, F.M.: Accurate numerical solutions of the time-dependent Schrödinger equation. Phys. Rev. E **75**, 1–10 (2007)
7. Poirier, B.: Efficient preconditioning scheme for block partitioned matrices with structured sparsity. Numer. Linear Algebra Appl. **7**, 715–726 (2000)
8. Benzi, M., Bertaccini, D.: Block preconditioning of real-valued iterative algorithms for complex linear systems. IMA J. Numer. Anal. **28**, 598–618 (2008)
9. Bai, Z.Z., Golub, G.H., Ng, M.K.: Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems. SIAM J. Matrix Anal. Appl. **24**, 603–626 (2003)
10. Bai, Z.Z., Benzi, M., Chen, F.: Modified HSS iteration methods for a class of complex symmetric linear systems. Computing **87**, 93–111 (2010)
11. Bai, Z.Z., Benzi, M., Chen, F.: On preconditioned MHSS iteration methods for complex symmetric linear systems. Numer. Algor. **56**, 297–317 (2011)
12. Li, X., Yang, A.L., Wu, Y.J.: Lopsided PMHSS iteration method for a class of complex symmetric linear systems. Numer. Algor. **66**, 555–568 (2014)
13. van der Vorst, H.A., Melissen, J.B.M.: A Petrov-Galerkin type method for solving $Ax = b$, where $A$ is symmetric complex. IEEE Trans. Mag. **26**, 706–708 (1990)
14. Freund, R.W.: Conjugate gradient-type methods for linear systems with complex symmetric coefficient matrices. SIAM J. Sci. Stat. Comput. **13**, 425–448 (1992)
15. Sogabe, T., Zhang, S.L.: A COCR method for solving complex symmetric linear systems. J. Comput. Appl. Math. **199**, 297–303 (2007)
16. Gu, X.M., Clemens, M., Huang, T.Z., Li, L.: The SCBiCG class of algorithms for complex symmetric linear systems with applications in several electromagnetic model problems. Comput. Phys. Commun. **191**, 52–64 (2015)
17. Gu, X.M., Huang, T.Z., Li, L., Li, H.-B., Sogabe, T., Clemens, M.: Quasi-minimal residual variants of the COCG and COCR methods for complex symmetric linear systems in electromagnetic simulations. IEEE Trans. Microw. Theory Technol. **62**, 2859–2867 (2014)

18. Bai, Z.Z., Benzi, M., Chen, F.: 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**, 343–369 (2013)
19. Salkuyeh, D.K., Hezari, D., Edalatpour, V.: Generalized SOR iterative method for a class of complex symmetric linear system of equations. Int. J. Comput. Math. **92**, 802–815 (2015)
20. Bai, Z.Z.: On preconditioned iteration methods for complex linear systems. J. Eng. Math. **93**, 41–60 (2015)
21. Bai, Z.Z., Chen, F., Wang, Z.Q.: Additive block diagonal preconditioning for block two-by-two linear systems of skew-Hamiltonian coefficient matrices. Numer. Algorithm **64**, 655–675 (2013)
22. Bai, Z.Z.: Rotated block triangular preconditioning based on PMHSS. Sci. China Math. **56**, 2523–2538 (2013)
23. Lang, C., Ren, Z.-R.: Inexact rotated block triangular preconditioners for a class of block two-by-two matrices. J. Eng. Math. **93**, 87–98 (2015)
24. Hezari, D., Edalatpour, V., Salkuyeh, D.K.: Preconditioned GSOR iteration method for a class of complex symmetric linear system. Numer. Linear Algebra Appl. **22**, 761–776 (2015)
25. Saad, Y., Schultz, M.H.: GMRES: a generalized minimal residual algorithm for nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **7**, 856–869 (1986)
26. Bai, Z.Z.: Structured preconditioners for nonsingular matrices of block two-by-two structures. Math. Comput. **75**, 791–815 (2006)
27. Bai, Z.Z., Ng, M.K.: On inexact preconditioners for nonsymmetric matrices. SIAM J. Sci. Comput. **26**, 1710–1724 (2005)
28. Zhang, J., Dai, H.: A new splitting preconditioner for the iterative solution of complex symmetric indefinite linear systems. Appl. Math. Lett. **49**, 100–106 (2015)