

## A PRECONDITIONED SSOR ITERATION METHOD FOR SOLVING COMPLEX SYMMETRIC SYSTEM OF LINEAR EQUATIONS

TAHEREH SALIMI SIAHKOLAEI

Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran

DAVOD KHOJASTEH SALKUYEH\*

Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran

(Communicated by the associate editor name)

**ABSTRACT.** We present a preconditioned version of the symmetric successive overrelaxation (SSOR) iteration method for a class of complex symmetric linear systems. The convergence results of the proposed method are established and conditions under which the spectral radius of the iteration matrix of the method is smaller than that of the SSOR method are analyzed. Numerical experiments illustrate the theoretical results and depict the efficiency of the new iteration method.

**1. Introduction.** Many scientific and engineering applications often require the solution of large and sparse complex symmetric system of linear equations

$$Au \equiv (W + iT)u = b, \quad A \in \mathbb{C}^{n \times n}, \quad u, b \in \mathbb{C}^n, \quad (1)$$

where  $W, T \in \mathbb{R}^{n \times n}$ . We assume that  $W$  and  $T$  are symmetric positive semidefinite with at least one of them, e.g.,  $W$ , being positive definite.

Several effective iteration methods have been proposed in the literature for solving the system (1). For solving such symmetric systems, Bunse-Gerstner et al. presented a new conjugate gradient-type iterative method in [11]. Splitting iteration methods such as generalized successive overrelaxation (GSOR) [18], preconditioned GSOR (PGSOR) [16], scale splitting (SCSP) [15], two-parameter two-step SCSP (TTSCSP) in [19] and parameterized splitting (PS) in [20] proposed for the linear system (1). Edalatpour et al. in [13] solved inexactly the linear system (1) by two efficient methods. Using the Hermitian and skew-Hermitian splitting [5] of the coefficient matrix  $A$ , Bai et al. proposed the modified HSS (MHSS) iteration method in [2] and preconditioned MHSS (PMHSS) iteration method in [6].

In [1] it has been shown that by assuming  $A = W + iT$ ,  $u = x + iy$  and  $b = p + iq$  and separating the real and imaginary parts of the system (1), it can be transformed into the real equivalent form

---

2010 *Mathematics Subject Classification.* 65F10, 65F50, 65F08.

*Key words and phrases.* Complex linear systems, symmetric positive definite, PMHSS, GSOR, SSOR, ASSOR..

The first author is supported by NSF grant xx-xxxx.

\* Corresponding author: xxxx.

$$\mathcal{A}u = \begin{bmatrix} W & -T \\ T & W \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} p \\ q \end{bmatrix}. \quad (2)$$

This linear system can be formally regarded as a special case of the generalized saddle point problem [9, 10]. Bai et al. in [7], applied the GSOR method for the block 2-by-2 linear system of different sub-blocks rather than the original complex linear system (1). Recently Salkuyeh et al. in [18] solved the real equivalent system (2) by the GSOR iterative method.

Under our hypotheses, it can be easily proved that the matrix  $\mathcal{A}$  is nonsingular. In [18] it has been shown that if  $W$  and  $T$  are symmetric positive definite and symmetric, respectively, then the GSOR method is convergent.

In [17], Liang et al. have applied the symmetric SOR (SSOR) method to the following preconditioned linear system

$$\begin{bmatrix} I & I \\ -I & I \end{bmatrix} \begin{bmatrix} W & -T \\ T & W \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} I & I \\ -I & I \end{bmatrix} \begin{bmatrix} p \\ q \end{bmatrix}, \quad (3)$$

and introduced the accelerated variant of the SSOR (ASSOR) method, in order to increase the convergence rate of the method. They proved that, under certain conditions on  $\omega$ , the ASSOR method is convergent and obtained the optimal value of the iteration parameter  $\omega$ .

For improving the proposed method, we use the preconditioner

$$\mathcal{P}_\alpha = \begin{bmatrix} \alpha I & I \\ -I & \alpha I \end{bmatrix}, \quad (4)$$

where  $0 < \alpha \in \mathbb{R}$  and instead of solving the linear system (2), we solve the preconditioned linear system

$$\mathcal{P}_\alpha \mathcal{A} \begin{bmatrix} x \\ y \end{bmatrix} = \mathcal{P}_\alpha \begin{bmatrix} p \\ q \end{bmatrix}, \quad (5)$$

by the SSOR method and define the PSSOR method (for preconditioned version of SSOR). Evidently, when  $\alpha = 1$  the PSSOR method reduces to the ASSOR method. We study conditions under which the spectral radius of the PSSOR iteration matrix becomes smaller than that of the SSOR method. It is noteworthy that the motivation of choosing this preconditioner stems from [8] in which Benzi et al. have presented some examples of preconditioners for the real formulation (2) of the system (1).

Throughout this paper, for a square matrix  $B$ ,  $\rho(B)$  and  $\sigma(B)$  denote for the spectral radius and spectrum of  $B$ , respectively.

The rest of the paper is organized as follows. In Section 2, we present the PSSOR method and discuss its convergence properties. Section 3 is devoted to numerical experiments to verify the theoretical results given in Section 2. Finally, in Section 4 we present a brief concluding remarks to end the paper.

**2. The PSSOR method and its convergence.** In this section, we present the PSSOR iterative method and study its convergence properties. It is easy to see that the system (5) is equivalent to

$$\begin{bmatrix} W_\alpha & -T_\alpha \\ T_\alpha & W_\alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} p_\alpha \\ q_\alpha \end{bmatrix} = f_\alpha, \quad (6)$$

where  $W_\alpha = \alpha W + T$ ,  $T_\alpha = \alpha T - W$  and  $p_\alpha = \alpha p + q$ ,  $q_\alpha = \alpha q - p$ . Now, we use the SSOR method to solve the preconditioned linear system (6) and split the matrix  $\mathcal{A}_\alpha = \mathcal{P}_\alpha \mathcal{A}$  as  $\mathcal{A}_\alpha = \mathcal{D}_\alpha - \mathcal{L}_\alpha - \mathcal{U}_\alpha$  with

$$\mathcal{D}_\alpha = \begin{bmatrix} W_\alpha & 0 \\ 0 & W_\alpha \end{bmatrix}, \quad \mathcal{L}_\alpha = \begin{bmatrix} 0 & 0 \\ -T_\alpha & 0 \end{bmatrix}, \quad \mathcal{U}_\alpha = \begin{bmatrix} 0 & T_\alpha \\ 0 & 0 \end{bmatrix}.$$

According to this splitting, the SSOR method for solving the system (6) is given by

$$\begin{cases} (\mathcal{D}_\alpha - \omega \mathcal{L}_\alpha) \begin{bmatrix} x^{(k+\frac{1}{2})} \\ y^{(k+\frac{1}{2})} \end{bmatrix} = ((1-\omega)\mathcal{D}_\alpha + \omega \mathcal{U}_\alpha) \begin{bmatrix} x^{(k)} \\ y^{(k)} \end{bmatrix} + \omega \begin{bmatrix} p_\alpha \\ q_\alpha \end{bmatrix}, \\ (\mathcal{D}_\alpha - \omega \mathcal{U}_\alpha) \begin{bmatrix} x^{(k+1)} \\ y^{(k+1)} \end{bmatrix} = ((1-\omega)\mathcal{D}_\alpha + \omega \mathcal{L}_\alpha) \begin{bmatrix} x^{(k+\frac{1}{2})} \\ y^{(k+\frac{1}{2})} \end{bmatrix} + \omega \begin{bmatrix} p_\alpha \\ q_\alpha \end{bmatrix}. \end{cases} \quad (7)$$

Obviously, the above system can be rewritten as

$$\begin{bmatrix} x^{(k+1)} \\ y^{(k+1)} \end{bmatrix} = \mathcal{H}_{\omega,\alpha} \begin{bmatrix} x^{(k)} \\ y^{(k)} \end{bmatrix} + \mathcal{C}_{\omega,\alpha}^{-1} f_\alpha, \quad (8)$$

in which

$$\mathcal{H}_{\omega,\alpha} = \begin{bmatrix} H_{11}(\omega, \alpha) & H_{12}(\omega, \alpha) \\ H_{21}(\omega, \alpha) & H_{22}(\omega, \alpha) \end{bmatrix},$$

is the iteration matrix of the PSSOR method, with

$$\begin{cases} H_{11}(\omega, \alpha) = (1-\omega)^2 I_n - \omega^2(1-\omega)(2-\omega)(W_\alpha^{-1} T_\alpha)^2, \\ H_{12}(\omega, \alpha) = \omega(1-\omega)(2-\omega)W_\alpha^{-1} T_\alpha - \omega^3(2-\omega)(W_\alpha^{-1} T_\alpha)^3, \\ H_{21}(\omega, \alpha) = -\omega(1-\omega)(2-\omega)W_\alpha^{-1} T_\alpha, \\ H_{22}(\omega, \alpha) = (1-\omega)^2 I_n - \omega^2(2-\omega)(W_\alpha^{-1} T_\alpha)^2, \end{cases} \quad (9)$$

and

$$\mathcal{C}_{\omega,\alpha}^{-1} = \omega(2-\omega) \begin{bmatrix} W_\alpha^{-1} - \omega^2 W_\alpha^{-1} T_\alpha W_\alpha^{-1} T_\alpha W_\alpha^{-1} & \omega W_\alpha^{-1} T_\alpha W_\alpha^{-1} \\ -\omega W_\alpha^{-1} T_\alpha W_\alpha^{-1} & W_\alpha^{-1} \end{bmatrix}.$$

It is not difficult to see that  $\mathcal{A}_\alpha = \mathcal{M}_{\omega,\alpha} - \mathcal{N}_{\omega,\alpha}$  and  $\mathcal{H}_{\omega,\alpha} = \mathcal{M}_{\omega,\alpha}^{-1} \mathcal{N}_{\omega,\alpha}$ , where

$$\begin{aligned} \mathcal{M}_{\omega,\alpha} &= \frac{1}{\omega(2-\omega)} (\mathcal{D}_\alpha - \omega \mathcal{L}_\alpha) (\mathcal{D}_\alpha - \omega \mathcal{U}_\alpha) \\ \mathcal{N}_{\omega,\alpha} &= \frac{1}{\omega(2-\omega)} ((1-\omega)\mathcal{D}_\alpha + \omega \mathcal{L}_\alpha) ((1-\omega)\mathcal{D}_\alpha + \omega \mathcal{U}_\alpha). \end{aligned}$$

Therefore the matrix  $\mathcal{M}_{\omega,\alpha}$  can be used as a preconditioner for the system (5).

From (7), the PSSOR method for the solving the system (2) can be stated as follows.

**The PSSOR iteration method:** Let  $(x^{(0)}; y^{(0)}) \in \mathbb{R}^n$  be an initial guess and relaxation factor  $\omega$  with  $\omega > 0$ . For  $k = 0, 1, 2, \dots$ , until  $\{(x^{(k)}; y^{(k)})\}$  converges, compute  $(x^{(k+1)}; y^{(k+1)})$  according to the following sequence

$$\begin{cases} y^{(k+1)} = (1-\omega)^2 y^{(k)} - \omega(2-\omega)W_\alpha^{-1}((1-\omega)T_\alpha x^{(k)} + \omega T_\alpha W_\alpha^{-1}(T_\alpha y^{(k)} + p_\alpha) - q_\alpha), \\ x^{(k+1)} = (1-\omega)^2 x^{(k)} + \omega W_\alpha^{-1}((1-\omega)T_\alpha y^{(k)} + T_\alpha y^{(k+1)} + (2-\omega)p_\alpha). \end{cases} \quad (10)$$

**Remark 1.** If  $W$  and  $T$  are respectively symmetric positive definite and symmetric positive semidefinite and  $\alpha$  is positive constant, then  $W_\alpha = \alpha W + T$  and  $T_\alpha = \alpha T - W$  are symmetric positive definite and symmetric, respectively.

**Theorem 2.1.** *Let  $W$  and  $T \in \mathbb{R}^{n \times n}$  be symmetric positive definite and symmetric positive semidefinite, respectively. Let also  $W_\alpha = \alpha W + T$  and  $T_\alpha = \alpha T - W$ , where  $\alpha$  is a positive constant. Then, the PSSOR method is convergent if one of the following conditions holds:*

- (i) if  $\rho(S_\alpha) \leq 1$ , then  $0 < \omega < 2$  should be satisfied;
- (ii) if  $\rho(S_\alpha) > 1$ , then  $0 < \omega < 1 - \sqrt{\frac{\rho(S_\alpha)-1}{\rho(S_\alpha)+1}}$  or  $1 + \sqrt{\frac{\rho(S_\alpha)-1}{\rho(S_\alpha)+1}} < \omega < 2$  should be satisfied,

where  $S_\alpha = (\alpha I + G)^{-1}(\alpha G - I)$  and  $G = W^{-\frac{1}{2}}TW^{-\frac{1}{2}}$ . The optimal value of the iteration parameters  $\omega$  and  $\alpha$  which minimize the spectral radius of the PSSOR method are given by

$$\omega_{opt} = 1 \pm \frac{\sqrt{\rho(S_{\alpha_{opt}})^2 + 1} - 1}{\rho(S_{\alpha_{opt}})}, \quad \alpha_{opt} = \frac{1 - \mu_{\min}\mu_{\max} + \sqrt{(1 + \mu_{\min}^2)(1 + \mu_{\max}^2)}}{\mu_{\min} + \mu_{\max}}, \quad (11)$$

where  $\mu_{\min}$  and  $\mu_{\max}$  are the smallest and largest eigenvalues of  $G$ , respectively. The corresponding optimal convergence factor of the PSSOR method is

$$\rho(\mathcal{H}_{\omega_{opt}, \alpha_{opt}}) = 1 - \frac{2}{\sqrt{\rho(S_{\alpha_{opt}})^2 + 1} + 1}. \quad (12)$$

*Proof.* Since  $W$  is symmetric positive definite, there is a symmetric positive definite matrix  $W^{\frac{1}{2}}$ , such that  $W = W^{\frac{1}{2}}W^{\frac{1}{2}}$ . Then, we have

$$W_\alpha^{-1}T_\alpha = (\alpha W + T)^{-1}(\alpha T - W) = W^{-\frac{1}{2}}S_\alpha W^{\frac{1}{2}}.$$

Therefore from (9), we can easily write

$$\begin{cases} H_{11}(\omega, \alpha) = W^{-\frac{1}{2}}H_{11}^*(\omega, \alpha)W^{\frac{1}{2}}; & H_{11}^*(\omega, \alpha) = (1 - \omega)^2I_n - \omega^2(1 - \omega)(2 - \omega)S_\alpha^2, \\ H_{12}(\omega, \alpha) = W^{-\frac{1}{2}}H_{12}^*(\omega, \alpha)W^{\frac{1}{2}}; & H_{12}^*(\omega, \alpha) = \omega(1 - \omega)(2 - \omega)S_\alpha - \omega^3(2 - \omega)S_\alpha^3, \\ H_{21}(\omega, \alpha) = W^{-\frac{1}{2}}H_{21}^*(\omega, \alpha)W^{\frac{1}{2}}; & H_{21}^*(\omega, \alpha) = -\omega(1 - \omega)(2 - \omega)S_\alpha, \\ H_{22}(\omega, \alpha) = W^{-\frac{1}{2}}H_{22}^*(\omega, \alpha)W^{\frac{1}{2}}; & H_{22}^*(\omega, \alpha) = (1 - \omega)^2I_n - \omega^2(2 - \omega)S_\alpha^2. \end{cases}$$

Defining

$$\varphi := \begin{bmatrix} W^{\frac{1}{2}} & 0 \\ 0 & W^{\frac{1}{2}} \end{bmatrix},$$

it follows from (9) that

$$\varphi \mathcal{H}_{\omega, \alpha} \varphi^{-1} = \begin{bmatrix} H_{11}^*(\omega, \alpha) & H_{12}^*(\omega, \alpha) \\ H_{21}^*(\omega, \alpha) & H_{22}^*(\omega, \alpha) \end{bmatrix} := \mathcal{H}_{\omega, \alpha}^*$$

Since the eigenvalues of the matrices  $\mathcal{H}_{\omega, \alpha}$  and  $\mathcal{H}_{\omega, \alpha}^*$  are the same, we study the eigenvalues of matrix  $\mathcal{H}_{\omega, \alpha}^*$ , instead of  $\mathcal{H}_{\omega, \alpha}$ .

Using Remark 1,  $W_\alpha$  and  $T_\alpha$  are symmetric positive definite and symmetric, respectively. Then the matrix  $S_\alpha$  is symmetric and has the spectral decomposition  $S_\alpha = \Gamma V \Gamma^{-1}$ , where  $V \in \mathbb{R}^{n \times n}$  is an invertible matrix and  $\Gamma = \text{diag}(\mu_1, \mu_2, \dots, \mu_n)$  in which  $\mu_i$ 's are the eigenvalues of  $S_\alpha$ . By defining

$$\Omega := \begin{bmatrix} V & 0 \\ 0 & V \end{bmatrix},$$

we have

$$\Omega^{-1}\mathcal{H}_{\omega,\alpha}^*\Omega = \begin{bmatrix} (1-\omega)^2I_n - \omega^2(1-\omega)(2-\omega)\Gamma^2 & \omega(1-\omega)(2-\omega)\Gamma - \omega^3(2-\omega)\Gamma^3 \\ -(1-\omega)(2-\omega)\Gamma & (1-\omega)^2I_n - \omega^2(2-\omega)\Gamma^2 \end{bmatrix}.$$

Hence the latter matrix is similar to  $\mathcal{H}_{\omega,\alpha}$ . Therefore, if  $\lambda$  is an eigenvalue of the above matrix, then we have

$$\lambda^2 - (2(1-\omega)^2 - \omega^2(2-\omega)^2\mu^2)\lambda + (1-\omega)^4 = 0, \quad (13)$$

for all  $\mu \in \sigma(S_\alpha)$ . Letting  $\beta = \omega(2-\omega)$ , Eq. (13) can be written as

$$\lambda^2 + (\beta^2\mu^2 + 2\beta - 2)\lambda + (\beta - 1)^2 = 0.$$

Then, from Theorem 1 in [18], we get the following convergence condition

$$0 < \beta = \omega(2-\omega) < \frac{2}{1 + \rho(S_\alpha)}. \quad (14)$$

After some simplifications, we conclude that the latter equation is equivalent to

- (i) if  $\rho(S_\alpha) \leq 1$ , then  $0 < \omega < 2$  should be satisfied;
- ii) if  $\rho(S_\alpha) > 1$ , then  $0 < \omega < 1 - \sqrt{\frac{\rho(S_\alpha)-1}{\rho(S_\alpha)+1}}$  or  $1 + \sqrt{\frac{\rho(S_\alpha)-1}{\rho(S_\alpha)+1}} < \omega < 2$  should be satisfied.

Also from Theorem 2 in [18], we see that for a given  $\alpha$  the optimal value of  $\beta$  is obtained from

$$\beta_{opt} = \frac{2}{1 + \sqrt{1 + \rho(S_\alpha)^2}},$$

and therefore from

$$\beta_{opt} = \omega_{opt}(2 - \omega_{opt}) = 2\omega_{opt} - \omega_{opt}^2,$$

we see that

$$\begin{aligned} \omega_{opt} &= 1 \pm \sqrt{1 - \beta_{opt}} = 1 \pm \sqrt{1 - \frac{2}{1 + \sqrt{1 + \rho(S_\alpha)^2}}} \\ &= 1 \pm \frac{\sqrt{\rho(S_\alpha)^2 + 1} - 1}{\rho(S_\alpha)}. \end{aligned}$$

On the other hand, using Theorem 2 in [18], we have

$$\rho(\mathcal{H}_{\omega_{opt},\alpha}) = 1 - \beta_{opt} = 1 - \frac{2}{1 + \sqrt{1 + \rho(S_\alpha)^2}}.$$

This equation shows that to make  $\rho(\mathcal{H}_{\omega_{opt},\alpha})$  minimum we should minimize the value of  $\rho(S_\alpha)$ . According to Theorem 2 in [16], the optimal value of the parameter  $\alpha$  which minimizes the spectral radius  $\rho(S_\alpha)$  is given by

$$\alpha_{opt} = \frac{1 - \mu_{\min}\mu_{\max} + \sqrt{(1 + \mu_{\min}^2)(1 + \mu_{\max}^2)}}{\mu_{\min} + \mu_{\max}}, \quad (15)$$

where  $\mu_{\min}$  and  $\mu_{\max}$  are the smallest and largest eigenvalues of  $G$ , respectively. Now, using

$$\rho(\mathcal{H}_{\omega_{opt},\alpha_{opt}}) = \min_{\alpha} \rho(\mathcal{H}_{\omega_{opt},\alpha}),$$

Eq. (12) is proved.  $\square$

The next theorem simplifies the convergence conditions of the PSSOR method.

**Theorem 2.2.** *Let  $W, T \in \mathbb{R}^{n \times n}$  be symmetric positive definite and symmetric positive semidefinite, respectively and  $\alpha$  be a positive constant. Suppose that  $\mu_{\min}$  and  $\mu_{\max}$  be the smallest and largest eigenvalues of  $G = W^{-\frac{1}{2}}TW^{-\frac{1}{2}}$ , respectively, and  $S_\alpha = (\alpha I + G)^{-1}(\alpha G - I)$ . The PSSOR method is convergent if one of the following conditions holds:*

- (i) *if  $\mu_{\max} > 1$ , then  $a < \alpha < d$  and  $0 < \omega < 2$  should be satisfied;*
  - (ii) *if  $\mu_{\max} > 1$ , then  $\alpha < a$  or  $\alpha > d$  and  $0 < \omega < 1 - c$  or  $1 + c < \omega < 2$  should be satisfied;*
  - (iii) *if  $\mu_{\max} \leq 1$ , then  $\alpha > a$  and  $0 < \omega < 2$  should be satisfied;*
  - (iv) *if  $\mu_{\max} \leq 1$ , then  $\alpha < a$  and  $0 < \omega < 1 - c$  or  $1 + c < \omega < 2$  should be satisfied;*
- where

$$a = \frac{1 - \mu_{\min}}{1 + \mu_{\min}}, \quad d = \frac{\mu_{\max} + 1}{\mu_{\max} - 1}, \quad c = \sqrt{\frac{\rho(S_\alpha) - 1}{\rho(S_\alpha) + 1}}. \quad (16)$$

*Proof.* We have seen that the PSSOR method is convergent, if one of the following conditions holds true:

- (a) *if  $\rho(S_\alpha) \leq 1$ , then  $0 < \omega < 2$  should be satisfied;*
- (b) *if  $\rho(S_\alpha) > 1$ , then  $0 < \omega < 1 - \sqrt{\frac{\rho(S_\alpha) - 1}{\rho(S_\alpha) + 1}}$  or  $1 + \sqrt{\frac{\rho(S_\alpha) - 1}{\rho(S_\alpha) + 1}} < \omega < 2$  should be satisfied.*

It is easy to see that the eigenvalues of  $S_\alpha = (\alpha I + G)^{-1}(\alpha G - I)$  are of the form

$$\lambda = \frac{\alpha\mu - 1}{\alpha + \mu} =: h(\mu),$$

where  $\mu$  is an eigenvalue of the  $G$ . Since, the function  $h$  is increasing, the spectral radius of  $S_\alpha$  is equal to

$$\rho(S_\alpha) = \max \left\{ \frac{1 - \alpha\mu_{\min}}{\alpha + \mu_{\min}}, \frac{\alpha\mu_{\max} - 1}{\alpha + \mu_{\max}} \right\}.$$

Hence  $\rho(S_\alpha) \leq 1$  if and only if

$$\max \left\{ \frac{1 - \alpha\mu_{\min}}{\alpha + \mu_{\min}}, \frac{\alpha\mu_{\max} - 1}{\alpha + \mu_{\max}} \right\} \leq 1. \quad (17)$$

It is easy to see that the latter relation is equivalent to

$$\begin{cases} 1 - \mu_{\min} \leq (1 + \mu_{\min})\alpha, \\ (\mu_{\max} - 1)\alpha \leq \mu_{\max} + 1. \end{cases} \quad (18)$$

The first inequality of (18) holds if and only if

$$\alpha \geq \frac{1 - \mu_{\min}}{1 + \mu_{\min}}.$$

Obviously, if  $\mu_{\max} \leq 1$ , then the second inequality of (18) always holds. When  $\mu_{\max} > 1$ , the second inequality of (18) is equivalent to

$$\alpha < \frac{\mu_{\max} + 1}{\mu_{\max} - 1}.$$

Therefore, from (a) we see that the PSSOR method is convergent in each of the cases (i) and (iii).

In the same manner, considering the case (b), it can be seen that the PSSOR method is convergent in the cases (ii) and (iv).  $\square$

**3. Numerical experiments.** In this section, we use three test problems from [2] to illustrate the feasibility and effectiveness of the PSSOR method to solve the system (2). We also compare the performance of this method with the PMHSS, GSOR, SSOR and ASSOR methods in terms of both the number of iterations (denoted by IT) and the total computing times denoted by CPU in the tables. The PMHSS iteration method are employed to solve the complex system (1) and other methods to solve the equivalent real system (2). In each iteration of the PMHSS, GSOR, SSOR, ASOR and PSSOR iteration methods, we use the Cholesky factorization of the coefficient matrices to solve the sub-systems.

All runs are implemented in MATLAB R2014b with a Laptop with 2.40 GHz central processing unit (Intel(R) Core(TM) i7-5500), 8 GB memory and Windows 10 operating system.

We use a zero vector as an initial guess and the stopping criterion

$$\frac{\|f - Au^{(k)}\|_2}{\|f\|_2} < 10^{-6},$$

where  $u^{(k)} = x^{(k)} + iy^{(k)}$  and  $f = (p; q)$ .

**Example.** (See [2]) Consider the system of linear equations

$$\left[ \left( K + \frac{3 - \sqrt{3}}{\tau} \right) + i \left( K + \frac{3 + \sqrt{3}}{\tau} I \right) \right] x = b, \quad (19)$$

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  $h = \frac{1}{m+1}$ . 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  $n = m^2$ . 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

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

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

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

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

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  $\omega$  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  $\mu = \pi$ ,  $\omega = \pi$ , 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  $h^2$ .

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

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

where  $V = \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{m \times m}$ ,  $V_c = V - e_1 e_m^T - e_m e_1^T \in \mathbb{R}^{m \times m}$  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.

Here  $T$  and  $W$  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  $h = 1/(m + 1)$ .

TABLE 1. The optimal parameters for MHSS, GSOR, SSOR, ASSOR and PSSOR.

			$m$					
Method			16	32	64	128	256	512
Example 1	PMHSS	$\alpha_{opt}$	1.09	1.36	1.35	1.05	1.05	1.05
	GSOR	$\alpha_{opt}$	0.551	0.495	0.457	0.432	0.418	0.412
	SSOR	$\omega_{opt}$	0.33	0.29	0.26	0.24	0.24	0.23
	ASSOR	$\omega_{opt}$	0.80	0.77	0.75	0.74	0.72	0.72
	PSSOR	$\alpha_{opt}$	0.47	0.48	0.54	0.54	0.55	0.55
		$\omega_{opt}$	0.83	0.83	0.82	0.82	0.82	0.82
Example 2	PMHSS	$\alpha_{opt}$	1.43	1.53	1.38	1.26	1.24	1.24
	GSOR	$\alpha_{opt}$	0.189	0.190	0.190	0.190	0.190	0.190
	SSOR	$\omega_{opt}$	0.09	0.09	0.10	0.10	0.10	0.10
	ASSOR	$\omega_{opt}$	0.64	0.64	0.64	0.64	0.64	0.64
	PSSOR	$\alpha_{opt}$	0.08	0.09	0.09	0.09	0.09	0.09
		$\omega_{opt}$	0.89	0.89	0.89	0.89	0.89	0.89
Example 3	PMHSS	$\alpha_{opt}$	0.61	0.42	0.57	0.78	0.73	0.73
	GSOR	$\alpha_{opt}$	0.908	0.776	0.566	0.354	0.199	0.105
	SSOR	$\omega_{opt}$	0.69	0.52	0.34	0.19	0.10	0.05
	ASSOR	$\omega_{opt}$	0.62	0.62	0.62	0.61	0.61	0.61
	PSSOR	$\alpha_{opt}$	1.93	1.50	1.31	1.02	0.90	0.90
		$\omega_{opt}$	0.82	0.74	0.68	0.62	0.61	0.61

In Table 1, the optimal parameters of the PMHSS, GSOR, SSOR, ASSOR and PSSOR methods are presented. The optimal parameters of the GSOR method are chosen based on Theorem 2 of [18] and the ones of the other methods are experimentally found.

In Tables 2, 3 and 4, we report the numerical results for Examples 3, 3 and 3. We present the iteration numbers (IT) and CPU time (CPU) for the PMHSS, GSOR, SSOR, ASSOR and PSSOR iteration methods for different grids. As we observe, for all the test examples the iteration counts of the PSSOR method is always less than that of the other methods. From the CPU time of view, this is the case for large problems. In fact, we see the the PSSOR method outperforms the other methods for large systems.



TABLE 2. Numerical results Example 3.

Method		$m = 16$	$m = 32$	$m = 64$	$m = 128$	$m = 256$	$m = 512$
PMHSS	IT	21	21	21	21	21	20
	CPU	0.02	0.03	0.08	0.36	1.94	1.48
GSOR	IT	20	22	24	26	27	27
	CPU	0.02	0.02	0.06	0.39	2.05	11.27
SSOR	IT	19	21	23	26	26	27
	CPU	0.02	0.03	0.09	0.55	3.02	16.89
ASSOR	IT	5	5	6	6	6	6
	CPU	0.01	0.02	0.09	0.16	0.91	4.82
PSSOR	IT	4	4	4	4	4	4
	CPU	0.01	0.02	0.03	0.13	0.63	3.31

TABLE 3. Numerical results Example 3.

Method		$m = 16$	$m = 32$	$m = 64$	$m = 128$	$m = 256$	$m = 512$
PMHSS	IT	34	37	38	38	38	38
	CPU	0.02	0.04	0.09	0.60	3.21	26.73
GSOR	IT	80	76	72	69	68	68
	CPU	0.03	0.04	0.16	1.04	4.85	27.01
SSOR	IT	74	74	66	66	66	66
	CPU	0.03	0.06	0.19	1.33	7.61	41.39
ASSOR	IT	7	7	7	7	7	7
	CPU	0.01	0.01	0.03	0.18	0.96	5.09
PSSOR	IT	3	3	3	3	3	3
	CPU	0.02	0.02	0.03	0.11	0.51	2.64

TABLE 4. Numerical results Example 3.

Method		$m = 16$	$m = 32$	$m = 64$	$m = 128$	$m = 256$	$m = 512$
PMHSS	IT	30	30	30	30	30	32
	CPU	0.02	0.04	0.16	1.09	6.33	34.32
GSOR	IT	7	11	20	44	71	131
	CPU	0.02	0.02	0.08	0.97	8.19	90.83
SSOR	IT	6	10	17	33	66	135
	CPU	0.02	0.02	0.10	1.12	11.86	140.55
ASSOR	IT	8	8	8	8	8	8
	CPU	0.01	0.01	0.05	0.30	1.62	9.34
PSSOR	IT	4	5	6	7	7	7
	CPU	0.02	0.02	0.05	0.27	1.48	8.46

**4. Conclusion.** In this paper we have presented a preconditioned version of the symmetric successive overrelaxation (PSSOR) iterative method to solve the equivalent real system of linear system (2), where  $W$  is symmetric positive definite and  $T$  is symmetric positive semidefinite. Convergence properties of this method have also been investigated. We have compared the numerical results of the PSSOR method with PMHSS, GSOR, SSOR and ASSOR. Numerical results show that PSSOR method is superior to the other methods in terms of both iteration number and CPU time for large problems.

**Acknowledgements.** The work of Davod Khojasteh Salkuyeh is partially supported by University of Guilan.

#### REFERENCES

- [1] O. Axelsson and A. Kucherov, Real valued iterative methods for solving complex symmetric linear systems, *Numer. Linear Algebra Appl.*, **7** (2000), 197-218.
- [2] Z.-Z. Bai, M. Benzi and F. Chen, Modified HSS iteration methods for a class of complex symmetric linear systems, *Computing*, **87** (2010), 93-111.
- [3] Z.-Z. Bai, M. Benzi and F. Chen, On preconditioned MHSS iteration methods for complex symmetric linear systems, *Numer. Algor.*, **56** (2011), 297-317.
- [4] Z.-Z. Bai, Rotated block triangular preconditioning based on PMHSS, *Sci. China Math.*, **56** (2013), 2523-2538.
- [5] Z.-Z. Bai, G.H. Golub and M.K. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, *SIAM. J. Matrix Anal. Appl.*, **24** (2003), 603-626.
- [6] Z.-Z. Bai, M. Benzi, F. Chen and Z.-Q. Wang, Preconditioned MHSS iteration methods for a class of block two-by-two linear systems with applications to distributed control problems, *IMA J. Numer. Anal.*, **33** (2013), 343-369.
- [7] Z.-Z. Bai, B.N. Parlett, Z.-Q. Wang, On generalized successive overrelaxation methods for augmented linear systems, *Numer. Math.*, **102** (2005), 1-38.
- [8] M. Benzi, D. Bertaccini, Block preconditioning of real-valued iterative algorithms for complex linear systems, *IMA J. Numer. Anal.*, **28** (2008), 598-618.
- [9] M. Benzi and G. Golub, A preconditioner for generalized saddle point problems, *SIAM J. Matrix Anal. Appl.*, **26** (2004), 20-41.
- [10] M. Benzi, G.H. Golub and J. Liesen, Numerical solution of saddle point problems, *Acta Numerica*, **14** (2005), 1-137.
- [11] A. Bunse-Gerstner and R. Stover, On a conjugate gradient-type method for solving complex symmetric linear systems, *Linear Algebra Appl.*, **287** (1999), 105-123.
- [12] M. Dehghan, M.M. Dehghani and M. Hajarian, A generalized preconditioned MHSS method for a class of complex symmetric linear systems, *J. Math. Model. Anal.*, **18** (2013), 561-576.
- [13] V. Edalatpour, D. Hezari and D.K. Salkuyeh, Two efficient inexact algorithms for a class of large sparse complex linear systems, *Mediterr. J. Math.*, **13** (2016), 2301-2318.
- [14] G.H. Golub and C.F. Van Loan, C.F., *Matrix Computations*, 3rd ed., The Johns Hopkins University Press, Baltimore, MD, (1996).
- [15] D. Hezari, D.K. Salkuyeh, V. Edalatpour, A new iterative method for solving a class of complex symmetric system of linear equations, *Numer. Algor.*, **73** (2016), 927-955
- [16] D. Hezari, V. Edalatpour and D.K. Salkuyeh, Preconditioned GSOR iterative method for a class of complex symmetric system of linear equations, *Numer. Linear Algebra Appl.*, **22** (2015), 761-776.
- [17] Z.-Z. Liang and G.-F. Zhang, On SSOR iteration method for a class of block two-by-two linear systems, *Numer. Algor.*, **71** (2016), 655-671.
- [18] D.K. Salkuyeh, D. Hezari and V. Edalatpour, Generalized successive overrelaxation iterative method for a class of complex symmetric linear system of equations, *Int. J. Comput. Math.*, **92** (2015), 802-815.
- [19] D.K. Salkuyeh and T.S. Siahkolaei, Two-parameter TSCSP method for solving complex symmetric system of linear equations, *Calcolo*, **55** (2018), 8.
- [20] G.-F. Zhang, Z. Zheng, A parameterized splitting iteration methods for complex symmetric linear systems, *Jpn. J. Indust. Appl. Math.*, **31** (2014), 265-278.

Received xxxx 20xx; revised xxxx 20xx.

*E-mail address:* [salimi.tahereh@phd.guilan.ac.ir](mailto:salimi.tahereh@phd.guilan.ac.ir)

*E-mail address:* [khojasteh@guilan.ac.ir](mailto:khojasteh@guilan.ac.ir)