A shift-splitting preconditioner for asymmetric saddle point problems*

Shi-Liang Wu, Davod Khojasteh Salkuyeh†

†School of Mathematics, Yunnan Normal University, Kunming, Yunnan, 650500, P.R. China
wushiliang1999@126.com

‡Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran
‡Center of Excellence for Mathematical Modelling, Optimization and Combinational Computing (MMOCC), University of Guilan, Rasht, Iran
khojasteh@guilan.ac.ir

Abstract

In this paper, we execute the shift-splitting preconditioner for asymmetric saddle point problems with its (1,2) block's transposition unequal to its (2,1) block under the removed minus of its (2,1) block. The proposed preconditioner is stemmed from the shift splitting (SS) iteration method for solving asymmetric saddle point problems, which is convergent under suitable conditions. The relaxed version of the shift-splitting preconditioner is obtained as well. The spectral distributions of the related preconditioned matrices are given. Numerical experiments from the Stokes problem are offered to show the convergence performance of these two preconditioners.

Keywords: Asymmetric saddle point problems; Shift-splitting preconditioner; Spectral distribution; Convergence

AMS classification: 65F10.

1 Introduction

Nowadays, a shift splitting iteration scheme has been successfully used to solve the large sparse system of linear equations

$$Ax = b \tag{1}$$

with A being non-Hermitian positive definite, which is deemed as one of the efficient stationary solvers and is first introduced in [2], and works as follows: Given an initial guess $x^{(0)}$, for $k = 0, 1, 2, \ldots$ until $\{x^{(k)}\}$ converges, compute

$$(\alpha I + A)x^{(k+1)} = (\alpha I - A)x^{(k)} + 2b, \tag{2}$$

*The research work of the first author is supported by National Natural Science Foundation of China (No.11961082). The work of the second author is supported by University of Guilan and the Center of Excellence for Mathematical Modelling, Optimization and Combinational Computing (MMOCC).

†Corresponding author: khojasteh@guilan.ac.ir

where α is a given positive constant. It is noteworthy that this shift splitting iteration scheme (2) not only is unconditionally convergent, but also can induce an economical and effective preconditioner $P = \alpha I + A$ for the non-Hermitian positive definite linear system (1). This induced precondtioner is called as the shift-splitting preconditioner. When the shift-splitting preconditioner $P = \alpha I + A$ together with Krylov subspace methods are employed to solve the non-Hermitian positive definite linear system (1), its highly efficiency has been confirmed by numerical experiments in [2].

Since both the shift splitting iteration scheme and the shift-splitting preconditioner are economical and effective, they have drawn much attention. Not only that, this approach has been successfully extended to other practical problems, such as the classical saddle point problems

$$\begin{bmatrix} A & B^T \\ -B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} p \\ q \end{bmatrix}, \tag{3}$$

where $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite (SPD), $B \in \mathbb{R}^{m\times n}$ with $\text{rank}(B) = m \leq n$, see [7]. Whereafter, Chen and Ma in [14] proposed the two-parameter shift-splitting preconditioner for saddle point problems (3). Based on the work in [14], Salkuyeh *et al.* in [30] use the two-parameter shift-splitting preconditioner for the saddle point problems (3) with symmetric positive semidefinite (2, 2)-block, and for the same problem when the symmetry of the (1,1)-block is omitted in [31], Cao *et al.* in [8] considered the saddle point problems (3) with nonsymmetric positive definite (1, 1)-block, Cao and Miao in [9] considered the singular nonsymmetric saddle point problems (3), and so on.

On the other hand, combining the shift splitting technique with the matrix splitting technique, some new efficient preconditioners have been developed, such as the modified shift-splitting preconditioner [34], the generalized modified shift-splitting preconditioner [22], the extended shift-splitting preconditioner [35], a general class of shift-splitting preconditioner [10], the modified generalized shift-splitting preconditioner [21,32], the generalized double shift-splitting preconditioner [16], and so on.

In this paper, we consider the asymmetric saddle point problems of the form

$$\mathcal{A}\mathbf{x} = \begin{bmatrix} A & B^T \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b \\ q \end{bmatrix} = \mathbf{f}, \tag{4}$$

where $A \in \mathbb{R}^{n\times n}$ is SPD, $B, C \in \mathbb{R}^{m\times n}$, $m \leq n$. Moreover, the matrices B and C are of full rank. In [11], Cao proposed the augmentation block triangular preconditioner

$$P_{Aug} = \begin{bmatrix} A + B^T W^{-1} C & B^T \\ 0 & W \end{bmatrix}, \tag{5}$$

for the system obtaining from multiplying the second block row of (4) by -1, where $W \in \mathbb{R}^{m\times n}$ is nonsingular and such that $A + B^T W^{-1} C$ is invertible. The performance of the preconditioner P_{Aug} was compared with several preconditioners presented in [12,13,26]. In [24], Li *et al.* presented the partial positive semidefinite and skew-Hermitian splitting (for short, PPSS) iteration method for the system (4). The PPSS iteration method induces the preconditioner

$$P_{PPSS} = \frac{1}{2\alpha}(\alpha I + H)(\alpha I + S), \tag{6}$$

where $\alpha > 0$,

$$H = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} \quad \text{and} \quad S = \begin{bmatrix} 0 & B^T \\ -C & 0 \end{bmatrix}.$$

Numerical results presented in [24] show that the P_{PPSS} preconditioner outperforms the classical HSS preconditioner [4]. Although the shift-splitting iteration scheme and the shift-splitting preconditioner have been successfully used to solve the classical saddle point problems, they have not been applied to the asymmetric saddle point problems (4). Based on this, our goal of this paper is to use the shift-splitting iteration scheme and the shift-splitting preconditioner for the asymmetric saddle point problems. One can see [5, 6, 15, 17, 18, 23] for more details. Theoretical analysis shows that the shift-splitting iteration method is convergent under suitable conditions and the spectral distributions of the corresponding preconditioned matrices are better clustered. Numerical experiments arising from a model Stokes problem are provided to show the effectiveness of the proposed two preconditioners.

We use the following notations throughout the paper. For a given matrix S, $\mathcal{N}(S)$ stands for the null space of S. The spectral radius of a square matrix G is denoted by $\rho(G)$. For a vector $x \in \mathbb{C}^n$, x^* is used for the conjugate transpose of x. The real and imaginary parts of any $y \in \mathbb{C}$ are denoted by $\Re(y)$ and $\Im(y)$, respectively. For two vectors x and y, the MATLAB notation $[x; y]$ is used for $[x^T, y^T]^T$. Finally, for two vectors $x, y \in \mathbb{C}^n$, the standard inner product of x and y is denoted by $\langle x, y \rangle = y^* x$.

The layout of this paper is organized as follows. In Section 2, the shift splitting iteration scheme and the related shift-splitting preconditioner are presented for the asymmetric saddle point problems (4). In Section 3, numerical experiments are provided to examine the convergence behaviors of the shift-splitting preconditioner and its relaxed version for solving the asymmetric saddle point problems (4). Finally, some conclusions are described in Section 4.

2 The shift-splitting method

Here, three lemmas are given for later discussion.

Lemma 1. [11] *The saddle point matrix*

$$\mathcal{A} = \begin{bmatrix} A & B^T \\ -C & 0 \end{bmatrix} \tag{7}$$

is nonsingular if and only if $rank(B) = rank(C) = m$, $\mathcal{N}(A) \cap \mathcal{N}(C) = \{0\}$ *and* $\mathcal{N}(A^T) \cap \mathcal{N}(B) = \{0\}$.

Lemma 2. [33] *Let λ be any root of the quadratic equation $x^2 - ax + b = 0$, where $a, b \in \mathbb{R}$. Then, $|\lambda| < 1$ if and only if $|b| < 1$ and $|a| < 1 + b$.*

Lemma 3. [3] *Let λ be any root of the quadratic equation $x^2 - \phi x + \psi = 0$, where $\phi, \psi \in \mathbb{C}$. Then, $|\lambda| < 1$ if and only if $|\psi| < 1$ and $|\phi - \phi^* \psi| + |\psi|^2 < 1$.*

First, to guarantee the unique solution of the asymmetric saddle point problems (4), Lemma 4 is obtained.

Lemma 4. *Let A be a SPD matrix and $rank(B) = rank(C) = m$. Then, saddle point matrix (7) is nonsingular.*

Proof. It is an immediate result of Lemma 1. □

Similarly, for every $\alpha > 0$ and under the conditions of Lemma 4, the matrix

$$\alpha I + \mathcal{A} = \begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix}$$

is nonsingular.

Next, under the condition of Lemma 4, we can establish the shift-splitting (SS) iteration method for solving the asymmetric saddle point problems (4). To this end, the shift-splitting of the coefficient matrix $\mathcal{A}$ in (4) can be constructed as follows

$$\begin{aligned} \mathcal{A} &= \frac{1}{2}(\alpha I + \mathcal{A}) - \frac{1}{2}(\alpha I - \mathcal{A}) \\ &= \frac{1}{2}\begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix} - \frac{1}{2}\begin{bmatrix} \alpha I - A & -B^T \\ C & \alpha I \end{bmatrix}, \end{aligned}$$

where $\alpha > 0$ and I is the identity matrix. This matrix splitting naturally leads to the shift splitting (SS) iteration method for solving the asymmetric saddle point problems (4) and works as follows.

The SS iteration method: *Let the initial vector $\mathbf{x}^{(0)} \in \mathbb{R}^{n+m}$ and $\alpha > 0$. For $k = 0, 1, 2, \ldots$ until the iteration sequence $\{\mathbf{x}^{(k)}\}_{k=0}^{+\infty}$ is converged, compute $\mathbf{x}^{(k+1)}$, by solving the linear system*

$$\begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix} \mathbf{x}^{(k+1)} = \begin{bmatrix} \alpha I - A & -B^T \\ C & \alpha I \end{bmatrix} \mathbf{x}^{(k)} + 2\begin{bmatrix} b \\ q \end{bmatrix}. \tag{8}$$

Clearly, the iteration matrix M_α of the SS method is

$$M_\alpha = \begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix}^{-1} \begin{bmatrix} \alpha I - A & -B^T \\ C & \alpha I \end{bmatrix}. \tag{9}$$

To study the convergence property of the SS method, the value of the spectral radius $\rho(M_\alpha)$ of the corresponding iteration matrix M_α is necessary to be estimated. As is known, when $\rho(M_\alpha) < 1$, the SS iteration method is convergent. Thereupon, we assume that λ is an eigenvalue of the matrix M_α and its corresponding eigenvector is $\mathbf{x} = [x; y]$. Therefore, we have

$$\begin{bmatrix} \alpha I - A & -B^T \\ C & \alpha I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

which is equivalent to

$$(\lambda - 1)\alpha x + (\lambda + 1)Ax + (\lambda + 1)B^T y = 0, \tag{10}$$

$$(1 + \lambda)Cx - \alpha(\lambda - 1)y = 0. \tag{11}$$

To obtain the convergence conditions of the SS method, the following lemmas are given.

Lemma 5. *Let the matrix A be SPD and $rank(B) = rank(C) = m$. If λ is an eigenvalue of the matrix M_α, then $\lambda \neq \pm 1$.*

Proof. If $\lambda = 1$, then based on Eqs. (10) and (11) we have

$$\begin{cases} Ax + B^T y = 0, \\ -Cx = 0. \end{cases} \tag{12}$$

Based on Lemma 4, we deduce that $x = 0$ and $y = 0$. This is a contradiction, because $\mathbf{x} = [x; y] = 0$ can not be an eigenvector of M_α. Hence, $\lambda \neq 1$.

When $\lambda = -1$, based on Eqs. (10) and (11) we have $\alpha x = 0$ and $\alpha y = 0$. Since $\alpha > 0$, we get $y = 0$ and $x = 0$, which is a contradiction, since $[x; y]$ is an eigenvector. Hence $\lambda \neq -1$. □

Based on the above discussion, the results in Lemma 6 are right.

Lemma 6. *Let the conditions of Lemma 5 be satisfied. Let also λ be an eigenvalue of M_α and $\mathbf{x} = [x; y]$ be the corresponding eigenvector. Then $x \neq 0$. Moreover, if $y = 0$, then $|\lambda| < 1$.*

Proof. When $x = 0$, from (11) we have $\alpha(\lambda - 1)y = 0$. Based on Lemma 5, $\lambda \neq 1$. Therefore, $y = 0$. This contradicts with the nonzero eigenvector $\mathbf{x} = [x; y]$. Hence $x \neq 0$.

When $y = 0$, based on Eq. (10) we get

$$(\alpha I + A)^{-1}(\alpha I - A)x = \lambda x.$$

Therefore, using the Kellogg's lemma (see [25, page 13]) we deduce

$$|\lambda| \leq \|(\alpha I + A)^{-1}(\alpha I - A)\|_2 < 1,$$

which completes the proof. □

For later use we define the set $\mathcal{S}$ as

$$\mathcal{S} = \{x \in \mathbb{C}^n : \mathbf{x} = [x; y] \text{ is an eigenvector of } M_\alpha \text{ with } \|x\|_2 = 1\}.$$

It follows from Lemma 6 that the members of $\mathcal{S}$ are nonzero.

Theorem 1. *Let the conditions of Lemma 5 be satisfied. For every $x \in \mathcal{S}$, let $a(x) = x^* A x$, $s(x) = \Re(x^H B^T C x)$ and $t(x) = \Im(x^H B^T C x)$. For each $x \in \mathcal{S}$, if $s(x) > 0$ and*

$$|t(x)| < a(x)\sqrt{s(x)}, \tag{13}$$

then

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

which implies that the SS iteration method (8) converges to the unique solution of the asymmetric saddle point problems (4).

Proof. Based on Lemma 5, from (11) we have

$$y = \frac{\lambda + 1}{\alpha(\lambda - 1)} Cx. \tag{14}$$

Substituting (14) into (10) leads to

$$(\lambda - 1)\alpha x + (\lambda + 1)Ax + \frac{(\lambda + 1)^2}{\alpha(\lambda - 1)} B^T Cx = 0. \tag{15}$$

Let $\|x\|_2 = 1$. Pre-multiplying x^* to the both sides of Eq. (15) leads to

$$\alpha^2(\lambda - 1)^2 + \alpha(\lambda^2 - 1)x^*Ax + (\lambda + 1)^2 x^* B^T Cx = 0, \tag{16}$$

which is equivalent to

$$\alpha^2(\lambda - 1)^2 + \alpha(\lambda^2 - 1)a + (\lambda + 1)^2(s(x) + t(x)i) = 0. \tag{17}$$

For the sake simplicity in notations, we use s, t and a for $s(x)$, $t(x)$ and $a(x)$, respectively. It follows from Eq. (17), that

$$\lambda^2 + \frac{2(s + ti - \alpha^2)}{\alpha^2 + \alpha a + s + ti}\lambda + \frac{\alpha^2 - \alpha a + s + ti}{\alpha^2 + \alpha a + s + ti} = 0. \tag{18}$$

Next, we will discuss two aspects: $t = 0$ and $t \neq 0$.

When $t = 0$, from (18), we get

$$\lambda^2 + \frac{2(s - \alpha^2)}{\alpha^2 + \alpha a + s}\lambda + \frac{\alpha^2 - \alpha a + s}{\alpha^2 + \alpha a + s} = 0. \tag{19}$$

By simples computations, we have

$$\left|\frac{\alpha^2 - \alpha a + s}{\alpha^2 + \alpha a + s}\right| < 1 \tag{20}$$

and

$$\left|\frac{2(s - \alpha^2)}{\alpha^2 + \alpha a + s}\right| < 1 + \frac{\alpha^2 - \alpha a + s}{\alpha^2 + \alpha a + s}. \tag{21}$$

Based on Lemma 2, the inequalities (20) and (21) imply that the roots of the real quadratic equation (19) satisfy $|\lambda| < 1$.

If $t \neq 0$, then Eq. (18) can be written as $\lambda^2 + \phi\lambda + \psi = 0$, where

$$\phi = \frac{2(s + ti - \alpha^2)}{\alpha^2 + \alpha a + s + ti} \text{ and } \psi = \frac{\alpha^2 - \alpha a + s + ti}{\alpha^2 + \alpha a + s + ti}.$$

By some calculations, we get

$$
\begin{aligned}
\phi - \phi^*\psi &= \frac{2(s+ti-\alpha^2)}{\alpha^2+\alpha a+s+ti} - \frac{2(s-ti-\alpha^2)}{\alpha^2+\alpha a+s-ti}\cdot\frac{\alpha^2-\alpha a+s+ti}{\alpha^2+\alpha a+s+ti} \\
&= \frac{2(s-\alpha^2+ti)}{\alpha^2+\alpha a+s+ti}\cdot\frac{\alpha^2+\alpha a+s-ti}{\alpha^2+\alpha a+s-ti} - \frac{2(s-\alpha^2-ti)}{\alpha^2+\alpha a+s-ti}\cdot\frac{\alpha^2-\alpha a+s+ti}{\alpha^2+\alpha a+s+ti} \\
&= 2\Big[\frac{(s-\alpha^2+ti)(\alpha^2+\alpha a+s-ti)}{(\alpha^2+\alpha a+s)^2+t^2} + \frac{(\alpha^2-s+ti)(\alpha^2-\alpha a+s+ti)}{(\alpha^2+\alpha a+s)^2+t^2}\Big] \\
&= 2\frac{(s-\alpha^2+ti)(\alpha^2+\alpha a+s-ti)+(\alpha^2-s+ti)(\alpha^2-\alpha a+s+ti)}{(\alpha^2+\alpha a+s)^2+t^2} \\
&= 4\frac{\alpha a(s-\alpha^2)+2\alpha^2 ti}{(\alpha^2+\alpha a+s)^2+t^2}.
\end{aligned}
$$

Further, we have

$$
\begin{aligned}
|\psi| &= \sqrt{\frac{(\alpha^2-\alpha a+s)^2+t^2}{(\alpha^2+\alpha a+s)^2+t^2}} < 1, \\
|\phi-\phi^*\psi| &= \frac{4\sqrt{\alpha^2 a^2(s-\alpha^2)^2+4t^2\alpha^4}}{(\alpha^2+\alpha a+s)^2+t^2}.
\end{aligned} \tag{22}
$$

Based on Lemma 3, the necessary and sufficient condition for $|\lambda| < 1$ is

$$
|\phi-\phi^*\psi| + |\psi|^2 < 1. \tag{23}
$$

Substituting (22) into (23) and solving the inequality (23) for t, gives $|t| < a\sqrt{s}$, which completes the proof. □

According to the definition of $t(x)$, we have

$$
\begin{aligned}
|t(x)| &= |\Im(x^H B^T C x)| = |\langle B^T C x, x\rangle| \\
&\le \|B^T C x\|_2 \|x\|_2 \qquad \text{(Cauchy-Schwarz inequality)} \\
&\le \|B^T C\|_2 \|x\|_2 = \|B^T C\|_2.
\end{aligned}
$$

Also we have $a(x) = x^*Ax \ge \lambda_{\min}(A)$, where $\lambda_{\min}(A)$ is the smallest eigenvalue of A. Therefore, the inequality (13) can be replaced by

$$
\|B^T C\|_2 \le \lambda_{\min}(A)\sqrt{s(x)}.
$$

In the special case that $C = kB$ with $k > 0$, we can state the following theorem.

Theorem 2. *Let the conditions of Lemma 5 be satisfied and $C = kB$ with $k > 0$. Then $\rho(M_\alpha) < 1$, $\forall \alpha > 0$, which implies that the SS iteration method (8) converges to the unique solution of the asymmetric saddle point problems (4).*

Proof. If $C = kB$ with $k > 0$, then the matrix $B^T C = kB^T B$ is symmetric positive semidefinite. Therefore, we have $s(x) = kx^*B^T Bx \ge 0$ and $t(x) = 0$. According to Theorem 1, all we need is to prove the convergence for the case that $s(x) = 0$. If $s(x) = 0$, then we get $Bx = 0$. Now, from Eq. (7) we deduce that

$$\alpha^2(\lambda - 1)^2 + \alpha(\lambda^2 - 1)a = 0,$$

which is equivalent to

$$\alpha(\lambda - 1)(\alpha^2(\lambda - 1) + \alpha(\lambda + 1)a) = 0.$$

Now, since $\alpha > 0$ and $\lambda \neq 1$ (from Lemma 5), we deduce that

$$\alpha^2(\lambda - 1) + \alpha(\lambda + 1)a = 0,$$

which gives the following equation for λ

$$\lambda = \frac{\alpha - a}{\alpha + a}.$$

Therefore, since $a = x^* Ax > 0$, we conclude that $|\lambda| < 1$, which completes the proof. □

Remark 1. *When $k = 1$, Theorem 2 is the main result in [7]. That is to say, Theorems 1 and 2 are generalizations Theorem 2.1 in [7].*

Finally, we consider the preconditioner induced by the SS iteration method (8). As is known, the advantage of matrix splitting technique often is twofold: one is to result in a splitting iteration method and the other is to induce a splitting preconditioner for improving the convergence speed of Krylov subspace methods in [2]. Based on the SS iteration method (8), the corresponding shift-splitting preconditioner can be defined by

$$P_{SS} = \frac{1}{2} \begin{bmatrix} \alpha I + A & B^T \\ -C & \alpha I \end{bmatrix}.$$

Since the multiplicative factor $\frac{1}{2}$ in the preconditioner P_{SS} has no effect and can be removed when P_{SS} is used as a preconditioner, in the implementations, we only consider the shift-splitting preconditioner P_{SS} without the multiplicative factor $\frac{1}{2}$. In this case, using P_{SS} with Krylov subspace methods (such as GMRES, or its restarted version GMRES(k)), a vector of the form

$$z = P_{SS}^{-1} r$$

needs to be computed.

Let $z = [z_1; z_2]$ and $r = [r_1; r_2]$. Then $z = P_{SS}^{-1} r$ is equal to

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ \frac{1}{\alpha} C & I \end{bmatrix} \begin{bmatrix} \alpha I + A + \frac{1}{\alpha} B^T C & 0 \\ 0 & \alpha I \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\alpha} B^T \\ 0 & I \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}. \tag{24}$$

Based on Eq. (24), the following algorithm can be used to obtain the vector z.

Algorithm 1. Let $z = [z_1; z_2]$ and $r = [r_1; r_2]$. Compute z by the following procedure
1. Compute $t = r_1 - \frac{1}{\alpha} B^T r_2$;
2. Solve $(\alpha I + A + \frac{1}{\alpha} B^T C) z_1 = t$ for z_1;
3. Compute $z_2 = \frac{1}{\alpha}(C z_1 + r_2)$.

In Step 2 of Alg. 1, in general the matrix $\alpha I + A + \frac{1}{\alpha}B^T C$ is indefinite, hence the corresponding system can be solved exactly using the LU factorization or inexactly using a Krylov subspace method like GMRES or its restarted version. However, when $C = kB$ with $k > 0$, this matrix is of the form $\alpha I + A + \frac{k}{\alpha}B^T B$ which is SPD. Therefore, the corresponding system can be solved exactly using the Cholesky factorization or inexactly using the conjugate gradient (CG) method.

In general the matrix $\alpha I + A + \frac{1}{\alpha}B^T C$ is dense (because of the term $B^T C$) and solving the corresponding system by a direct method may be impractical. Hence, it is recommended to solve the system by an iteration method, as we will shortly do in the section of the numerical experiments. From theoretical point of view, when $\alpha = 0$ the preconditioner $P_{SS} = \alpha I + \mathcal{A}$ coincides with the coefficient matrix of original system. In this case, implementation of the preconditioner would be as difficult as solving the original system. Hence, it is better to choose a small value of α to obtain a more well-conditioned matrix. Since the condition of the matrix $\alpha I + A + \frac{1}{\alpha}B^T C$ strongly depends on the term $\frac{1}{\alpha}B^T C$, similar to [8,19] we choose the parameter α equals to

$$\alpha_{est} = \frac{\|B^T C\|_2}{\|A\|_2},$$

which balances the matrices A and $B^T C$.

When Krylov subspace methods together with the preconditioner $P_{SS} = \alpha I + \mathcal{A}$ are applied to solve the asymmetric saddle point problems (4), we need to establish the spectral distribution of the preconditioned matrix $P_{SS}^{-1}A$ to investigate the convergence performance of the preconditioner P_{SS} for Krylov subspace methods.

The following theorem on the spectral distribution of the preconditioned matrix $P_{SS}^{-1}\mathcal{A}$ can be obtained.

Theorem 3. *Let the conditions of Theorem 1 or 2 be satisfied. Then the preconditioned matrix $P_{SS}^{-1}\mathcal{A}$ are positive stable for $\alpha > 0$ and its the eigenvalues satisfy $|\lambda| < 1$, where λ denotes the eigenvalue of the preconditioned matrix $P_{SS}^{-1}\mathcal{A}$.*

Proof. It follows from

$$2P_{SS}^{-1}\mathcal{A} = I - M_\alpha,$$

that for each $\mu \in \sigma(M_\alpha)$, there is a $\lambda \in \sigma(P_{SS}^{-1}\mathcal{A})$, such that $2\lambda = 1 - \mu$. Therefore, we

$$\frac{\mu}{2} = \frac{1}{2} - \lambda = \frac{1}{2} - \Re(\lambda) - i\Im(\lambda).$$

Hence, from the fact that $|\mu| < 1$ we conclude

$$(\frac{1}{2} - \Re(\lambda))^2 + (\Im(\lambda))^2 < \frac{1}{4},$$

which shows that the eigenvalues of the preconditioned matrix $P_{SS}^{-1}\mathcal{A}$ are contained in a circle with radius $\frac{1}{2}$ centered at $(\frac{1}{2}, 0)$. Hence, the real parts of the eigenvalues of the matrix $P_{SS}^{-1}\mathcal{A}$ are all positive. This means that the matrix $P_{SS}^{-1}\mathcal{A}$ is positive stable for $\alpha > 0$. On the other hand, from $|\mu| < 1$ we deduce that

$$2|\lambda| = |1 - \mu| \leq 1 + |\mu| < 2,$$

which completes the proof. □

Here, we present a relaxed version of the shift-splitting preconditioner as well, which is defined by

$$P_{RSS} = \begin{bmatrix} A & B^T \\ -C & \alpha I \end{bmatrix}.$$

Similarly, using P_{RSS} with Krylov subspace methods (such as GMRES, or its restarted version GMRES(k)), a vector of the form

$$z = P_{RSS}^{-1} r$$

has to be computed as well. Let $z = [z_1; z_2]$ and $r = [r_1; r_2]$. Then, we have

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} I & 0 \\ \frac{1}{\alpha}C & I \end{bmatrix} \begin{bmatrix} A + \frac{1}{\alpha}B^T C & 0 \\ 0 & \alpha I \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\alpha}B^T \\ 0 & I \end{bmatrix} \begin{bmatrix} r_1 \\ r_2 \end{bmatrix}. \tag{25}$$

Based on Alg. 1, by a simple modification, we obtain Alg. 2 to obtain the vector z as follows.

Algorithm 2. Let $z = [z_1; z_2]$ and $r = [r_1; r_2]$. Compute z by the following procedure
1. Compute $t = r_1 - \frac{1}{\alpha}B^T r_2$;
2. Solve $(A + \frac{1}{\alpha}B^T C)z_1 = t$;
3. Compute $z_2 = \frac{1}{\alpha}(Cz_1 + r_2)$.

In the same way, we can obtain the spectral distribution of the preconditioned matrix $P_{RSS}^{-1}\mathcal{A}$, as follows.

Theorem 4. *Let the conditions of Theorem 1 be satisfied. Then the preconditioned matrix $P_{RSS}^{-1}\mathcal{A}$ has an eigenvalue 1 with algebraic multiplicity n and the remaining eigenvalues are the eigenvalues of matrix $\frac{1}{\alpha}C(A + \frac{1}{\alpha}B^T C)^{-1}B^T$.*

Proof. By calculation, we get

$$\begin{aligned} P_{RSS}^{-1}\mathcal{A} &= \begin{bmatrix} I & 0 \\ \frac{1}{\alpha}C & I \end{bmatrix} \begin{bmatrix} A + \frac{1}{\alpha}B^T C & 0 \\ 0 & \alpha I \end{bmatrix}^{-1} \begin{bmatrix} I & -\frac{1}{\alpha}B^T \\ 0 & I \end{bmatrix} \begin{bmatrix} A & B^T \\ -C & 0 \end{bmatrix} \\ &= \begin{bmatrix} (A + \frac{1}{\alpha}B^T C)^{-1} & 0 \\ \frac{1}{\alpha}C(A + \frac{1}{\alpha}B^T C)^{-1} & \frac{1}{\alpha}I \end{bmatrix} \begin{bmatrix} I & -\frac{1}{\alpha}B^T \\ 0 & I \end{bmatrix} \begin{bmatrix} A & B^T \\ -C & 0 \end{bmatrix} \\ &= \begin{bmatrix} (A + \frac{1}{\alpha}B^T C)^{-1} & -\frac{1}{\alpha}(A + \frac{1}{\alpha}B^T C)^{-1}B^T \\ \frac{1}{\alpha}C(A + \frac{1}{\alpha}B^T C)^{-1} & -\frac{1}{\alpha^2}C(A + \frac{1}{\alpha}B^T C)^{-1}B^T + \frac{1}{\alpha}I \end{bmatrix} \begin{bmatrix} A & B^T \\ -C & 0 \end{bmatrix} \\ &= \begin{bmatrix} I & (A + \frac{1}{\alpha}B^T C)^{-1}B^T \\ 0 & \frac{1}{\alpha}C(A + \frac{1}{\alpha}B^T C)^{-1}B^T \end{bmatrix}. \end{aligned}$$

Therefore, the proof is completed. □

Obviously, for each $\alpha > 0$ the preconditioner P_{RSS} is more closer than the preconditioner P_{SS} to the original matrix $\mathcal{A}$. However, the subsystem appeared in the implementation of the P_{SS} preconditioner in a Krylov subspace method is more well-conditioned than that of P_{RSS}. Hence, it is recommend to apply the preconditioner P_{SS} when the subsystems are solved inexactly using an iteration method and the P_{RSS} when the subsystems are solved exactly using direct method.

3 Numerical experiments

In this section, we present some numerical experiments to demonstrate the performance of the shift-splitting preconditioner. In the meantime, the numerical comparison are provided to show the advantage of the shift-splitting preconditioner (P_{SS}) and its relaxed version (P_{RSS}) over the PPSS preconditioner given by Eq. (6) (denoted by P_{PPSS}) and the augmentation block triangular preconditioner given by Eq. (5) (denoted by P_{Aug}). In our computations, we apply the flexible GMRES (FGMRES) [27, 29] together with these four preconditioners to solve the assymmetric saddle point systems (4) and adjust the right-hand side $\mathbf{f}$ such that the exact solution is a vector of all ones. The iterations start with a zero vector as an initial guess and are stopped when the numbers of iteration exceeds 1000 or

$$R_k = \frac{\|\mathbf{f} - \mathcal{A}\mathbf{x}^{(k)}\|_2}{\|\mathbf{f}\|_2} \leq 10^{-7},$$

where $\mathbf{x}^{(k)}$ is the computed solution at iteration k. In the implementation of the preconditioners the subsystems are solved inexactly using the iterations method. When the coefficient matrix is SPD, the corresponding system is solved using the conjugate gradient (CG) method, otherwise by the restarted version of GMRES(10). For the subsystems, the iteration is stopped as soon as the residual 2-norm is reduced by a factor of 10^2 and the maximum number of iterations is set to be 100. Similar to the outer iterations, a null vector is used as an initial guess. Finally, for the augmentation block triangular preconditioner the matrix W is set to be $W = \alpha I$ with $\alpha > 0$. In this case, the preconditioner P_{Aug} takes the following form

$$P_{Aug} = \begin{bmatrix} A + \frac{1}{\alpha}B^T C & B^T \\ 0 & \alpha I \end{bmatrix},$$

For all the methods the optimal value of parameter are obtained experimentally (denoted by α_*) and are the ones resulting in the least numbers of iterations. We also report the numerical results for the parameter $\alpha_{est} = \|B^T C\|_2 / \|A\|_2$.

We present the numerical results in the tables. In the tables, "CPU" and "Iters" stand for the elapsed CPU time (in second) and the number of iterations for the convergence. A dagger (†) means that the iteration has not converged in 1000 iterations. All runs are implemented in MATLAB R2017, equipped with a Laptop with 1.80 GHz central processing unit (Intel(R) Core(TM) i7-4500), 6 GB memory and Windows 7 operating system.

Example 1. Let the asymmetric saddle point problems (4) be given by

$$A = \begin{bmatrix} I \otimes T + T \otimes I & 0 \\ 0 & I \otimes T + T \otimes I \end{bmatrix} \in \mathbb{R}^{2s^2 \times 2s^2}$$

and

$$B^T = \begin{bmatrix} I \otimes F \\ F \otimes I \end{bmatrix} \in \mathbb{R}^{2s^2 \times s^2}, \ C = kB,$$

with

$$T = \frac{\mu}{h^2}\text{tridiag}(-1, 2, -1) \in \mathbb{R}^{s \times s}, \quad F = \frac{1}{h}\text{tridiag}(-1, 1, 0) \in \mathbb{R}^{s \times s}, k > 0.$$

Table 1: Matrix properties for Example 1.

s	n	m	$nnz(A)$	$nnz(B)$	$nnz(C)$
16	512	256	2432	992	992
32	2048	1024	9984	4032	4032
64	8192	4096	40448	16256	16256
128	32768	16384	162816	65280	65280
256	131072	65536	653312	261632	261632

where $\otimes$ denotes the Kronecker product and $h = 1/(s+1)$ is the discretization mesh-size. Therefore, the total number of variables $n = 3s^2$.

This asymmetric saddle point problems (4) can be obtained by using the upwind scheme to discretize the Stokes problem in the region $\Omega = (0,1) \times (0,1) \subset \mathbb{R}^2$ with its boundary being $\partial\Omega$: find u and p such that

$$\begin{cases} -\mu\Delta u + \nabla p = f, & \text{in } \Omega, \\ \nabla \cdot u = g, & \text{in } \Omega, \\ u = 0, & \text{on } \partial\Omega, \\ \int_\Omega p(x)dx = 0, \end{cases}$$

where μ, Δ, u and p are the viscosity scalar, the componentwise Laplace operator, a vector-valued function representing the velocity, and a scalar function representing the pressure, respectively.

We set $s = 16, 32, 64, 128, 256$ and $k = 2$. Generic properties of the test matrices are presented in Table 1. In this table, $nnz(.)$ stands for number of nonzero entries of the matrix. Numerical results for $\mu = 1$ and $\mu = 0.1$ are presented in the Tables 2 and 3, respectively. From the numerical results in Tables 2-3, it is easy to find that the computational efficiency of GMRES can not be satisfy when it is directly used to solve the asymmetric saddle point problems (4). Whereas, FGMRES together with these four preconditioners for solving the asymmetric saddle point problems (4) can rapidly converge. This also confirms that all four preconditioners indeed can improve the convergence speed of GMRES. Among the preconditioners, P_{SS} and P_{RSS} outperform the others from the iteration steps and the CPU time point of review. On the other hand we observe the parameter α_{est} often gives suitable results, especially for large problems.

In the sequel, we investigate the spectral distribution of four preconditioned matrices $P_{SS}^{-1}\mathcal{A}$, $P_{RSS}^{-1}\mathcal{A}$, $P_{PPSS}^{-1}\mathcal{A}$ and $P_{Aug}^{-1}\mathcal{A}$. To do so, we set $s = 16$ and use the optimal value of the parameters given in Tables 2 and 3. Figs. 1-2 plot the spectral distribution of the matrices. Fig. 1 plots the spectral distribution of five matrices $\mathcal{A}$, $P_{SS}^{-1}\mathcal{A}$, $P_{RSS}^{-1}\mathcal{A}$ $P_{PPSS}^{-1}\mathcal{A}$ and $P_{Aug}^{-1}\mathcal{A}$ with $\mu = 1$ and Fig. 2 for $\mu = 0.1$. From the spectral distribution in Figs. 1-2, four preconditioners P_{SS}, P_{RSS}, P_{PPSS} and P_{Aug} improve the spectral distribution of the original coefficient matrix $\mathcal{A}$. As we observe, the eigenvalues of $P_{SS}^{-1}\mathcal{A}$ are $P_{RSS}^{-1}\mathcal{A}$ better clustered than the two other preconditioned matrices. Moreover, the spectral distribution of $P_{SS}^{-1}\mathcal{A}$ and $P_{RSS}^{-1}\mathcal{A}$ are almost in line with the theoretical results, see Theorem 3 and Theorem 4.

Table 2: Numerical results of FGMRES for Example 1 with $\mu = 1$.

s		No Prec.	P_{SS}	P_{RSS}	P_{PPSS}	P_{Aug}		P_{SS}	P_{RSS}
16	α_*	–	0.10	0.20	98.50	0.11	α_{est}	2.03	2.03
	Iters	133	8	8	38	21	Iters	12	11
	CPU	0.13	0.03	0.02	0.05	0.06	CPU	0.03	0.03
	R_k	8.1e-8	8.4e-8	5.9e-8	1.0e-7	7.5e-8	R_k	6.6e-8	9.3e-8
32	α_*	–	0.20	0.34	100.60	0.10	α_{est}	2.01	2.01
	Iters	285	9	9	45	21	Iters	13	12
	CPU	2.93	0.06	0.05	0.14	0.14	CPU	0.06	0.06
	R_k	9.6e-8	2.4e-8	9.6e-8	1.0e-7	8.9e-8	R_k	5.2e-8	7.4e-8
64	α_*	–	0.60	1.50	102.20	0.37	α_{est}	2.01	2.01
	Iters	617	12	12	63	29	Iters	14	13
	CPU	36.20	0.39	0.3	0.89	0.74	CPU	0.38	0.36
	R_k	9.6e-8	7.5e-8	8.2e-8	9.3e-8	8.2e-8	R_k	5.6e-8	6.4e-8
128	α_*	–	0.60	0.64	103.90	4.20	α_{est}	2.02	2.02
	Iters	†	22	23	111	31	Iters	24	23
	CPU	–	2.37	2.48	8.69	3.19	CPU	2.55	2.33
	R_k	–	8.4e-8	8.5e-8	8.8e-8	6.5e-8	R_k	5.2e-8	5.4e-8
256	α_*	–	1.39	1.39	102.00	22.00	α_{est}	2.02	2.02
	Iters	†	57	52	217	78	Iters	64	54
	CPU	–	34.89	32.38	175.49	47.18	CPU	40.66	33.84
	R_k	–	9.5e-8	8.1e-8	9.9e-8	7.1e-8	R_k	9.5e-8	4.2e-8

Table 3: Numerical results of FGMRES for Example 1 with $\mu = 0.1$.

s		No Prec.	P_{SS}	P_{RSS}	P_{PPSS}	P_{Aug}		P_{SS}	P_{RSS}
16	α_*	–	0.25	0.25	15.40	0.53	α_{est}	18.34	18.34
	Iters	117	8	8	36	17	Iters	28	12
	CPU	0.16	0.02	0.02	0.04	0.04	CPU	0.02	0.02
	R_k	8.9e-8	1.5e-8	1.4e-8	9.4e-8	8.6e-8	R_k	8.2e-8	6.6e-8
32	α_*	–	0.23	0.23	29.80	2.42	α_{est}	19.45	19.45
	Iters	238	11	11	56	20	Iters	31	13
	CPU	1.67	0.07	0.07	0.16	0.11	CPU	0.08	0.06
	R_k	9.0e-8	9.0e-8	5.6e-8	9.2e-8	1.0e-7	R_k	6.9e-8	5.3e-8
64	α_*	–	1.50	2.1	53.20	4.60	α_{est}	19.87	19.87
	Iters	483	11	11	86	26	Iters	32	14
	CPU	22.48	0.26	0.25	0.95	0.65	CPU	0.46	0.38
	R_k	9.9e-8	9.7e-8	5.8e-8	9.6e-8	9.4e-8	R_k	8.8e-8	4.2e-8
128	α_*	–	4.90	6.4	92.80	19.10	α_{est}	19.98	19.98
	Iters	908	18	19	129	39	Iters	33	20
	CPU	302.64	1.91	1.96	7.07	4.06	CPU	3.07	2.17
	R_k	9.9e-8	9.2e-8	7.2e-8	9.9e-8	9.9e-8	R_k	7.4e-8	9.2e-8
256	α_*	–	10.90	12.96	131.00	25.90	α_{est}	20.05	20.05
	Iters	†	30	37	192	90	Iters	37	46
	CPU	–	26.03	22.73	151.26	55.38	CPU	23.26	29.10
	R_k	–	9.0e-8	9.6e-8	9.7e-8	7.8e-8	R_k	6.2e-8	9.1e-8

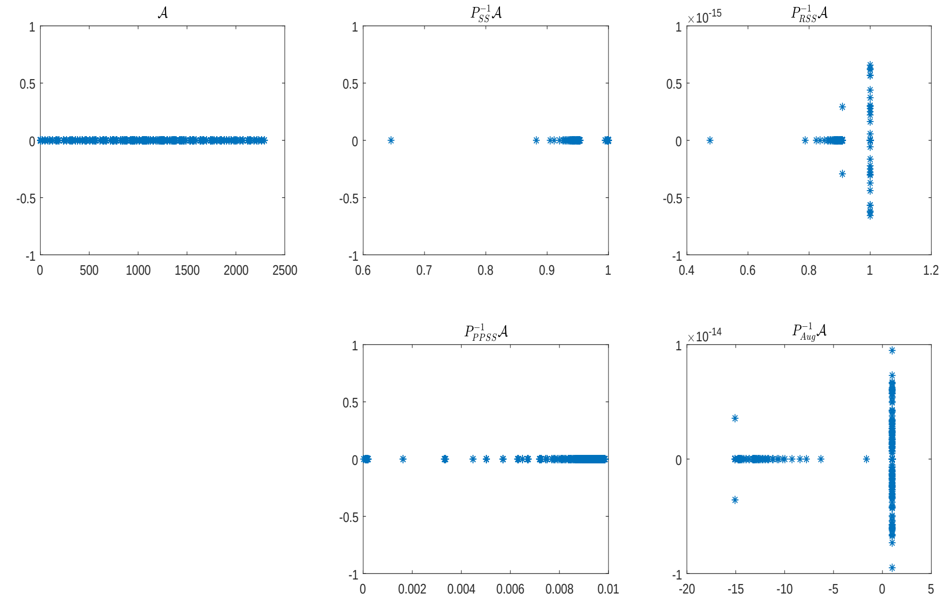

Figure 1: Spectra distribution of Example 1 for $s = 16$ with $\mu = 1$ and $k = 2$.

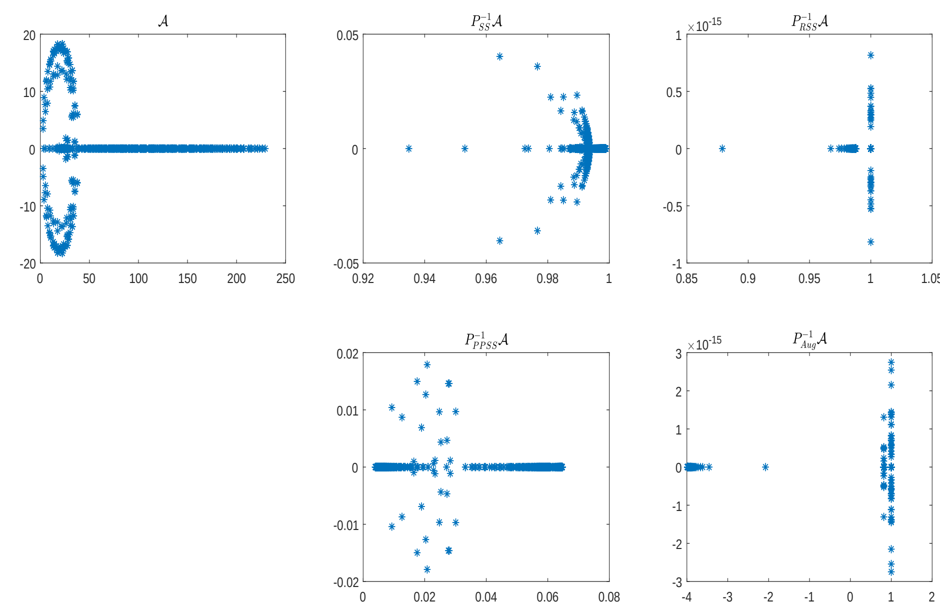

Figure 2: Spectra distribution of Example 1 for $s = 16$ with $\mu = 0.1$ and $k = 2$.

Table 4: Matrix properties for Example 2.

Matrix	n	m	$nnz(A)$	$nnz(B)$	$nnz(C)$
Szczerba/Ill_Stokes	15672	5224	73650	58242	59476

Table 5: Numerical results for Example 2 for different values of α.

	P_{SS}			P_{RSS}			P_{PPSS}			P_{Aug}		
α	Iters	CPU	R_k	Iters	CPU	R_k	Iters	CPU	R_k	Iters	CPU	R_k
0.1	471	31.32	9.8e-8	179	15.11	9.7e-8	467	22.97	1.0e-7	188	8.31	9.9e-8
0.05	358	22.14	9.9e-8	169	13.57	9.8e-8	348	13.65	9.7e-8	180	7.94	9.6e-8
0.01	210	13.56	9.7e-8	145	11.44	9.7e-8	164	5.19	9.1e-8	171	7.23	9.9e-8
0.005	173	11.40	9.6e-8	134	10.12	9.9e-8	121	**4.44**	9.2e-8	175	**7.19**	9.9e-8
0.001	115	7.49	9.9e-8	110	7.21	9.5e-8	66	6.97	9.1e-8	193	7.61	9.8e-8
0.0005	99	5.84	9.8e-8	97	5.93	9.8e-8	66	15.23	9.0e-8	208	8.20	9.7e-8
0.0001	64	**3.91**	9.7e-8	63	**3.95**	1.0e-7	84	123.13	9.4e-8	311	16.30	1.0e-7
0.00005	62	4.26	9.6e-8	61	4.23	1.0e-7	92	168.02	9.3e-8	313	18.61	9.9e-8

Table 6: Numerical results for Example 2 for α_{est}.

P_{SS}				P_{RSS}			
α_{est}	Iters	CPU	R_k	α_{est}	Iters	CPU	R_k
0.000169	74	4.57	9.5e-8	0.000169	73	4.24	9.9e-8

Example 2. We use the matrix **Szczerba/Ill_Stokes** from the UF Sparse Matrix Collection[1], which is an ill-conditioned matrix arisen from computational fluid dynamics problems. Generic properties of the test matrix are given in Table 4. The FGMRES (GMRES) method without preconditioning fails to converge in 1000 iterations. So, we present the numerical results of the FGMRES method with the preconditioners P_{SS}, P_{RSS}, P_{PPSS} and P_{Aug} for different values of the parameter α in Table 5. As we observe all the preconditioners reduce the number of iterations of the GMRES method. The minimum value of the CPU time for each of the preconditioner have been underlined. As we see the minimum value of the CPU time is due to the P_{SS} preconditioner. Numerical results of the preconditioners P_{SS} and P_{RSS} have been presented in Table 6. As we there is a good agreement between the results of the P_{SS} and P_{RSS} preconditioners with α_* and those of with α_{est}.

4 Conclusion

For the asymmetric saddle point problems, we have presented the shift-splitting preconditioner and its relaxed version to improve the convergence speed of Krylov subspace method (such as GMRES/FGMRES). The eigenvalue distribution of the related preconditioned matrices have been provided. Moreover, we have proved that the shift-splitting iteration method for the asymmetric saddle point problems is convergent under suitable conditions. Numerical experiments

[1] https://www.cise.ufl.edu/research/sparse/matrices/Szczerba/Ill_Stokes.html

from the Stokes problem are given to verify the efficiency of the shift-splitting preconditioner and its relaxed version.

Acknowledgment

The authors would like to thank the anonymous referee for helpful comments and suggestions.

References

[1] Z.-Z. Bai, G.H. Golub, M.K. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, SIAM J. Matrix Anal. Appl., 24 (2003) 603-626.

[2] Z.-Z. Bai, J.-F. Yin, Y.-F. Su, A shift-splitting preconditioner for non-Hermitian positive definite matrices, J. Comput. Math., 24 (2006) 539-552.

[3] Z.-Z. Bai, Z.-Q. Wang, On parameterized inexact Uzawa methods for generalized saddle point problems, Linear Algebra Appl., 428 (2008) 2900-2932.

[4] M. Benzi, G.H. Golub, A Preconditioner for Generalized Saddle Point Problems, SIAM J. Matrix Anal. Appl. 26 (2004) 20-41.

[5] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numerica., 14 (2005) 1-137.

[6] S. Cafieri, M. D′Apuzzo, V. De Simone, D. Di Serafino, On the iterative solution of KKT systems in potential reduction software for large-scale quadratic probems, Comput. Optim. Appl., 38 (2007) 27-45.

[7] Y. Cao, J. Du, Q. Niu, Shift-splitting preconditioners for saddle point problems, J. Comput. Appl. Math., 272 (2014) 239-250.

[8] Y. Cao, S. Li, L.-Q. Yao, A class of generzalized shift-splitting preconditioners for nonsymmeric saddle point problems, Appl. Math. Lett., 49 (2015) 20-27.

[9] Y. Cao , S.-X. Miao, On semi-convergence of the generalized shift-splitting iteration method for singular nonsymmetric saddle point problems, Comput. Math. Appl., 71 (2016) 1503-1511.

[10] Y. Cao, A general class of shift-splitting preconditioners for non-Hermitian saddle point problems with applications to time-harmonic eddy current models, Comput. Math. Appl., 77 (2019) 1124-1143.

[11] Z.-H. Cao, Augmentation block preconditioners for saddle point-type matrices with singular (1,1) blocks, Numer. Linear Algebra Appl., 15 (2008) 515–533.

[12] Z.-H Cao, A note on block diagonal and constraint preconditioners for non-symmetric indefinite linear systems, Inter. J. Comput. Math 83 (2006) 383395.

[13] Z.-H. Cao, Block triangular Schur complement preconditioners for saddle point problems and application to the Oseen equations, Appl. Numer. Math. 3 (2010) 193-207.

[14] C. Chen, C.-F. Ma, A generalized shift-splitting preconditioner for saddle point problems, Appl. Math. Lett., 43 (2015) 49-55.

[15] H.C. Elman, Preconditioning for the steady-state Navier-Stokes equations with low viscosity, SIAM J. Sci. Comput., 20 (1999) 1299-1316.

[16] H.-T. Fan, X.-Y. Zhu, B. Zheng, The generalized double shift-splitting preconditioner for nonsymmetric generalized saddle point problems from the steady Navier-Stokes equations, Comp. Appl. Math., 37 (2018) 3256-3266.

[17] C. Greif, D. Schötzau, Preconditioners for the discretized time-harmonic Maxwell equaitons in mixed form, Numer. Linear Algebra Appl., 14 (2007) 281-297

[18] C. Greif, D. Schötzau, Preconditioners for saddle point linear systems with highly singular (1,1) blocks, ETNA, 22 (2006) 114-121.

[19] G.H. Golub, C. Greif, On solving block-structured indefinite linear systems, SIAM J. Sci. Comput., 24 (2003) 2076–2092.

[20] R.A. Horn, C.R. Johnson, Matrix analysis. Cambridge University Press, 1986.

[21] Z.-G. Huang, L.-G. Wang, Z. Xu, J.-J. Cui, A modified generalized shift-splitting preconditioner for nonsymmetric saddle point problems, Numer. Algor., 78 (2018) 297-331.

[22] Z.-G. Huang, L.-G. Wang, Z. Xu, J.-J. Cui, The generalized modified shift-splitting preconditioners for nonsymmetric saddle point problems, Appl. Math. Comput., 299 (2017) 95-118.

[23] T. Rees, C. Greif, A preconditioner for linear systems arising from interior point optimization methods, SIAM J. Sci. Comput., 29 (2007) 1992-2007

[24] J.-L. Li, T.-Z. Huang, L. Li, The spectral properties of the preconditioned matrix for nonsymmetric saddle point problems, J. Comput. Appl. Math., 235 (2010) 270–285.

[25] G.I. Marchuk, Methods of Numerical Mathematics, Springer-Verlag, New York, 1984.

[26] M.F. Murphy, G.H. Golub, A.J Wathen, A note on preconditioning for indefinite linear systems, SIAM J. Sci. Comput. 21 (2000) 1969–1972.

[27] Y. Saad. A flexible inner-outer preconditioned GMRES algorithm, SIAM J. Sci. Comput., 14 (1993) 461–469.

[28] Y. Saad, M. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 7 (1986) 856–869.

[29] Y. Saad, Iterative Methods for Sparse Linear Systems, Second edition, SIAM, Philadelphia, PA, 2003.

[30] D.K. Salkuyeh, M. Masoudi, D. Hezari, On the generalized shift-splitting preconditioner for saddle point problems, Appl. Math. Lett., 48 (2015) 55-61.

[31] D.K. Salkuyeh, M. Masoud, D. Hezari, A preconditioner based on the shift-splitting method for generalized saddle point problems, in: The 46th Annual Iranian Mathematics Conference, Yazd University, Yazd, 25–28 August 2015.

[32] D.K. Salkuyeh, M. Rahimian, A modification of the generalized shift-splitting method for singular saddle point problems, Comput. Math. Appl., 74 (2017) 2940-2949.

[33] S.-L. Wu, T.-Z. Huang, X.-L. Zhao, A modified SSOR iterative method for augmented systems. J. Comput. Appl. Math., 228 (2009) 424-433.

[34] S.-W. Zhou, A.-L. Yang, Y. Dou, Y.-J. Wu, The modified shift-splitting preconditioners for nonsymmetric saddle-point problems, Appl. Math. Lett., 59 (2016) 109-114.

[35] Q.-Q. Zheng, L.-Z. Lu, Extended shift-splitting preconditioners for saddle point pr oblems, J. Comput. Appl. Math., 313 (2017) 70-81.