

A Local Positive (Semi)Definite Shift-Splitting Preconditioner for Saddle Point Problems with Applications to Time-Harmonic Eddy Current Models

Yang Cao¹ and Zhi-Ru Ren^{2,*}

¹*School of Transportation and Civil Engineering, Nantong University, Nantong 226019, P.R. China.*

²*School of Statistics and Mathematics, Central University of Finance and Economics, Beijing 100081, P.R. China.*

Received 15 March 2019; Accepted (in revised version) 20 June 2019.

Abstract. A local positive (semi)definite shift-splitting preconditioner for non-Hermitian saddle point problems arising in finite element discretisations of hybrid formulations of time-harmonic eddy current models is constructed. The convergence of the corresponding iteration methods is proved and the spectral properties of the associated preconditioned saddle point matrices are studied. Numerical experiments show the efficiency of the proposed preconditioner for Krylov subspace methods.

AMS subject classifications: 65F10, 65F50

Key words: Saddle point problem, splitting iteration, preconditioning, convergence, time-harmonic eddy current model.

1. Introduction

Let $A \in \mathbb{C}^{n \times n}$ be a non-Hermitian positive (semi)definite matrix — i.e. $H = (1/2)(A+A^*)$ is positive (semi)definite, and $B \in \mathbb{C}^{m \times n}$ with $m \leq n$ be a full rank matrix. We consider iterative solutions of the following large sparse saddle point problem:

$$\mathcal{A}w \equiv \begin{bmatrix} A & B^* \\ -B & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} f \\ g \end{bmatrix} \equiv p, \quad (1.1)$$

where B^* denotes the conjugate transpose of B and $f \in \mathbb{C}^n$, $g \in \mathbb{C}^m$ are given vectors. It is well-known that if A and B are, respectively, positive definite and full rank matrices or if

$$\text{null}(A) \cap \text{null}(B) = \{0\}, \quad \text{null}(B^*) = \{0\},$$

*Corresponding author. *Email address:* renzr@cufe.edu.cn (Z.-R. Ren)

then the non-Hermitian saddle point matrix \mathcal{A} is nonsingular [6, 17]. The saddle point problem (1.1) appears in various applications, including computational fluid dynamics [30], constrained and weighted least squares problems [34], electromagnetism [19], time-harmonic eddy current models [5, 36, 38, 39], geomechanics [18], and meshfree discretisation of elasticity problems [15, 27]. The reader can also consult [17] and the references therein for more information about the problem.

Recently, a lot of efforts have been spent on iteration methods for the problem (1.1). The list of the methods studied includes classical Uzawa iteration method [2] and its generalisations [13, 20, 28, 31], Hermitian and skew-Hermitian splitting (HSS) iteration methods [11] and its variants [9, 10, 26, 33, 45], shift-splitting iteration methods [1, 14, 22–24, 29, 41, 42], residual reduction algorithms [3] and Krylov subspace iteration methods [40]. If \mathcal{A} is a non-Hermitian and/or ill-conditioned matrix, the preconditioning is often used to accelerate the convergence. For example, block diagonal and block triangular preconditioners are considered in [4, 12], HSS preconditioners in [16, 21, 25] and shift-splitting preconditioners in [23, 43] — cf. also [17, 35] and references therein. These iteration methods and preconditioners often depend on the problem studied and have to be adjusted with respect to the corresponding coefficient matrices.

In this work, we focus on a class of non-Hermitian saddle point problems arising in the finite element discretisations of hybrid formulations of time-harmonic eddy current models [38, 39]. A model often used to simulate the electromagnetic phenomena of alternating currents at low frequencies can be described by the equations

$$\begin{aligned}
\mathbf{curl}(\sigma^{-1}\mathbf{curl}\mathbf{H}_C) + i\omega\mu\mathbf{H}_C &= \mathbf{curl}(\sigma^{-1}\mathbf{J}_{e,C}) && \text{in } \Omega_C, \\
\mathbf{curl}(\mu^{-1}\mathbf{curl}\mathbf{E}_I) &= -i\omega\mathbf{J}_{e,I} && \text{in } \Omega_I, \\
\operatorname{div}(\epsilon\mathbf{E}_I) &= 0 && \text{in } \Omega_I, \\
\mu^{-1}\mathbf{curl}\mathbf{E}_I \times \mathbf{n} &= 0 && \text{on } \partial\Omega, \\
\epsilon\mathbf{E}_I \cdot \mathbf{n} &= 0 && \text{on } \partial\Omega, \\
\mathbf{H}_C \times \mathbf{n} &= (-i\omega\mu)^{-1}\mathbf{curl}\mathbf{E}_I \times \mathbf{n} && \text{on } \Gamma, \\
\mathbf{E}_I \times \mathbf{n} &= \sigma^{-1}(\mathbf{curl}\mathbf{H}_C - \mathbf{J}_{e,C}) \times \mathbf{n} && \text{on } \Gamma,
\end{aligned} \tag{1.2}$$

where \mathbf{E} , \mathbf{H} , \mathbf{J}_e , μ , σ , ω and i , respectively, denote electric field, magnetic field, generator current, magnetic permeability, electric conductivity, a nonzero angular frequency and the imaginary unit. The computational domain $\Omega \subset \mathbb{R}^3$ is a simply connected Lipschitz polyhedron, which consists of the conducting region $\Omega_C \subset \Omega$ and its complement $\Omega_I = \Omega \setminus \bar{\Omega}_C$. We assume that Ω_C and Ω_I are Lipschitz polyhedrons, Ω_C is connected but not necessarily simply connected and by $\bar{\Omega}_C$ and $\bar{\Omega}_I$ we denote the closures of Ω_C and Ω_I , respectively. Applying the finite element method of [39] to (1.2), one obtains the following linear system:

$$\begin{bmatrix} M_C - iS_C & -iD^T & B_C^T & 0 \\ -iD & S_I + \tau B_I^T B_I & 0 & B_I^T \\ -B_C & 0 & 0 & 0 \\ 0 & -B_I & 0 & 0 \end{bmatrix} \begin{bmatrix} H_C \\ \tilde{E}_I \\ \tilde{Q} \\ \tilde{\Phi}_I \end{bmatrix} = \begin{bmatrix} -iF_C \\ -iG_I \\ 0 \\ 0 \end{bmatrix}, \tag{1.3}$$

where $M_C \in \mathbb{R}^{n_1 \times n_1}$ is a symmetric positive definite matrix, $S_C \in \mathbb{R}^{n_1 \times n_1}$ and $S_I \in \mathbb{R}^{n_2 \times n_2}$ are symmetric positive semidefinite matrices, $B_C \in \mathbb{R}^{m_1 \times n_1}$, $B_I \in \mathbb{R}^{m_2 \times n_2}$ and $D \in \mathbb{R}^{n_2 \times n_1}$ are real matrices, and τ is a given parameter. Unknown complex vectors H_C , \tilde{E}_I , $Q = i\tilde{Q}$ and $\Phi_I = i\tilde{\Phi}_I$ are, respectively, the coefficients of the finite element approximations of \mathbf{H}_C , $\tilde{\mathbf{E}}_I$, q and ϕ_I in some bases of finite element spaces. Recall that $\tilde{\mathbf{E}}_I$ is a suitable magnetic vector potential such that $\mathbf{H}_I = -(i\omega\mu)^{-1} \mathbf{curl} \tilde{\mathbf{E}}_I$, and q and ϕ_I are Lagrange multipliers — cf. [5, 38].

Setting

$$A := \begin{bmatrix} M_C - iS_C & -iD^T \\ -iD & S_I + \tau B_I^T B_I \end{bmatrix}, \quad B := \begin{bmatrix} B_C & 0 \\ 0 & B_I \end{bmatrix},$$

we can regard the system (1.3) as a non-Hermitian saddle point problem (1.1). Besides, considering the system (1.3), we usually distinguish the situations where the first Betti number of Ω_I is zero or greater than zero [5, 38]. In the former case, the matrix $S_I + \tau B_I^T B_I$ is symmetric positive definite whereas in the later it is symmetric positive semidefinite.

Iteration methods, such as the modified block SOR method and the Uzawa-like method [38], the block alternating splitting implicit (BASI) iteration method [5] and the alternating positive semidefinite splitting (APSS) iteration method [36] have been applied to the saddle point system (1.3). However, the convergence and the choice of relaxation parameters in modified block SOR and Uzawa-like methods are not well studied and for non-Hermitian saddle point system (1.3), the convergence of BASI and APSS iteration methods is slow. Therefore, the aim of this work is to construct a more effective preconditioner for the saddle point system (1.3).

Recently, shift-splitting (SS) preconditioners have been studied [14, 22–24, 29, 41]. For non-Hermitian saddle point problem, the shift-splitting preconditioner is defined by

$$\mathcal{P}_{SS} = \frac{1}{2}(\alpha I + \mathcal{A}) = \frac{1}{2} \begin{bmatrix} \alpha I + A & B^* \\ -B & \alpha I \end{bmatrix}, \quad (1.4)$$

where α is a positive constant and I is the identity matrix of a suitable size. In fact, the shift-splitting preconditioner can be generated by the following shift-splitting iteration method.

Method 1.1 (Shift-splitting iteration method). Let α be a positive constant and $[x^{(0)}; y^{(0)}]$ the initial guess vector. For $k = 0, 1, 2, \dots$ until certain stopping criterion is satisfied, compute

$$\frac{1}{2} \begin{bmatrix} \alpha I + A & B^* \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} x^{(k+1)} \\ y^{(k+1)} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \alpha I - A & -B^* \\ B & \alpha I \end{bmatrix} \begin{bmatrix} x^{(k)} \\ y^{(k)} \end{bmatrix} + \begin{bmatrix} f \\ g \end{bmatrix}. \quad (1.5)$$

Theoretical analysis shows that the stationary iteration method (1.5) for saddle point problems is unconditionally convergent. For saddle point problems with a Hermitian positive definite (1, 1) block matrix, the spectral properties of the shift-splitting preconditioned matrix are discussed in [37, 44]. A generalised shift-splitting preconditioner with an additional parameter β has been studied in [23, 24, 41–43]. Numerical results suggest that the shift-splitting preconditioner \mathcal{P}_{SS} and its generalisations are efficient in saddle point

problems of computational fluid dynamics [22–24, 44] and in meshfree discretisations of elasticity problems [37]. However, for time-harmonic current eddy models there are implementation difficulties associated with special block structure of the non-Hermitian saddle point matrix in (1.3).

Using the shift-splitting preconditioner \mathcal{P}_{SS} in (1.4) and following the ideas of [10, 28, 47], we establish a local positive (semi)definite shift-splitting (LPSS) preconditioner for non-Hermitian saddle point problems of time-harmonic current eddy models. A new LPSS preconditioner can be induced by the LPSS stationary iteration method. We analyse the convergence of the LPSS iteration method and spectral properties of the LPSS preconditioned matrix. Two numerical examples illustrate the feasibility and effectiveness of the new LPSS preconditioner. It is shown that this preconditioner outperforms the existing BASI, APSS and SS preconditioners for the saddle point problem (1.3).

The remainder of this paper is as follows. In Section 2, we introduce an LPSS preconditioner, study the convergence of the corresponding LPSS iteration method and analyse the spectral properties of the LPSS preconditioned saddle point matrix. In Section 3, the LPSS preconditioner is applied to time-harmonic eddy current models. Numerical results presented in Section 4, show the effectiveness of the LPSS preconditioner. Final conclusions and remarks are in Section 5.

2. An LPSS Preconditioner

Here, we consider a new local positive (semi)definite shift-splitting (LPSS) preconditioner for the non-Hermitian saddle point problem (1.1) and study its convergence. Besides, we also study the spectral properties of the LPSS preconditioned saddle point matrix.

Let $\alpha > 0$ be a positive constant, A be a non-Hermitian positive definite matrix and $A = P + S$ a splitting of A such that $P \in \mathbb{C}^{n \times n}$ and $S \in \mathbb{C}^{n \times n}$ are, respectively, positive definite and skew-Hermitian matrices. This yields the local positive definite shift-splitting

$$\mathcal{A} = \mathcal{P}_{LPSS} - \mathcal{Q}_{LPSS} \quad (2.1)$$

of \mathcal{A} with the matrices

$$\mathcal{P}_{LPSS} = \frac{1}{2} \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix}, \quad \mathcal{Q}_{LPSS} = \frac{1}{2} \begin{bmatrix} \alpha I - 2S & -B^* \\ B & \alpha I \end{bmatrix}. \quad (2.2)$$

Splitting (2.1) is used in the construction of the following LPSS iteration method for the problem (1.1).

Method 2.1. (LPSS iteration method) Let $A = P + S$ be the above splitting of the matrix A in (1.1), α a given positive constant and $[x^{(0)}; y^{(0)}]$ an initial guess vector. For $k = 0, 1, 2, \dots$ compute

$$\frac{1}{2} \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} x^{(k+1)} \\ y^{(k+1)} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} \alpha I - 2S & -B^* \\ B & \alpha I \end{bmatrix} \begin{bmatrix} x^{(k)} \\ y^{(k)} \end{bmatrix} + \begin{bmatrix} f \\ g \end{bmatrix} \quad (2.3)$$

until a stopping criterion is satisfied.

The LPSS iteration method is similar to a shift-splitting iteration method. In particular, if P coincides with the Hermitian part H of A , then the LPSS iteration method reduces to the modified shift-splitting iteration method in [47]. We note that in actual implementation, the one step LPSS iteration method requires significantly smaller computation costs than two-step iteration methods such as the HSS iteration method [11], the PSS iteration method [10], the BASI iteration method [5] and the APSS iteration method [36]. In order to accelerate the convergence rate of Krylov subspace iteration methods, the splitting matrix \mathcal{P}_{LPSS} can be used as a preconditioner. It is called the LPSS preconditioner. The linear system $\mathcal{A}w = p$ can be now written as a left-preconditioned system — viz.

$$\mathcal{P}_{LPSS}^{-1}\mathcal{A}w = \mathcal{P}_{LPSS}^{-1}p.$$

If A is non-Hermitian positive semidefinite matrix and P is a positive-semidefinite matrix, the LPSS preconditioner (2.2) can be constructed similar to the previous considerations. At any step of the method (2.3) and the LPSS preconditioned Krylov subspace methods, one has to find solutions of the generalised residual equation

$$\mathcal{P}_{LPSS}z = \frac{1}{2} \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \end{bmatrix} = r. \quad (2.4)$$

Algorithm 2.1 provides a framework for implementation of the LPSS iteration method and the LPSS preconditioner. It shows that the main cost is related to solving the Eq. (2.5). However, for specific problems, the linear subsystem may be positive definite and highly reducible, so that it can be easily solved by certain iterative methods. In particular, in Section 3, we consider certain applications connected with time-harmonic eddy current models.

Algorithm 2.1

Let $r = [r_1; r_2]$ and $z = [z_1; z_2]$ be, respectively, current and generalised residual vectors with $r_1, z_1 \in \mathbb{C}^n$ and $r_2, z_2 \in \mathbb{C}^m$.

The linear system (2.4) can be solved as follows:

- (1) Compute $t_1 = 2r_1 - (2/\alpha)B^*r_2$.
- (2) Find solutions of the equation

$$\left(\alpha I + 2P + \frac{1}{\alpha}B^*B \right) z_1 = t_1. \quad (2.5)$$

- (3) Compute $z_2 = (1/\alpha)(Bz_1 + 2r_2)$.
-

2.1. Convergence of LPSS iteration method

Consider the convergence of the LPSS iteration method (2.3). It can be rewritten as a fixed point problem — viz.

$$w^{(k+1)} = \Gamma w^{(k)} + q,$$

$$w^{(k)} = [x^{(k)}; y^{(k)}]$$

with the iteration matrix

$$\Gamma = \mathcal{P}_{LPSS}^{-1} \mathcal{Q}_{LPSS} = \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix}^{-1} \begin{bmatrix} \alpha I - 2S & -B^* \\ B & \alpha I \end{bmatrix},$$

and $q = \mathcal{P}_{LPSS}^{-1} p$.

Let $\rho(\Gamma)$ denote the spectral radius of Γ . The LPSS iteration method converges if and only if $\rho(\Gamma) < 1$. If λ is an eigenvalue of Γ and $[u; v]$ is the corresponding eigenvector, then

$$\begin{bmatrix} \alpha I - 2S & -B^* \\ B & \alpha I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix}, \quad (2.6)$$

or, equivalently, if

$$\alpha(\lambda - 1)u + 2(\lambda P + S)u + (\lambda + 1)B^*v = 0, \quad (2.7)$$

$$(\lambda + 1)Bu = \alpha(\lambda - 1)v. \quad (2.8)$$

Assuming that $\lambda \neq 0$, we prove some auxiliary results.

Lemma 2.1. *Let α be a positive constant, $A \in \mathbb{C}^{n \times n}$ a non-Hermitian positive definite matrix, $A = P + S$ a splitting of A such that $P \in \mathbb{C}^{n \times n}$ and $S \in \mathbb{C}^{n \times n}$ are, respectively, positive-definite and skew-Hermitian matrices and $B \in \mathbb{C}^{m \times n}$ a full row rank matrix. If λ is an eigenvalue of the LPSS iteration matrix Γ and $[u; v]$ its corresponding eigenvector, then*

(1) $\lambda \neq \pm 1$;

(2) u is a nonzero vector.

Proof. We prove this lemma by contradiction. If $\lambda = 1$, the Eq. (2.6) takes the form

$$\begin{bmatrix} A & B^* \\ -B & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = 0.$$

Since the matrix \mathcal{A} is nonsingular, we obtain $u = 0$ and $v = 0$ that contradicts the assumption that $[u; v]$ is an eigenvector. Hence, $\lambda \neq 1$.

If $\lambda = -1$, the Eq. (2.6) takes the form

$$\begin{bmatrix} \alpha I + P - S & 0 \\ 0 & \alpha I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = 0.$$

The non-singularity of $\alpha I + P - S$ and the condition $\alpha > 0$ yield $u = 0$ and $v = 0$, contradicting the assumption that $[u; v]$ is an eigenvector. Hence, $\lambda \neq -1$.

Assume now that $u = 0$. The Eq. (2.8) takes the form

$$\alpha(\lambda - 1)v = 0,$$

and since $\alpha > 0$ and $\lambda \neq 1$, this implies $v = 0$, which is not possible. Thus, $u \neq 0$, which finishes the proof. \square

Lemma 2.2 (cf. Horn and Johnson [32]). *If $S \in \mathbb{C}^{n \times n}$ is a skew-Hermitian matrix, then iS is a Hermitian matrix and u^*Su is a purely imaginary number or zero for any $u \in \mathbb{C}^n$.*

Lemma 2.3 (cf. Young [46]). *Both roots of the complex quadratic equation $\lambda^2 + \phi\lambda + \psi = 0$ have modulus less than one if and only if $|\phi - \bar{\phi}\psi| + |\psi|^2 < 1$, where $\bar{\phi}$ denotes the complex conjugate of ϕ .*

Theorem 2.1. *Let α be a positive constant, $A \in \mathbb{C}^{n \times n}$ a non-Hermitian positive definite matrix, $A = P + S$ a splitting of A such that $P \in \mathbb{C}^{n \times n}$ and $S \in \mathbb{C}^{n \times n}$ are, respectively, positive-definite and skew-Hermitian matrices and $B \in \mathbb{C}^{m \times n}$ a full row rank matrix. Besides, let λ be an eigenvalue of the LPSS iteration matrix Γ and $[u; v]$ the corresponding eigenvector. Set*

$$a + ib := \frac{u^*Pu}{u^*u}, \quad c := \frac{u^*B^*Bu}{u^*u}, \quad id := \frac{u^*Su}{u^*u}. \quad (2.9)$$

The LPSS iteration method converges if and only if the parameter α satisfies the inequality

$$\alpha > \frac{d^2 - a^2 - b^2 + \sqrt{(d^2 - a^2 - b^2)^2 + 4c(b-d)^2}}{2a}. \quad (2.10)$$

Proof. According to Lemma 2.1, $\lambda \neq 1$. Therefore, it follows from (2.8) that

$$v = \frac{\lambda + 1}{\alpha(\lambda - 1)}Bu.$$

Substituting it into (2.7) yields

$$\alpha(\lambda - 1)u + 2(\lambda P + S)u + \frac{(\lambda + 1)^2}{\alpha(\lambda - 1)}B^*Bu = 0. \quad (2.11)$$

Since $u \neq 0$, we can multiply the Eq. (2.11) by $u^*/(u^*u)$ from the left, thus obtaining

$$\alpha(\lambda - 1) + 2\lambda \frac{u^*Pu}{u^*u} + 2 \frac{u^*Su}{u^*u} + \frac{(\lambda + 1)^2}{\alpha(\lambda - 1)} \frac{u^*B^*Bu}{u^*u} = 0. \quad (2.12)$$

Using the notation (2.9), we write the Eq. (2.12) in the form

$$\lambda^2 + \frac{2c - 2\alpha^2 - 2\alpha a + i2\alpha(d - b)}{\alpha^2 + 2\alpha a + c + i2\alpha b} \lambda + \frac{\alpha^2 + c - i2\alpha d}{\alpha^2 + 2\alpha a + c + i2\alpha b} = 0, \quad (2.13)$$

where $a, b, c, d \in \mathbb{R}$, $a > 0$ and $c \geq 0$ due to the properties of the matrices P, S and B^*B . Set

$$h_1 := \alpha^2 + 2\alpha a + c, \quad h_2 := 2\alpha b, \quad h_3 := 2c - 2\alpha^2 - 2\alpha a, \quad h_4 = \alpha^2 + c, \quad h_5 := 2\alpha d,$$

and write the Eq. (2.13) as

$$\lambda^2 + \phi\lambda + \psi = 0, \quad (2.14)$$

where

$$\phi = \frac{h_3 + i(h_5 - h_2)}{h_1 + ih_2} = \frac{h_1h_3 + h_2h_5 - h_2^2 - i(h_2h_3 - h_1h_5 + h_1h_2)}{h_1^2 + h_2^2},$$

$$\psi = \frac{h_4 - ih_5}{h_1 + ih_2} = \frac{h_1h_4 - h_2h_5 - i(h_2h_4 + h_1h_5)}{h_1^2 + h_2^2}.$$

According to Lemma 2.3, the roots of the Eq. (2.14) satisfy the inequality $|\lambda| < 1$ if and only if

$$|\phi - \bar{\phi}\psi| + |\psi|^2 < 1. \quad (2.15)$$

Direct computations show that

$$\phi - \bar{\phi}\psi = \frac{(h_1 - h_4)h_3 - h_2^2 + h_5^2 - i(h_1 + h_3 + h_4)(h_2 - h_5)}{h_1^2 + h_2^2}, \quad |\psi|^2 = \frac{h_4^2 + h_5^2}{h_1^2 + h_2^2}.$$

Therefore, the inequality (2.15) holds if and only if

$$\begin{aligned} & [(h_1 - h_4)h_3 - h_2^2 + h_5^2]^2 + (h_1 + h_3 + h_4)^2(h_2 - h_5)^2 < (h_1^2 + h_2^2 - h_4^2 - h_5^2)^2, \\ & h_1^2 + h_2^2 - h_4^2 - h_5^2 > 0, \end{aligned}$$

which is equivalent to

$$\alpha^2 a^2 + \alpha a(a^2 + b^2 - d^2) - c(b - d)^2 > 0. \quad (2.16)$$

Since $\alpha > 0$, the inequality (2.16) holds if and only if α satisfies the condition (2.10). Therefore, the spectral radius $\rho(\Gamma)$ is smaller than one and the method converges if and only if the inequality (2.10) holds. \square

Remark 2.1. If $P = H$, then (2.9) implies $b = 0$ and the condition (2.10) takes the form

$$\alpha > \frac{d^2 - a^2 + \sqrt{(d^2 - a^2)^2 + 4cd^2}}{2a},$$

which is consistent with the convergence condition for a modified shift-splitting iteration method in [47, Theorem 2.4]. A condition sufficient for the convergence of the LPSS iteration method in this case is

$$\alpha > \frac{\rho^2(S) - \lambda_{\min}^2(H) + \sqrt{(\rho^2(S) - \lambda_{\min}^2(H))^2 + 4\rho(B^*B)\rho^2(S)}}{2\lambda_{\min}(H)},$$

where $\lambda_{\min}(H)$ is the smallest eigenvalue of H and $\rho(S)$ the spectral radius of S . However, for general non-Hermitian positive definite matrices P , sufficient convergence conditions are not known.

Remark 2.2. If A is a positive semidefinite matrix, then the matrix P in the splitting $A = P + S$ of A is also positive semidefinite. If in addition $a = 0$, the convergence condition (2.10) fails and the LPSS iteration method (2.3) diverges. Nevertheless, the matrix \mathcal{P}_{LPSS} can be used as a preconditioner for Krylov subspace methods.

The proof of Theorem 2.1 shows that under the condition (2.10), the spectral radius of the LPSS iteration matrix Γ is smaller than one, and the LPSS iteration method for the non-Hermitian saddle point problem (1.1) with a positive definite matrix A converges. However, in actual computations these parameters are difficult to determine. Moreover, even for an optimal parameter α , the method can converge slowly. Therefore, the splitting matrix \mathcal{P}_{LPSS} in (2.2) is often used as a preconditioner to accelerate the convergence rate of Krylov subspace methods such as GMRES. It is well-known that the clustered spectrum of the preconditioned matrix is related to a fast convergence, so now we are going to discuss the spectral properties of the LPSS preconditioned matrices $\mathcal{P}_{LPSS}^{-1}\mathcal{A}$.

2.2. Spectral properties of preconditioned matrix

In this subsection, we describe the spectral properties of the LPSS preconditioned matrices $\mathcal{P}_{LPSS}^{-1}\mathcal{A}$ when the (1,1) leading block matrix A is positive definite or positive semidefinite.

Theorem 2.2. Let α be a positive constant, $A \in \mathbb{C}^{n \times n}$ a non-Hermitian positive definite matrix, $A = P + S$ a splitting of A such that $P \in \mathbb{C}^{n \times n}$ and $S \in \mathbb{C}^{n \times n}$ are, respectively, positive-definite and skew-Hermitian matrices and $B \in \mathbb{C}^{m \times n}$ a full row rank matrix. If μ is an eigenvalue of the LPSS preconditioned matrix $\mathcal{P}_{LPSS}^{-1}\mathcal{A}$ and $[u; v]$ an associated eigenvector, then

- (1) $\mu \neq 2$, $\mu \neq 0$ and $u \neq 0$.
- (2) If $v = 0$, then $u \in \text{null}(B)$ and $\mu \rightarrow 1 + id/(a + ib)$ as $\alpha \rightarrow 0_+$, where a, b and d are defined as in (2.9).
- (3) If $v \neq 0$, then $\mu \rightarrow 2$ as $\alpha \rightarrow 0_+$.

Proof. Let λ be an eigenvalue of the iteration matrix Γ . Because

$$\mathcal{P}_{LPSS}^{-1}\mathcal{A} = \mathcal{P}_{LPSS}^{-1}(\mathcal{P}_{LPSS} - \mathcal{Q}_{LPSS}) = I - \mathcal{P}_{LPSS}^{-1}\mathcal{Q}_{LPSS} = I - \Gamma,$$

we have $\mu = 1 - \lambda$. It follows from Lemma 2.1 that $\mu \neq 2$, $\mu \neq 0$ and $u \neq 0$.

The eigenvalue and the eigenvector of the LPSS preconditioned matrix $\mathcal{P}_{LPSS}^{-1}\mathcal{A}$ satisfy the following generalised eigenvalue problem:

$$\begin{bmatrix} A & B^* \\ -B & 0 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \frac{\mu}{2} \begin{bmatrix} \alpha I + 2P & B^* \\ -B & \alpha I \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix},$$

which is equivalent to the equations

$$[2A - \mu(\alpha I + 2P)]u = (\mu - 2)B^*v, \quad (2.17)$$

$$(\mu - 2)Bu = \alpha\mu v. \quad (2.18)$$

If $v = 0$, the Eq. (2.18) implies $Bu = 0$ since $\mu \neq 2$ and $\alpha > 0$. Therefore, the Eq. (2.17) takes the form

$$2Au = \mu(\alpha I + 2P)u. \quad (2.19)$$

Multiplying it by $u^*/(u^*u)$ from the left yields

$$\mu = \frac{2u^*Au/(u^*u)}{\alpha + 2u^*Pu/(u^*u)} = 1 - \frac{\alpha - 2u^*Su/(u^*u)}{\alpha + 2u^*Pu/(u^*u)} = 1 - \frac{\alpha - i2d}{\alpha + 2a + i2b}. \quad (2.20)$$

It follows that if $\alpha \rightarrow 0_+$, then $\mu \rightarrow 1 + id/(a + ib)$.

If $v \neq 0$, then the Eq. (2.18) can be written as $v = ((\mu - 2)/\alpha\mu)Bu$ and substituting it into the Eq. (2.17), we obtain

$$2Au - \mu(\alpha I + 2P)u - \frac{(\mu - 2)^2}{\alpha\mu}B^*Bu = 0. \quad (2.21)$$

Multiplying (2.21) by $u^*/(u^*u)$ from the left leads to the equation

$$\mu^2(\alpha^2 + 2a\alpha + c + i2b\alpha) - 2[a\alpha + 2c + i(b + d)\alpha]\mu + 4c = 0.$$

This equation has the roots

$$\mu_{\pm} = \frac{\alpha\alpha + 2c + i(b + d)\alpha \pm (\gamma_1 + i\gamma_2)}{\alpha^2 + 2a\alpha + c + i2b\alpha} = 1 + \frac{c - \alpha[\alpha + a + i(b - d)] \pm (\gamma_1 + i\gamma_2)}{c + \alpha(\alpha + 2a + i2b)}, \quad (2.22)$$

where $\gamma_1, \gamma_2 \in \mathbb{R}$ and $\gamma_1 + i\gamma_2$ is one of the square roots of $\theta_1 + i\theta_2$ with

$$\theta_1 = \alpha^2[a^2 - (b + d)^2 - 4c] - 4aac, \quad \theta_2 = 2\alpha^2a(b + d) + 4ac(d - b). \quad (2.23)$$

Direct computations show that

$$\gamma_1 = \sqrt{\frac{\sqrt{\theta_1^2 + \theta_2^2} + \theta_1}{2}}, \quad \gamma_2 = \frac{\theta_2}{2\gamma_1} = \text{sign}(\theta_2) \sqrt{\frac{\sqrt{\theta_1^2 + \theta_2^2} - \theta_1}{2}}.$$

It follows from (2.23) that $\theta_1 \rightarrow 0$ and $\theta_2 \rightarrow 0$ as $\alpha \rightarrow 0_+$, so that $\gamma_1 \rightarrow 0$ and $\gamma_2 \rightarrow 0$. In addition, $\mu \rightarrow 2$ as $\alpha \rightarrow 0_+$. Therefore, the eigenvalues of the LPSS preconditioned matrix are close to $1 + id/(a + ib)$ or to 2 if α approaches zero but $\mu \neq 2$. \square

Following Theorem 2.2, we can formulate a result for positive semidefinite matrices A and P .

Theorem 2.3. *Let α be a positive constant, $A \in \mathbb{C}^{n \times n}$ a non-Hermitian positive semidefinite matrix, $A = P + S$ a splitting of A such that $P \in \mathbb{C}^{n \times n}$ and $S \in \mathbb{C}^{n \times n}$ are, respectively, positive semidefinite and skew-Hermitian matrices and $B \in \mathbb{C}^{m \times n}$ a full row rank matrix. If μ is an eigenvalue of the LPSS preconditioned matrix $\mathcal{P}_{LPSS}^{-1}A$ and $[u; v]$ an associated eigenvector, then*

- (1) $u \neq 0$.
- (2) If $v = 0$ and $\mu \neq 2$, then $u \in \text{null}(B)$ and $\mu \rightarrow 1 + id/(a + ib)$ as $\alpha \rightarrow 0_+$, where a, b and d are defined as in (2.9).
- (3) If $v \neq 0$, then $\mu \rightarrow 2$ as $\alpha \rightarrow 0_+$.

3. Applications to Time-Harmonic Eddy Current Models

We now apply the LPSS preconditioner to the non-Hermitian saddle point problem (1.3) arising in finite element discretisations of the hybrid formulations of time-harmonic eddy current model (1.2). The problem (1.3) is considered in both simple and general topology cases.

We note that the splitting matrices P and S can be chosen in different ways [10, 28]. For example, if $P = A$ and $S = 0$, the LPSS preconditioner reduces to the SS preconditioner. If $P = H$ and $S = \bar{S}$, the LPSS preconditioner is the same as in [47], where \bar{S} is the skew-Hermitian part of the non-Hermitian positive (semi)definite matrix A . One can also choose $P = D_H + 2L_H$ and $S = L_H^T - L_H + \bar{S}$, where D_H and L_H are, respectively, the diagonal and the strictly lower triangular parts of H . In this case, P and S are, respectively, positive (semi)definite and skew-Hermitian matrices. Here, we choose special splitting matrices such that the corresponding LPSS preconditioner is easily implementable and the LPSS preconditioned GMRES method converges fast. Using the special structure of the non-Hermitian saddle point matrix in (1.3), we present a special LPSS preconditioner in both simple and general topology cases.

3.1. Simple topology case

In the simple topology case, the first Betti number of Ω_I is equal to zero. This condition guarantees the invertibility of the matrix

$$\mathcal{A}_I = \begin{bmatrix} A_I & B_I^T \\ -B_I & 0 \end{bmatrix}, \quad (3.1)$$

where $A_I = iS_I$ and the matrix S_I is symmetric positive semidefinite. The matrix \mathcal{A}_I is nonsingular if and only if

$$\text{null}(S_I) \cap \text{null}(B_I) = \{0\} \quad \text{and} \quad \text{null}(B_I^T) = \{0\}.$$

Hence, for any constant $\tau > 0$, the matrix $S_I + \tau B_I^T B_I$ is symmetric positive definite. Thus, the non-Hermitian 2×2 block matrix

$$A = \begin{bmatrix} M_C - iS_C & -iD^T \\ -iD & S_I + \tau B_I^T B_I \end{bmatrix}$$

is positive definite. In this case, we consider the splitting $A = P + S$ with a positive definite matrix P and a skew-Hermitian matrix S — viz.

$$P = \begin{bmatrix} M_C - iS_C & 0 \\ 0 & S_I + \tau B_I^T B_I \end{bmatrix}, \quad S = \begin{bmatrix} 0 & -iD^T \\ -iD & 0 \end{bmatrix}.$$

The corresponding splitting of \mathcal{A} is

$$\mathcal{A} = \mathcal{P}_{LPSS} - \mathcal{Q}_{LPSS} = \frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & 0 & B_C^T & 0 \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 & B_I^T \\ -B_C & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix}$$

$$-\frac{1}{2} \begin{bmatrix} \alpha I & i2D^T & -B_C^T & 0 \\ i2D & \alpha I & 0 & -B_I^T \\ B_C & 0 & \alpha I & 0 \\ 0 & B_I & 0 & \alpha I \end{bmatrix} \quad (3.2)$$

with the induced LPSS preconditioner \mathcal{P}_{LPSS} .

In order to improve the convergence of the GMRES method by the above LPSS preconditioner \mathcal{P}_{LPSS} , at each iteration we have to solve the following generalised residual equation:

$$\begin{aligned} \mathcal{P}_{LPSS} \bar{r} &= \frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & 0 & B_C^T & 0 \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 & B_I^T \\ -B_C & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix} \\ &\times \begin{bmatrix} \bar{r}_1 \\ \bar{r}_2 \\ \bar{r}_3 \\ \bar{r}_4 \end{bmatrix} = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_2 \\ \bar{z}_3 \\ \bar{z}_4 \end{bmatrix} = \bar{z}, \end{aligned} \quad (3.3)$$

where $\bar{r} = [\bar{r}_1; \bar{r}_2; \bar{r}_3; \bar{r}_4]$ and $\bar{z} = [\bar{z}_1; \bar{z}_2; \bar{z}_3; \bar{z}_4]$ are, respectively, current and generalised residual vectors. System (3.3) can be reduced to the following linear subsystems:

$$\frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & B_C^T \\ -B_C & \alpha I \end{bmatrix} \begin{bmatrix} \bar{r}_1 \\ \bar{r}_3 \end{bmatrix} = \begin{bmatrix} \bar{z}_1 \\ \bar{z}_3 \end{bmatrix}, \quad (3.4)$$

$$\frac{1}{2} \begin{bmatrix} \alpha I + 2(S_I + \tau B_I^T B_I) & B_I^T \\ -B_I & \alpha I \end{bmatrix} \begin{bmatrix} \bar{r}_2 \\ \bar{r}_4 \end{bmatrix} = \begin{bmatrix} \bar{z}_2 \\ \bar{z}_4 \end{bmatrix}. \quad (3.5)$$

The systems (3.4) and (3.5) are positive definite and coupled. For the linear subsystem (3.4), we can first determine \bar{r}_1 from the equation

$$\left(\alpha I + 2M_C + \frac{1}{\alpha} B_C^T B_C - i2S_C \right) \bar{r}_1 = 2\bar{z}_1 - \frac{2}{\alpha} B_C^T \bar{z}_3,$$

and then obtain \bar{r}_3 — viz.

$$\bar{r}_3 = \frac{1}{\alpha} (2\bar{z}_3 + B_C \bar{r}_1).$$

Analogously, for the system (3.5), we first derive \bar{r}_2 from the equation

$$\left(\alpha I + 2S_I + \left(2\tau + \frac{1}{\alpha} \right) B_I^T B_I \right) \bar{r}_2 = 2\bar{z}_2 - \frac{2}{\alpha} B_I^T \bar{z}_4,$$

and then write

$$\bar{r}_4 = \frac{1}{\alpha} (2\bar{z}_4 + B_I \bar{r}_2).$$

These considerations lead to the following algorithm for implementing the LPSS preconditioner \mathcal{P}_{LPSS} in (3.2).

Algorithm 3.1

Let $\bar{r} = [\bar{r}_1; \bar{r}_2; \bar{r}_3; \bar{r}_4]$ and $\bar{z} = [\bar{z}_1; \bar{z}_2; \bar{z}_3; \bar{z}_4]$, where $\bar{r}_1, \bar{z}_1 \in \mathbb{C}^{n_1}$, $\bar{r}_2, \bar{z}_2 \in \mathbb{C}^{n_2}$, $\bar{r}_3, \bar{z}_3 \in \mathbb{C}^{m_1}$ and $\bar{r}_4, \bar{z}_4 \in \mathbb{C}^{m_2}$.

The solution of the linear system (3.3) can be found in the following way:

(1) Solve the equation

$$\left(\alpha I + 2M_C + \frac{1}{\alpha} B_C^T B_C - i2S_C \right) \bar{r}_1 = 2\bar{z}_1 - \frac{2}{\alpha} B_C^T \bar{z}_3$$

for \bar{r}_1 .

(2) Solve the equation

$$\left(\alpha I + 2S_I + \left(2\tau + \frac{1}{\alpha} \right) B_I^T B_I \right) \bar{r}_2 = 2\bar{z}_2 - \frac{2}{\alpha} B_I^T \bar{z}_4$$

for \bar{r}_2 .

(3) Compute $\bar{r}_3 = (1/\alpha)(2\bar{z}_3 + B_C \bar{r}_1)$.

(4) Compute $\bar{r}_4 = (1/\alpha)(2\bar{z}_4 + B_I \bar{r}_2)$.

We note that at each iteration step the main computational cost lies in solving the linear systems with the coefficient matrices $\alpha I + 2M_C + (1/\alpha)B_C^T B_C - i2S_C$ and $\alpha I + 2S_I + (2\tau + (1/\alpha))B_I^T B_I$. The matrix $\alpha I + 2M_C + (1/\alpha)B_C^T B_C - i2S_C$ can be regarded as a complex symmetric matrix with a symmetric positive definite real part and a symmetric positive semidefinite imaginary part. Therefore, the modified HSS iteration method [7] and the preconditioned MHSS iteration method [8] can be used as inexact solvers. For the linear system with the coefficient matrix $\alpha I + 2S_I + (2\tau + (1/\alpha))B_I^T B_I$, one can use the sparse Cholesky factorisation or an incomplete Cholesky factorisation.

3.2. General topology case

In the general topology case, $B_I \tilde{E}_I = 0$ and the Lagrange multiplier Φ_I introduced in the insulator region satisfies the condition $\Phi_I = 0$. The saddle point problem (1.3) can be reduced to the following system:

$$\begin{bmatrix} M_C - iS_C & -iD^T & B_C^T \\ -iD & S_I + \tau B_I^T B_I & 0 \\ -B_C & 0 & 0 \end{bmatrix} \begin{bmatrix} H_C \\ \tilde{E}_I \\ \tilde{Q} \end{bmatrix} = \begin{bmatrix} -iF_C \\ -iG_I \\ 0 \end{bmatrix}. \quad (3.6)$$

Set

$$A = \begin{bmatrix} M_C - iS_C & -iD^T \\ -iD & S_I + \tau B_I^T B_I \end{bmatrix}, \quad B = \begin{bmatrix} -B_C & 0 \end{bmatrix}.$$

System (3.6) is also a non-Hermitian saddle point problem but now the first Betti number of Ω_I is greater than zero, so that the matrix \mathcal{A}_I in (3.1) is singular [5, 38]. Thus,

the (2, 2) block matrix $S_I + \tau B_I^T B_I$ in (3.6) is symmetric positive semidefinite. Note that A is a non-Hermitian positive semidefinite block matrix and B is a full rank matrix. We consider a splitting $A = P + S$ with a positive semidefinite matrix P and a skew-Hermitian matrix S — viz.

$$P = \begin{bmatrix} M_C - iS_C & 0 \\ 0 & S_I + \tau B_I^T B_I \end{bmatrix}, \quad S = \begin{bmatrix} 0 & -iD^T \\ -iD & 0 \end{bmatrix}.$$

The corresponding splitting of \mathcal{A} is

$$\begin{aligned} \mathcal{A} = \mathcal{P}_{LPSS} - \mathcal{Q}_{LPSS} = & \frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & 0 & B_C^T \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 \\ -B_C & 0 & \alpha I \end{bmatrix} \\ & - \frac{1}{2} \begin{bmatrix} \alpha I & i2D^T & -B_C^T \\ i2D & \alpha I & 0 \\ B_C & 0 & \alpha I \end{bmatrix} \end{aligned} \quad (3.7)$$

with the induced LPSS preconditioner \mathcal{P}_{LPSS} .

In order to use the LPSS preconditioner \mathcal{P}_{LPSS} in (3.7), at each iteration we have to find solution of the following generalised residual equation:

$$\begin{aligned} \mathcal{P}_{LPSS} \tilde{r} = & \frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & 0 & B_C^T \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 \\ -B_C & 0 & \alpha I \end{bmatrix} \\ & \times \begin{bmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{bmatrix} = \begin{bmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{bmatrix} = \tilde{z}, \end{aligned} \quad (3.8)$$

where $\tilde{r} = [\tilde{r}_1; \tilde{r}_2; \tilde{r}_3]$ and $\tilde{z} = [\tilde{z}_1; \tilde{z}_2; \tilde{z}_3]$ are, respectively, current and generalised residual vectors. For (3.8), we can solve the equation

$$(\alpha I + 2S_I + 2\tau B_I^T B_I) \tilde{r}_2 = 2\tilde{z}_2,$$

and the reduced linear system

$$\frac{1}{2} \begin{bmatrix} \alpha I + 2(M_C - iS_C) & B_C^T \\ -B_C & \alpha I \end{bmatrix} \begin{bmatrix} \tilde{r}_1 \\ \tilde{r}_3 \end{bmatrix} = \begin{bmatrix} \tilde{z}_1 \\ \tilde{z}_3 \end{bmatrix}.$$

The system above has the same structure as the system (3.4) and to implement the LPSS preconditioner \mathcal{P}_{LPSS} in (3.7), we have the following algorithm.

Algorithm 3.2

Let $\tilde{r} = [\tilde{r}_1; \tilde{r}_2; \tilde{r}_3]$ and $\tilde{z} = [\tilde{z}_1; \tilde{z}_2; \tilde{z}_3]$, where $\tilde{r}_1, \tilde{z}_1 \in \mathbb{C}^{n_1}$, $\tilde{r}_2, \tilde{z}_2 \in \mathbb{C}^{n_2}$ and $\tilde{r}_3, \tilde{z}_3 \in \mathbb{C}^{m_1}$. The linear system (3.8) can be solved in the following way:

(1) Solve

$$\left(\alpha I + 2M_C + \frac{1}{\alpha} B_C^T B_C - i2S_C \right) \tilde{r}_1 = 2\tilde{z}_1 - \frac{2}{\alpha} B_C^T \tilde{z}_3$$

for \tilde{r}_1 .

(2) Solve $(\alpha I + 2S_I + 2\tau B_I^T B_I) \tilde{r}_2 = 2\tilde{z}_2$ for \tilde{r}_2 .

(3) Compute $\tilde{r}_3 = (1/\alpha)(2\tilde{z}_3 + B_C \tilde{r}_1)$.

The systems above can be solved by the methods discussed in Algorithm 3.1.

4. Numerical Experiments

In this section, we use numerical examples from [38] to test the feasibility and effectiveness of the LPSS preconditioner proposed. We compare it with the BASI preconditioner [5], the APSS preconditioner [36], the SS preconditioner [23] and the MSS preconditioner [47] in terms of the number of iteration steps IT and elapsed CPU time (in seconds). These preconditioned GMRES methods are used to solve non-Hermitian saddle point problems arising in finite element discretisations of time-harmonic eddy current models.

In actual computations, all experiments start with the zero vector and stop if

$$\text{RES} = \frac{\|b - Ax^{(k)}\|_2}{\|b\|_2} \leq 10^{-5}$$

or if the number of iterations exceeds 1500. Computations are carried out in MATLAB environment (version R2016a) on the high-performance computing platform of Nantong University. The linear subsystems are solved by sparse Cholesky or LU factorisation in combination with an approximate minimum degree (AMD) of column AMD reordering. The parameter τ involved in the saddle point problem (1.3) and in four preconditioners is $\tau = \|S_I\|_2 / \|B_I\|_2^2$. This balances matrices S_I and $B_I^T B_I$ in the Euclidean norm — cf. [5, 38].

Example 4.1. Consider the hybrid formulation of the complete eddy current model (1.2), where the conductor Ω_C and the domain Ω are two cubes centered at the origin with edges of length 2 and 10, respectively. We set $\omega = \mu = \sigma_{|\Omega_C} = 1$ and $\sigma_{|\Omega_I} \equiv 0$, cf. [38].

The first Betti number here is zero, so that the LPSS preconditioner \mathcal{P}_{LPSS} is defined as in (3.2). On the other hand, the BASI preconditioner

$$\mathcal{P}_{BASI(a_2)} = \frac{1}{2\alpha} (\alpha I + \mathcal{A}_{1-BASI(a_2)}) (\alpha I + \mathcal{A}_{2-BASI(a_2)})$$

$$= \frac{1}{2\alpha} \begin{bmatrix} \alpha I + M_C - iS_C & 0 & B_C^T & 0 \\ 0 & \alpha I + S_I + \tau B_I^T B_I & 0 & B_I^T \\ -B_C & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix} \\ \times \begin{bmatrix} \alpha I & -iD^T & 0 & 0 \\ -iD & \alpha I & 0 & 0 \\ 0 & 0 & \alpha I & 0 \\ 0 & 0 & 0 & \alpha I \end{bmatrix},$$

in [5] has the best numerical performance in the three forms. The APSS preconditioner in [36], the SS preconditioner in [23] and the MSS preconditioner in [47] are respectively defined by

$$\mathcal{P}_{APSS} = \frac{1}{2\alpha} (\alpha I + \mathcal{A}_{1-APSS})(\alpha I + \mathcal{A}_{2-APSS}) \\ = \frac{1}{2\alpha} \begin{bmatrix} \alpha I + M_C - iS_C & -iD^T & B_C^T & 0 \\ -iD & \alpha I & 0 & 0 \\ -B_C & 0 & \alpha I & 0 \\ 0 & 0 & 0 & \alpha I \end{bmatrix} \begin{bmatrix} \alpha I & 0 & 0 & 0 \\ 0 & \alpha I + S_I + \tau B_I^T B_I & 0 & B_I^T \\ 0 & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix}, \\ \mathcal{P}_{SS} = \frac{1}{2}(\alpha I + \mathcal{A}) = \frac{1}{2} \begin{bmatrix} \alpha I + M_C - iS_C & -iD^T & B_C^T & 0 \\ -iD & \alpha I + S_I + \tau B_I^T B_I & 0 & B_I^T \\ -B_C & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix}, \\ \mathcal{P}_{MSS} = \frac{1}{2} \begin{bmatrix} \alpha I + 2M_C & 0 & B_C^T & 0 \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 & B_I^T \\ -B_C & 0 & \alpha I & 0 \\ 0 & -B_I & 0 & \alpha I \end{bmatrix}.$$

We use four successively refined meshes and obtain four saddle point linear systems (1.3). Table 1 shows the degree of freedom (DOF) for each discretised mesh and the corresponding parameter τ .

Table 1: Example 4.1. DOF and parameter τ .

DOF	n_1	n_2	m_1	m_2	τ
1958	236	1378	107	237	0.2803
13860	1516	10298	431	1615	1.3201
44820	4722	33963	971	5164	3.0582
103954	10736	79576	1728	11914	5.8942

Here, we adopt a practical algebraic estimation method from [36] to obtain the parameter α in the BASI, APSS, SS, MSS and LPSS preconditioners. In particular, we have

$$\alpha_{BASI(a_2)} = (\|\mathcal{A}_{1-BASI(a_2)}\|_F + \|\mathcal{A}_{2-BASI(a_2)}\|_F) / 2n, \quad \alpha_{MSS} = \|\mathcal{A}_{MSS}\|_F / n, \\ \alpha_{APSS} = (\|\mathcal{A}_{1-APSS}\|_F + \|\mathcal{A}_{2-APSS}\|_F) / 2n, \quad \alpha_{SS} = \|\mathcal{A}\|_F / n, \quad \alpha_{LPSS} = \|\mathcal{A}_{LPSS}\|_F / n,$$

where $\|\cdot\|_F$ is the Frobenius norm and

$$\mathcal{A}_{LPSS} = \begin{bmatrix} -2S & -B^* \\ B & 0 \end{bmatrix}, \quad \mathcal{A}_{MSS} = \begin{bmatrix} -2\bar{S} & -B^* \\ B & 0 \end{bmatrix}$$

with the matrix

$$\bar{S} = \begin{bmatrix} -iS_c & -iD^T \\ -iD & 0 \end{bmatrix}.$$

In Example 4.1, the MSS preconditioned GMRES method with α_{MSS} diverges even with small DOF. For the MSS preconditioner with the same parameter as the LPSS preconditioner, the numerical results are better. Table 2 shows the values of α for four successively refined meshes. Table 3 presents numerical results for the GMRES method without any preconditioner (denoted by I) and the preconditioned GMRES method with BASI(a2), APSS, SS, MSS, and LPSS preconditioners with respect to different DOFs. Table 3 shows that the convergence of GMRES and MSS preconditioned GMRES methods is very slow and if DOF is large, they diverge within the prescribed iteration steps. The other preconditioned GMRES methods provide satisfactory approximations for the exact solution. This indicates that the practical estimation of the parameter α suggested in [36], is efficient for BASI, APSS, SS and LPSS preconditioners but not for MSS one. For DOF=1958, the SS preconditioned GMRES method is only slightly better than the LPSS preconditioned GMRES method in terms of

Table 2: Example 4.1. Parameter α for various preconditioners.

DOF	BASI(a2)	APSS	SS	MSS	LPSS
1958	0.0662	0.0871	0.1294	0.0331	0.0331
13860	0.0589	0.0793	0.1170	0.0081	0.0081
44820	0.0498	0.0679	0.0992	0.0033	0.0033
103954	0.0444	0.0607	0.0885	0.0017	0.0017

Table 3: Example 4.1. Numerical results for preconditioned GMRES methods.

DOF		I	BASI(a2)	APSS	SS	MSS	LPSS
1958	IT	853	146	56	35	183	36
	CPU	36.32	4.56	1.69	0.88	5.26	0.92
	RES	9.85e-6	9.82e-06	9.68e-06	5.18e-6	5.73e-6	7.67e-6
13860	IT	1500	209	100	47	1154	33
	CPU	338.07	63.47	29.80	14.38	364.60	8.47
	RES	3.42e-5	9.50e-6	8.55e-06	8.84e-6	9.96e-6	7.82e-6
44820	IT	1500	385	64	40	1500	37
	CPU	925.67	534.15	104.58	56.23	1384.21	41.08
	RES	2.03e-4	9.65e-6	9.05e-6	7.65e-6	6.18e-5	9.26e-6
103954	IT	1500	512	82	44	1500	36
	CPU	1802.69	2141.16	442.31	250.35	2356.46	130.24
	RES	3.83e-4	9.76e-6	9.62e-6	7.56e-6	8.61e-4	7.50e-6

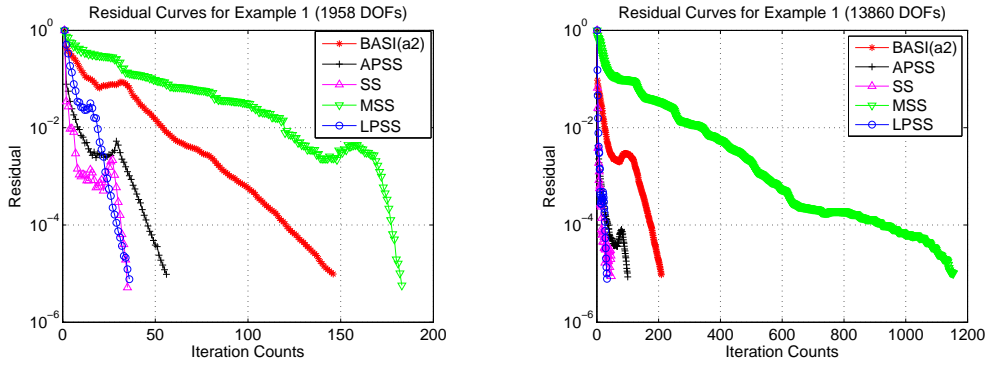


Figure 1: Example 4.1. Residual curves for various preconditioned GMRES methods. Left: DOF=1958. Right: DOF=13860.

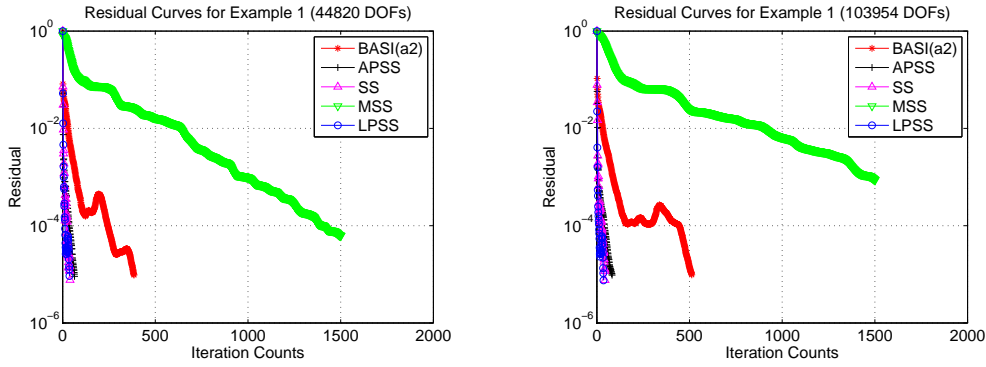


Figure 2: Example 4.1. Residual curves for various preconditioned GMRES methods. Left: DOF=44820. Right: DOF=103954.

iteration steps and CPU time. For larger DOFs, however, the LPSS preconditioned GMRES method performs much better than BASI(a2), APSS and SS preconditioned GMRES methods. In addition, the number of iteration steps in the LPSS preconditioned GMRES method is almost the same for each of the four DOF. Thus for Example 4.1, the LPSS preconditioner proposed here efficiently accelerate the GMRES method.

For other comparison of the preconditioned GMRES methods, in Figs. 1 and 2 we plot the residual curves of the methods for four successively refined meshes. The graphs demonstrate that the residual curves of SS and LPSS preconditioned GMRES decrease very sharply. Moreover, if DOF grows, the efficiency of LPSS preconditioner improves.

Example 4.2. We consider the hybrid formulation of the complete eddy current model (1.2), where the cube Ω with the edge 27cm contains two coaxial tori of square section with the edge 1cm and the radius 6.5cm. The upper torus is a coil, part of the insulator region Ω_I with the clockwise current density $\mathbf{J}_{e,I} = 10^6 A/m^2$, and the second torus is the conductor. Besides, $\mu = \mu_0 = 4\pi \times 10^{-7} H/m$ are the physical magnitudes, $\sigma = 10^7 S/m$ is the magnetic permeability of the air and $\omega = 2\pi \times 50 rad/s$ the angular frequency — cf. [38].

Now the first Betti number is one. In this case, the LPSS preconditioner \mathcal{P}_{LPSS} is defined in (3.8) and the BASI preconditioners considered in [5] are

$$\mathcal{P}_{BASI(b2)} = \frac{1}{2\alpha} \begin{bmatrix} \alpha I + M_C - iS_C & 0 & B_C^T \\ 0 & \alpha I + S_I + \tau B_I^T B_I & 0 \\ -B_C & 0 & \alpha I \end{bmatrix} \begin{bmatrix} \alpha I & -iD^T & 0 \\ -iD & \alpha I & 0 \\ 0 & 0 & \alpha I \end{bmatrix}.$$

The numerical performance of the two BASI schemes in [5] is almost the same. Therefore, we only compare the LPSS preconditioner with the BASI scheme (b2). Besides, we also consider the APSS preconditioner [36], the SS preconditioner [23] and the MSS preconditioner [47], which have the form

$$\begin{aligned} \mathcal{P}_{APSS} &= \frac{1}{2\alpha} \begin{bmatrix} \alpha I + M_C - iS_C & -iD^T & B_C^T \\ -iD & \alpha I & 0 \\ -B_C & 0 & \alpha I \end{bmatrix} \begin{bmatrix} \alpha I & 0 & 0 \\ 0 & \alpha I + S_I + \tau B_I^T B_I & 0 \\ 0 & 0 & \alpha I \end{bmatrix}, \\ \mathcal{P}_{SS} &= \frac{1}{2} \begin{bmatrix} \alpha I + M_C - iS_C & -iD^T & B_C^T \\ -iD & \alpha I + S_I + \tau B_I^T B_I & 0 \\ -B_C & 0 & \alpha I \end{bmatrix}, \\ \mathcal{P}_{MSS} &= \frac{1}{2} \begin{bmatrix} \alpha I + 2M_C & 0 & B_C^T \\ 0 & \alpha I + 2(S_I + \tau B_I^T B_I) & 0 \\ -B_C & 0 & \alpha I \end{bmatrix}. \end{aligned}$$

To discretise model (1.2), we now use two successively refined meshes and obtain two saddle point linear systems. For each problem, the order of discretisation matrices and the parameter τ are displayed in Table 4.

Table 4: Example 4.2. DOF and parameter τ .

DOF	n_1	n_2	m_1	τ
4763	580	4020	163	2.2528e8
34795	3416	30724	655	9.5772e8

For BASI(b2) preconditioner an experimentally found optimal parameters α is used — cf. [5, Table I]. For APSS, SS, MSS and LPSS preconditioners, parameter α is obtained by the practical algebraic estimation method mentioned in Example 4.1. The values of the parameters are listed in Table 5. Table 6 shows numerical results for GMRES and preconditioned GMRES methods incorporated with BASI(b2), APSS, SS, MSS and LPSS preconditioners. The residual curves of the preconditioned GMRES methods are plotted in Fig. 3. We note that LPSS preconditioned GMRES method requires the least iteration steps and the least CPU time. Moreover, it is found that the residual curves of the LPSS preconditioned GMRES have the fastest decrease.

Table 5: Example 4.2. Parameter α for various preconditioners.

DOF	BASI(b2)	APSS	SS	MSS	LPSS
4763	1.7e5	8.3e3	1.5e4	1.7e4	8.3e3
34795	1.9e5	7.3e3	1.3e4	1.5e4	7.3e3

Table 6: Example 4.2. Numerical results for preconditioned GMRES methods.

DOFs		I	BASI(b2)	APSS	SS	MSS	LPSS
4763	IT	39	39	11	14	11	9
	CPU	3.58	3.66	1.61	2.33	1.58	1.42
	RES	6.78e-6	6.84e-6	4.48e-6	2.97e-6	4.60e-6	2.45e-6
34795	IT	381	65	15	19	15	11
	CPU	91.28	82.46	33.11	45.90	35.99	30.93
	RES	9.73e-6	9.82e-6	7.29e-6	9.48e-6	7.15e-6	9.05e-6

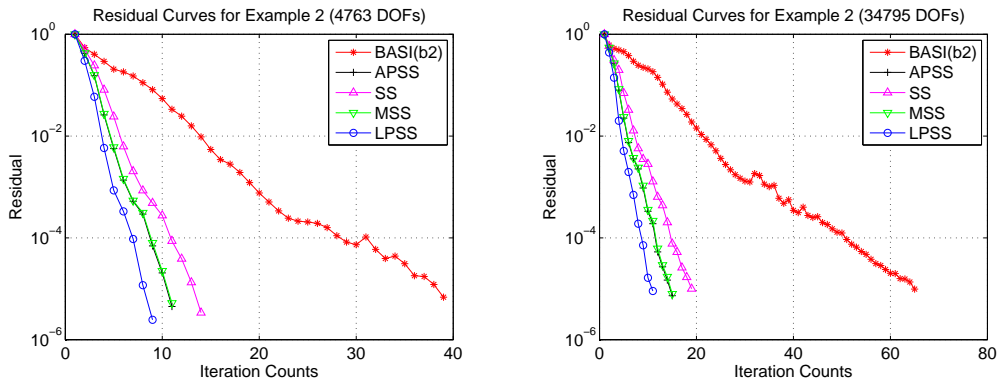


Figure 3: Example 4.2. Residual curves for different preconditioned GMRES methods. Left: DOF=4763. Right: DOF=34795.

5. Concluding Remarks

Using positive (semi)definite and skew-Hermitian splitting of a (1,1) block matrix and shift-splitting technique, we developed a new local positive (semi)definite shift-splitting (LPSS) preconditioner for a non-Hermitian saddle point problem. The convergence of the corresponding LPSS stationary iteration method and the spectral properties of corresponding preconditioned matrix are studied. The method is applied to non-Hermitian saddle point problems arising in time-harmonic eddy current models with different first Betti numbers. The LPSS preconditioner is easily implementable and numerical results show that it is more effective than BASI, APSS, SS, and MSS preconditioners.

Acknowledgments

The authors thank the Modern Education Technology Center of the Nantong University for assistance with numerical simulations. This work is supported by the National Natural Science Foundation of China (Nos. 11771225, 11771467, 11572210) and the Qing Lan Project of Excellent-Young-Backbone Teacher of Jiangsu Province.

References

- [1] M. Ardehshiry, H.S. Goughery and H.N. Pour, *New modified shift-splitting preconditioners for non-symmetric saddle point problems*, Arab. J. Math. <https://doi.org/10.1007/s40065-019-0256-6> (2019).
- [2] K.J. Arrow, L. Hurwicz and H. Uzawa, *Studies in linear and nonlinear programming*, Stanford University Press (1958).
- [3] C. Bacuta, B. McCracken and L. Shu, *Residual reduction algorithms for nonsymmetric saddle point problems*, J. Comput. Appl. Math. **235**, 1614-1628 (2011).
- [4] Z.Z. Bai, *Structured preconditioners for nonsingular matrices of block two-by-two structures*, Math. Comput. **75**, 791-815 (2006).
- [5] Z.Z. Bai, *Block alternating splitting implicit iteration methods for saddle-point problems from time-harmonic eddy current models*, Numer. Linear Algebra Appl. **19**, 914-936 (2012).
- [6] Z.Z. Bai, *Eigenvalue estimates for saddle point matrices of Hermitian and indefinite leading blocks*, J. Comput. Appl. Math. **237**, 295-306 (2013).
- [7] Z.Z. Bai, M. Benzi and F. Chen, *Modified HSS iteration methods for a class of complex symmetric linear systems*, Computing **87**, 93-111 (2010).
- [8] Z.Z. Bai, M. Benzi and F. Chen, *On preconditioned MHSS iteration methods for complex symmetric linear systems*, Numer. Algor. **56**, 297-317 (2011).
- [9] Z.Z. Bai and G.H. Golub, *Accelerated Hermitian and skew-Hermitian splitting iteration methods for saddle-point problems*, IMA J. Numer. Anal. **27**, 1-23 (2007).
- [10] Z.Z. Bai, G.H. Golub, L.Z. Lu and J.F. Yin, *Block triangular and skew-Hermitian splitting methods for positive-definite linear systems*, SIAM J. Sci. Comput. **26**, 844-863 (2005).
- [11] 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**, 603-626 (2003).
- [12] Z.Z. Bai and M.K. Ng, *On inexact preconditioners for nonsymmetric matrices*, SIAM J. Sci. Comput. **26**, 1710-1724 (2005).
- [13] Z.Z. Bai, B.N. Parlett and Z.Q. Wang, *On generalized successive overrelaxation methods for augmented linear systems*, Numer. Math. **102**, 1-38 (2005).
- [14] Z.Z. Bai, J.F. Yin and Y.F. Su, *A shift-splitting preconditioner for non-Hermitian positive definite matrices*, J. Comput. Math. **24**, 539-552 (2006).
- [15] T. Belytschko, Y.Y. Lu and L. Gu, *Element-free Galerkin methods*, Int. J. Numer. Meth. Engrg. **37**, 229-256 (1994).
- [16] M. Benzi and G.H. Golub, *A preconditioner for generalized saddle point problems*, SIAM J. Matrix Anal. Appl. **26**, 20-41 (2004).
- [17] M. Benzi, G.H. Golub and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer. **14**, 1-137 (2005).
- [18] L. Bergamaschi and Á. Martínez, *RMCP: Relaxed mixed constraint preconditioners for saddle point linear systems arising in geomechanics*, Comput. Methods Appl. Mech. Engrg. **221-222**, 54-62 (2012).

- [19] A. Bossavit, "Mixed" systems of algebraic equations in computational electromagnetics, *Compe*l **17**, 59-63 (1998).
- [20] J.H. Bramble, J.E. Pasciak and A.T. Vassilev, *Uzawa type algorithms for nonsymmetric saddle point problems*, *Math. Comput.* **69**, 667-689 (1999).
- [21] Y. Cao, J.L. Dong and Y.M. Wang, *A relaxed deteriorated PSS preconditioner for nonsymmetric saddle point problems from the steady Navier-Stokes equation*, *J. Comput. Appl. Math.* **273**, 41-60 (2015).
- [22] Y. Cao, J. Du and Q. Niu, *Shift-splitting preconditioners for saddle point problems*, *J. Comput. Appl. Math.* **272**, 239-250 (2014).
- [23] Y. Cao, S. Li and L.Q. Yao, *A class of generalized shift-splitting preconditioners for nonsymmetric saddle point problems*, *Appl. Math. Letters* **49**, 20-27 (2015).
- [24] Y. Cao and S.X. Miao, *On semi-convergence of the generalized shift-splitting iteration method for singular nonsymmetric saddle point problems*, *Comput. Math. Appl.* **71**, 1503-1511 (2016).
- [25] Y. Cao, Z.R. Ren and Q. Shi, *A simplified HSS preconditioner for generalized saddle point problems*, *BIT Numer. Math.* **56**, 423-439 (2016).
- [26] Y. Cao, A. Wang and Y.J. Chen, *A modified relaxed positive-semidefinite and skew-Hermitian splitting preconditioner for generalized saddle point problems*, *East Asian J. Appl. Math.* **7**, 192-210 (2017).
- [27] Y. Cao, L.Q. Yao, M.Q. Jiang and Q. Niu, *A relaxed HSS preconditioner for saddle point problems from meshfree discretization*, *J. Comput. Math.* **31**, 398-421 (2013).
- [28] Y. Cao and S.C. Yi, *A class of Uzawa-PSS iteration methods for nonsingular and singular saddle point problems*, *Appl. Math. Comput.* **275**, 41-49 (2016).
- [29] Q.Y. Dou, J.F. Yin and Z.Y. Liao, *A fast shift-splitting iteration method for nonsymmetric saddle point problems*, *East Asian J. Appl. Math.* **7**, 172-191 (2017).
- [30] H.C. Elman, D.J. Silvester and A.J. Wathen, *Finite elements and fast iterative solvers: with applications in incompressible fluid dynamics* (2nd edn), Oxford University Press, Oxford (2014).
- [31] T.T. Feng, G.L. Chen and X.P. Guo, *An accelerated SOR-like method for generalized saddle point problems*, *East Asian J. Appl. Math.* **8**, 70-81 (2018).
- [32] R.A. Horn and C.R. Johnson, *Matrix analysis*, Cambridge University Press (1985).
- [33] M.Q. Jiang and Y. Cao, *On local Hermitian and skew-Hermitian splitting iteration methods for generalized saddle point problems*, *J. Comput. Appl. Math.* **231**, 973-982 (2009).
- [34] J. Nocedal and S.J. Wright, *Numerical optimization*, Springer (1999).
- [35] J. Pestana and A.J. Wathen, *Natural preconditioning and iterative methods for saddle point problems*, *SIAM Rev.* **57**, 71-91 (2015).
- [36] Z.R. Ren and Y. Cao, *An alternating positive-semidefinite splitting preconditioner for saddle point problems from time-harmonic eddy current models*, *IMA J. Numer. Anal.* **36**, 922-946 (2016).
- [37] Z.R. Ren, Y. Cao and Q. Niu, *Spectral analysis of generalized shift-splitting preconditioned saddle point problem*, *J. Comput. Appl. Math.* **311**, 539-550 (2017).
- [38] A. Rodríguez and R. Hernández, *Iterative methods for the saddle-point problem arising from the H_C/E_I formulation of the eddy current problem*, *SIAM J. Sci. Comput.* **31**, 3155-3178 (2009).
- [39] A. Rodríguez, R. Hiptmair and A. Valli, *A hybrid formulation of eddy current problems*, *Numer. Methods Partial Differential Equations* **21**, 742-763 (2005).
- [40] Y. Saad, *Iterative methods for sparse linear systems*, SIAM (2003).
- [41] D.K. Salkuyeh, M. Masoudi and D. Hezari, *On the generalized shift-splitting preconditioner for saddle point problems*, *Appl. Math. Letters* **48**, 55-61 (2015).
- [42] D.K. Salkuyeh and M. Rahimian, *A modification of the generalized shift-splitting method for singular saddle point problems*, *Comput. Math. Appl.* **74**, 2940-2949 (2017).
- [43] Q.Q. Shen and Q. Shi, *Generalized shift-splitting preconditioners for nonsingular and singular*

- generalized saddle point problems*, Comput. Math. Appl. **72**, 632-641 (2016).
- [44] Q. Shi, Q.Q. Shen and L.Q. Yao, *Eigenvalue bounds of the shift-splitting preconditioned singular nonsymmetric saddle-point matrices*, J. Inequal. Appl. **2016**, 256 (2016).
- [45] A.L. Yang and Y.J. Wu, *The Uzawa-HSS method for saddle-point problems*, Appl. Math. Letters **38**, 38-42 (2014).
- [46] D.M. Young, *Iterative solution of large linear systems*, Academic Press (1971).
- [47] S.W. Zhou, A.L. Yang, Y. Dou and Y.J. Wu, *The modified shift-splitting preconditioners for non-symmetric saddle-point problems*, Appl. Math. Letters **59**, 109-114 (2016).