REVISTA DE LA UNIÓN MATEMÁTICA ARGENTINA
Vol. 68, No. 1, 2025, Pages 131–144
Published online: April 3, 2025
https://doi.org/10.33044/revuma.3672

AN IMPROVED LOPSIDED SHIFT-SPLITTING PRECONDITIONER FOR THREE-BY-THREE BLOCK SADDLE POINT PROBLEMS

JUN LI AND XIANGTUAN XIONG

Abstract. We consider an improved lopsided shift-splitting (ILSS) preconditioner for solving three-by-three block saddle point problems. This method enhances the work of Zhang et al. [Comput. Appl. Math. 41 (2022), 261]. We prove that the iteration method produced by the ILSS preconditioner is unconditionally convergent. Additionally, we show that all eigenvalues of the ILSS preconditioned matrix are real, with non-unit eigenvalues located in a positive interval. Numerical experiments demonstrate the effectiveness of the ILSS preconditioner.

1. Introduction

In this paper, we focus on the following three-by-three block system of linear equations:

$$\hat{\mathcal{A}}\mathfrak{u} = \begin{pmatrix} A & B^T & 0 \\ B & 0 & C^T \\ 0 & C & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} f \\ g \\ h \end{pmatrix} \equiv \hat{b}, \tag{1.1}$$

where $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite, $B \in \mathbb{R}^{m\times n}$ and $C \in \mathbb{R}^{p\times m}$ are of full row rank, $f \in \mathbb{R}^n$, $g \in \mathbb{R}^m$ and $h \in \mathbb{R}^p$ are given vectors. With the above conditions, the coefficient matrix of the linear system (1.1) is non-singular [9, 10, 8]. The linear system (1.1) arises in many scientific computing and engineering applications such as solving quadratic programs [6], least squares problems [11], the Picard iteration method for a class of mixed finite element scheme for stationary magnetohydrodynamics models [7], the finite element method to solve the time-dependent Maxwell equations having discontinuous coefficients in polyhedral domains with Lipschitz boundary [5, 13] and so on.

2020 *Mathematics Subject Classification.* Primary 65F08; Secondary 65F10, 65F50.

Key words and phrases. three-by-three block saddle point, matrix splitting, preconditioning, eigenvalue distribution.

This work was supported by the Foundation for Distinguished Young Scholars of Gansu Province (20JR5RA540) and the National Natural Science Foundation of China (11661072).

It is obvious that (1.1) can equivalently be rewritten in the following form:

$$\mathcal{A}\mathfrak{u} = \begin{pmatrix} A & B^T & 0 \\ -B & 0 & -C^T \\ 0 & C & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} f \\ -g \\ h \end{pmatrix} \equiv b, \tag{1.2}$$

which facilitates the possibility to apply classical iteration solution methods. To solve (1.2), many numerical methods have been considered. Huang et al. [10] proposed an exact block diagonal preconditioner, which has the following structure:

$$P_{\mathrm{BD}} = \begin{pmatrix} A & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & CS^{-1}C^T \end{pmatrix}, \tag{1.3}$$

where $S = BA^{-1}B^T$. The inexact versions of the block preconditioner were also studied. Then, Cao [4] established the shift splitting (SS) and the relaxed shift splitting (RSS) preconditioner:

$$P_{\mathrm{SS}} = \frac{1}{2}\begin{pmatrix} \alpha I + A & B^T & 0 \\ -B & \alpha I & -C^T \\ 0 & C & \alpha I \end{pmatrix}, \quad P_{\mathrm{RSS}} = \frac{1}{2}\begin{pmatrix} A & B^T & 0 \\ -B & \alpha I & -C^T \\ 0 & C & \alpha I \end{pmatrix}. \tag{1.4}$$

Besides, a generalized shift-splitting (GSS) preconditioner was considered in [15] on the basis of the SS preconditioner. Furthermore, Zhang et al. [18] proposed a lopsided shift-splitting (LSS) preconditioner for the saddle point problems (1.2):

$$P_{\mathrm{LSS}} = \frac{1}{2}\begin{pmatrix} \alpha I + A & B^T & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & \beta I \end{pmatrix}. \tag{1.5}$$

They also proved that the LSS prconditioner is superior to the SS, RSS and GSS preconditioners by numerical results.

To further enhance the preconditioning effect of the block diagonal preconditioner (1.3), Xie et al. [17] considered three efficient block preconditioners:

$$P_1 = \begin{pmatrix} A & 0 & 0 \\ B & -S & C^T \\ 0 & 0 & -CS^{-1}C^T \end{pmatrix}, \quad P_2 = \begin{pmatrix} A & 0 & 0 \\ B & -S & C^T \\ 0 & 0 & CS^{-1}C^T \end{pmatrix},$$
$$P_3 = \begin{pmatrix} A & B^T & 0 \\ B & -S & 0 \\ 0 & 0 & -CS^{-1}C^T \end{pmatrix} \tag{1.6}$$

for the linear system (1.1). They showed that the above three preconditioners can significantly improve the convergence speed of the GMRES method, and preconditioner P_3 outperforms the other two preconditioners in terms of both required CPU time and number of iterations for convergence. Later, Aslani et al. [3] presented a new matrix splitting and deduced a preconditioner for solving the saddle point system (1.2). Abdolmaleki et al. [1] proposed a block three-by-three diagonal preconditioner for (1.2). Wang et al. [16] considered an exact parameterized block SPD preconditioner and its inexact version for the saddle point problems (1.2).

The alternating positive semi-definite splitting iteration methods and corresponding preconditioners were also proposed for solving three-by-three block saddle point problems [2, 14].

In this paper, in order to improve the preconditioning effect of the LSS preconditioner, we propose an improved LSS (ILSS) preconditioner for solving the linear system (1.2). In the solving process of the residual equations, the coefficient matrices of the linear subsystems corresponding to the ILSS preconditioner are simpler than that of the linear subsystems corresponding to the LSS preconditioner, so it has an advantage in solving time. Theoretically, we show the unconditional convergence of the iteration method produced by the ILSS preconditioner, and the eigenvalue distribution of the ILSS preconditioned matrix will also be discussed. Numerical experiments reveal that the ILSS preconditioner is easy to implement and takes less time compared with some existing preconditioners.

2. The improved lopsided shift-splitting preconditioner

The main purpose of this section is to establish a new preconditioner and discuss the convergence of the iteration method it produces.

Firstly, let us recall the lopsided shift-splitting (LSS) preconditioner. Zhang et al. [18] proposed in 2022 the LSS preconditioner (1.5) for solving the problems (1.2). When it is used to accelerate Krylov subspace methods (such as the GMRES method), the following generalized residual equation needs to be solved:

$$P_{\mathrm{LSS}} z = r, \tag{2.1}$$

where $r = (r_1; r_2; r_3)$ and $z = (z_1; z_2; z_3)$ are a given residual vector and the current vector, respectively, and $r_1,\ z_1 \in \mathbb{R}^n,\ r_2,\ z_2 \in \mathbb{R}^m,\ r_3,\ z_3 \in \mathbb{R}^p$. Thus we have the following algorithmic implementation to solve the linear system (2.1) in actual computation.

LSS algorithm. For a given residual vector $r = [r_1^T, r_2^T, r_3^T]^T$, the current vector $z = [z_1^T, z_2^T, z_3^T]^T$. (2.1) can be computed according to the following procedures:

(1) Solve $(\beta I + \frac{1}{\alpha} CC^T) z_3 = 2r_3 - \frac{2}{\alpha} C r_2$.
(2) Solve $z_2 = \frac{1}{\alpha}(C^T z_3 + 2r_2)$.
(3) Solve $(\alpha I + A) z_1 = 2r_1 - B^T z_2$.

We see from the above algorithm that only two subsystems need to be solved. In order to further improve the preconditioning effect of the LSS preconditioner, we can simplify the coefficient matrices $\beta I + \frac{1}{\alpha} CC^T$ and $\alpha I + A$ in the above algorithm as CC^T and A, respectively, and an extra preconditioner is produced:

$$P = \frac{1}{2} \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix}.$$

Because the per-factor $\frac{1}{2}$ has no effect on the preconditioned linear system in actual computations, it will be neglected and an improved LSS (ILSS) preconditioner is

established:

$$P_{\mathrm{ILSS}} = \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix}.$$

Therefore, for the preconditioner P_{ILSS}, we can describe the implementing process as the following algorithm:

ILSS algorithm. For a given residual vector $r = [r_1^T, r_2^T, r_3^T]^T$, the current vector $z = [z_1^T, z_2^T, z_3^T]^T$, residual equation corresponding to P_{ILSS} can be computed according to the following procedures:

(1) Solve $Az_1 = r_1$.
(2) Solve $CC^T z_3 = \alpha r_3 - Cr_2$.
(3) Solve $z_2 = \frac{1}{\alpha}(C^T z_3 + r_2)$.

By comparing the LSS and ILSS algorithms, we see that the preconditioner P_{ILSS} consumes less computational cost than the preconditioner P_{LSS}. Due to the symmetric positive definiteness of the coefficient matrices in linear subsystems, we can choose the Cholesky decomposition or the PCG method to solve linear subsystems at each step for the ILSS algorithm.

In fact, $\mathcal{A}$ admits the following splitting:

$$\mathcal{A} = P_{\mathrm{ILSS}} - Q_{\mathrm{ILSS}} = \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix} - \begin{pmatrix} 0 & -B^T & 0 \\ B & \alpha I & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

which results in a matrix splitting iteration method, called the ILSS iteration method, for solving the linear system (1.2).

ILSS iteration method. Let $\alpha > 0$ be a given parameter and $(x_0^T, y_0^T, z_0^T)^T$ be the initial guess vector. For $k = 0, 1, 2, \ldots$ until a certain stopping criterion is satisfied, compute

$$\begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix} \begin{pmatrix} x_{k+1} \\ y_{k+1} \\ z_{k+1} \end{pmatrix} = \begin{pmatrix} 0 & -B^T & 0 \\ B & \alpha I & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} + \begin{pmatrix} f \\ -g \\ h \end{pmatrix}.$$

It is easy to rewrite the ILSS iteration method in the following fixed-point form:

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \\ z_{k+1} \end{pmatrix} = \Gamma_\alpha \begin{pmatrix} x_k \\ y_k \\ z_k \end{pmatrix} + \Upsilon, \tag{2.2}$$

where

$$\Gamma_\alpha = \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 & -B^T & 0 \\ B & \alpha I & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

is the iteration matrix and

$$\Upsilon = \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix}^{-1} \begin{pmatrix} f \\ -g \\ h \end{pmatrix}.$$

It is well known that the fixed-point iteration method (2.2) converges to the exact solution $x_* = \mathcal{A}^{-1}b$ for arbitrary initial guess if the spectral radius $\rho(\Gamma_\alpha)$ is less than 1.

Next, we will discuss the convergence of the ILSS iteration method.

Theorem 2.1. *Assume that $A \in \mathbb{R}^{n\times n}$ is a symmetric positive definite (SPD) matrix, $B \in \mathbb{R}^{m\times n}$ and $C \in \mathbb{R}^{p\times m}$ are of full row rank, and α is a given positive constant. Then the ILSS iteration method is unconditionally convergent.*

Proof. Let λ be an eigenvalue of the iteration matrix Γ_α and $[u; v; w]$ be the corresponding eigenvector. If $\lambda = 0$, the ILSS iteration method is convergent. Next, we consider the case of $\lambda \neq 0$. It follows, according to the relationships between eigenvalues and eigenvectors, that

$$\begin{pmatrix} 0 & -B^T & 0 \\ B & \alpha I & 0 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix} = \lambda \begin{pmatrix} A & 0 & 0 \\ 0 & \alpha I & -C^T \\ 0 & C & 0 \end{pmatrix} \begin{pmatrix} u \\ v \\ w \end{pmatrix}. \tag{2.3}$$

Eq.(2.3) can equivalently be reformulated into

$$\begin{cases} \lambda Au + B^T v = 0, \\ Bu + \alpha(1-\lambda)v + \lambda C^T w = 0, \\ Cv = 0. \end{cases} \tag{2.4}$$

It is obvious that both u and v are not zero vectors, otherwise $(u; v; w) = (0; 0; 0)$, which is a contradiction. For simplicity, we can assume that $\|v\| = 1$.

If $\lambda = 1$,

$$\begin{cases} Au + B^T v = 0, \\ Bu + C^T w = 0, \\ Cv = 0. \end{cases} \tag{2.5}$$

Multiplying the second equation in (2.5) from the left by v^*, combining with the first and third equations in (2.5) gives $u^*Au = 0$, i.e., $u = 0$; furthermore, it follows that $v = 0$ and $w = 0$, a contradiction. Thus $\lambda \neq 1$.

Similarly, multiplying the second equation in (2.4) from the left by v^*, combining with the first and third equations in (2.4), we have

$$-\bar{\lambda}u^*Au + \alpha(1-\lambda) = 0,$$

i.e.,

$$\alpha\lambda + u^*Au\bar{\lambda} = \alpha,$$

where $\bar{\lambda}$ denotes the conjugate number of λ. Because the matrix A is symmetric positive definite and α is a positive constant, $\bar{\lambda} = \lambda$. Thus it holds that

$$\lambda = \frac{\alpha}{\alpha + u^* A u} < 1.$$

In this case, the ILSS iteration method is convergent.

In conclusion, the ILSS iteration method is unconditionally convergent. □

3. Spectral properties of the preconditioned matrix

In this section, we derive the eigenvalue distribution of the preconditioned matrix $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$.

Theorem 3.1. *Assume that the conditions in Theorem* 2.1 *are satisfied. Then all the eigenvalues of the preconditioned matrix* $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ *are real numbers and*

$$\mathrm{sp}\left(P_{\mathrm{ILSS}}^{-1}\mathcal{A}\right) \subseteq \left(0, \frac{a}{\alpha + a}\right] \cup \{1\},$$

where $\mathrm{sp}(\cdot)$ *represents the spectrum of the corresponding matrix, and* $\lambda_{\min}(A) \leq a \leq \lambda_{\max}(A)$.

Proof. Let θ be the eigenvalue of the preconditioned matrix $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$. Since $P_{\mathrm{ILSS}}^{-1}\mathcal{A} = I - P_{\mathrm{ILSS}}^{-1}Q_{\mathrm{ILSS}}$, it holds from Theorem 2.1 that

$$\theta = 1 - \lambda = 1 - \frac{\alpha}{\alpha + u^* A u} = \frac{u^* A u}{\alpha + u^* A u}. \tag{3.1}$$

If $\lambda = 0$, the eigenvalue of $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ is 1. It is obvious from (3.1) that θ is a real number if $\lambda \neq 0$. Let $a = u^* A u$; we have

$$0 < \theta \leq \frac{a}{\alpha + a}.$$

This completes the proof. □

Corollary 3.2. *Assume that the conditions in Theorem* 2.1 *hold. Then the eigenvalues of the ILSS preconditioned matrix* $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ *are either clustered at* 1 *if the parameter* α *is close to* 0_+ *or clustered at points* 0 *and* 1 *if* $\alpha \to +\infty$.

Proof. The conclusion is obvious by the expression of (3.1). □

As a matter of fact, the convergence rate of the preconditioned Krylov subspace method is determined not only by the eigenvalue distribution of the preconditioned matrix but also by its eigenvector distribution. The eigenvector distribution of the preconditioned matrix $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ is presented in the following theorem.

Theorem 3.3. *Assume that the conditions of Theorem* 2.1 *hold. Then, the preconditioned matrix* $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ *has* $n + i$ *linearly independent eigenvectors, where* $0 \leq i \leq m + p$. *These eigenvectors are classified as follows:*

(1) *There exist* n *eigenvectors of the form* $[u_\ell; 0; w_\ell]$ *for* $\ell = 1, 2, \ldots, n$, *corresponding to the eigenvalue* 1. *Here,* w_ℓ $(\ell = 1, 2, \ldots, n)$ *are arbitrary linearly independent vectors, and* $u_\ell \in N(B)$, *where* $N(\cdot)$ *denotes the null space of the corresponding matrix.*

(2) *There exist* i $(0 \leq i \leq m+p)$ *eigenvectors of the form* $[u_\ell; v_\ell; w_\ell]$ *for* $1 \leq \ell \leq i$, *corresponding to the eigenvalues* $\theta_\ell \neq 1$. *Here,* v_ℓ *is a nonzero vector in* $N(C)$, *while*

$$u_\ell = \frac{1}{\theta_\ell - 1} A^{-1} B^T v_\ell$$

and

$$w_\ell = \frac{1}{(\theta_\ell - 1)^2} (CC^T)^{-1} CBA^{-1} B^T v_\ell.$$

The proof of the above theorem is similar to that of [12, Theorem 2], and thus we omit it. Additionally, we provide a precise elaboration on the degree of the minimal polynomial of the ILSS preconditioned matrix $P_{\text{ILSS}}^{-1}\mathcal{A}$.

Theorem 3.4. *Assume that the conditions of Theorem* 2.1 *hold. Then the dimension of the Krylov subspace* $\mathcal{K}(P_{\text{ILSS}}^{-1}\mathcal{A}, b)$ *is at most* $m + p + 1$.

4. Numerical experiments

In this section, we give two examples which demonstrate the effectiveness of the proposed preconditioner. In our implementations, the initial guess is chosen to be the zero vector, the exact solution is the vector with all its element being ones and $b = \mathcal{A} \cdot \mathbf{1}$. The iteration is terminated if the current iteration satisfies $\text{RES} = \|b - \mathcal{A}u^{(k)}\|_2 / \|b\|_2 \leq 10^{-6}$ or if the prescribed iteration number $k_{\max} = 1500$ is exceeded. Besides, the relative errors of the computed approximations, denoted as ERR, i.e.,

$$\text{ERR} = \frac{\|u_k - u_*\|_2}{\|u_*\|_2}$$

are reported with u_* the initially known exact solution. All experiments were performed in MATLAB R2017a on a Windows 10 system with an Intel Core processor and 8 GB of RAM.

To compare with the ILSS preconditioner, we also tested the BD preconditioner (1.3), SS preconditioner (1.4), LSS preconditioner (1.5), and block preconditioner P_3 (1.6) for solving the three-by-three block saddle point problems (1.2). When preconditioners are applied to the GMRES method, the coefficient matrices of the corresponding linear subsystems are SPD. Therefore, we can use sparse Cholesky decomposition to solve them. All of them are used as the left preconditioned GMRES iteration method to solve the three-by-three block saddle point problems.

Example 4.1 ([4, 14])**.** Consider the three-by-three block saddle point problems (1.2), in which

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

$$B = \begin{pmatrix} I \otimes F & F \otimes I \end{pmatrix} \in \mathbb{R}^{p^2 \times 2p^2}$$

$$C = E \otimes F \in \mathbb{R}^{p^2 \times p^2},$$

where

$$T = \frac{1}{h^2}\,\mathrm{tridiag}(-1, 2, -1) \in \mathbb{R}^{p\times p},$$
$$F = \frac{1}{h}\,\mathrm{tridiag}(0, 1, -1) \in \mathbb{R}^{p\times p},$$
$$E = \mathrm{diag}(1, p+1, \ldots, p^2-p+1),$$

$\otimes$ means the Kronecker product symbol, and $h = \frac{1}{p+1}$.

TABLE 1. The CPU times of the ILSS preconditioned GMRES method with various values of the parameter α in different dimensions for Example 4.1.

$p \setminus \alpha$	1	10^{-1}	10^{-2}	10^{-3}	10^{-4}	10^{-5}
16	0.0195	0.0119	0.0034	0.0019	**0.0015**	0.0022
32	0.0284	0.0178	0.0085	0.0068	**0.0061**	50.7946
48	0.0526	0.0481	0.0318	**0.0302**	0.0376	94.4258
56	0.0372	0.0344	0.0308	**0.0293**	123.5037	-

Example 4.2 ([4, 14])**.** We consider the three-by-three saddle point problems (1.2), in which

$$A = \mathrm{diag}(2W^TW + D_1, D_2, D_3) \in \mathbb{R}^{n\times n}$$

is a block diagonal matrix,

$$B = [E, -I_{2\tilde{p}}, I_{2\tilde{p}}] \in \mathbb{R}^{m\times n} \quad \text{and} \quad C = E^T \in \mathbb{R}^{l\times m}$$

are both full row rank matrices, where $\tilde{p} = p^2$, $\hat{p} = p(p+1)$; $W = (w_{ij}) \in \mathbb{R}^{\hat{p}\times\hat{p}}$ with $w_{ij} = e^{-2((i/3)^2+(j/3)^2)}$; $D_1 = I_{\hat{p}}$ is an identity matrix; $D_i = \mathrm{diag}(d_j^{(i)}) \in \mathbb{R}^{2\tilde{p}\times 2\tilde{p}}$, $i = 2, 3$, are diagonal matrices, with

$$d_j^2 = \begin{cases} 1 & \text{for } 1 \le j \le \tilde{p}, \\ 10^{-5}(j-\tilde{p})^2 & \text{for } \tilde{p}+1 \le j \le 2\tilde{p}, \end{cases}$$
$$d_j^3 = 10^{-5}(j+\tilde{p})^2 \quad \text{for } 1 \le j \le 2\tilde{p};$$

and

$$E = \begin{pmatrix} \hat{E} \otimes I_p \\ I_p \otimes \hat{E} \end{pmatrix}, \quad \hat{E} = \begin{pmatrix} 2 & -1 & & & \\ & 2 & -1 & & \\ & & \ddots & \ddots & \\ & & & 2 & -1 \end{pmatrix} \in \mathbb{R}^{p\times(p+1)}.$$

The experiment problems are formed by setting different values of p. For existing preconditioners, in order to make the preconditioners closer to the coefficient

TABLE 2. Numerical results of preconditioned GMRES iteration methods for Example 4.1.

Pre.	I	P_{BD}	P_3	P_{SS}	P_{LSS}	P_{ILSS}
$p = 16$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-4}, -)$
IT	865	4	3	2	3	3
RES	8.7e-07	1.5e-10	1.3e-10	7.6e-07	2.3e-09	2.0e-08
ERR	2.3e-06	4.1e-11	3.5e-11	4.9e-06	3.5e-10	1.9e-09
CPU	3.1177	0.0575	0.0213	0.0336	0.0129	0.0103
$p = 32$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-4}, -)$
IT	–	4	3	2	2	3
RES	–	1.2e-08	1.4e-08	3.4e-07	9.6e-07	2.4e-07
ERR	–	3.0e-09	3.6e-09	3.5e-06	1.1e-05	1.5e-08
CPU	–	0.5097	0.3952	0.1618	0.0204	0.0186
$p = 48$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-3}, -)$
IT	–	4	3	2	2	3
RES	–	7.7e-07	1.3e-09	2.1e-07	6.5e-07	1.0e-07
ERR	–	1.8e-07	3.1e-10	2.8e-06	9.7e-06	5.1e-09
CPU	–	4.7037	3.6303	0.8774	0.0315	0.0296
$p = 56$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-3}, -)$
IT	–	4	3	2	2	3
RES	–	9.9e-08	3.2e-07	1.7e-07	5.8e-07	1.7e-07
ERR	–	2.3e-08	7.5e-08	2.6e-06	9.1e-06	8.3e-09
CPU	–	11.1635	9.5971	1.7265	0.0403	0.0379
$p = 64$						
(α, β)	–	–		$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-2}, -)$
IT	–	5	3	2	2	3
RES	–	4.9e-07	6.6e-07	1.5e-07	5.4e-07	2.4e-08
ERR	–	1.1e-07	1.5e-07	2.4e-06	8.6e-06	1.1e-09
CPU	–	28.4426	25.4994	3.1639	0.0582	0.0432
$p = 80$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(10^{-3}, 10^{-6})$	$(10^{-2}, -)$
IT	–	6	5	2	2	3
RES	–	2.5e-07	5.7e-10	1.3e-07	6.9e-07	5.6e-08
ERR	–	5.8e-08	1.1e-10	2.2e-06	7.9e-06	2.2e-09
CPU	–	54.2340	48.7028	9.4477	0.1443	0.1192

matrix $\mathcal{A}$, we use the parameter selection method in [4, 18]. For the ILSS preconditioner, we know by Corollary 3.2 that the parameter α should be as small or big as possible to make the spectrum of the preconditioned matrix more clustered,

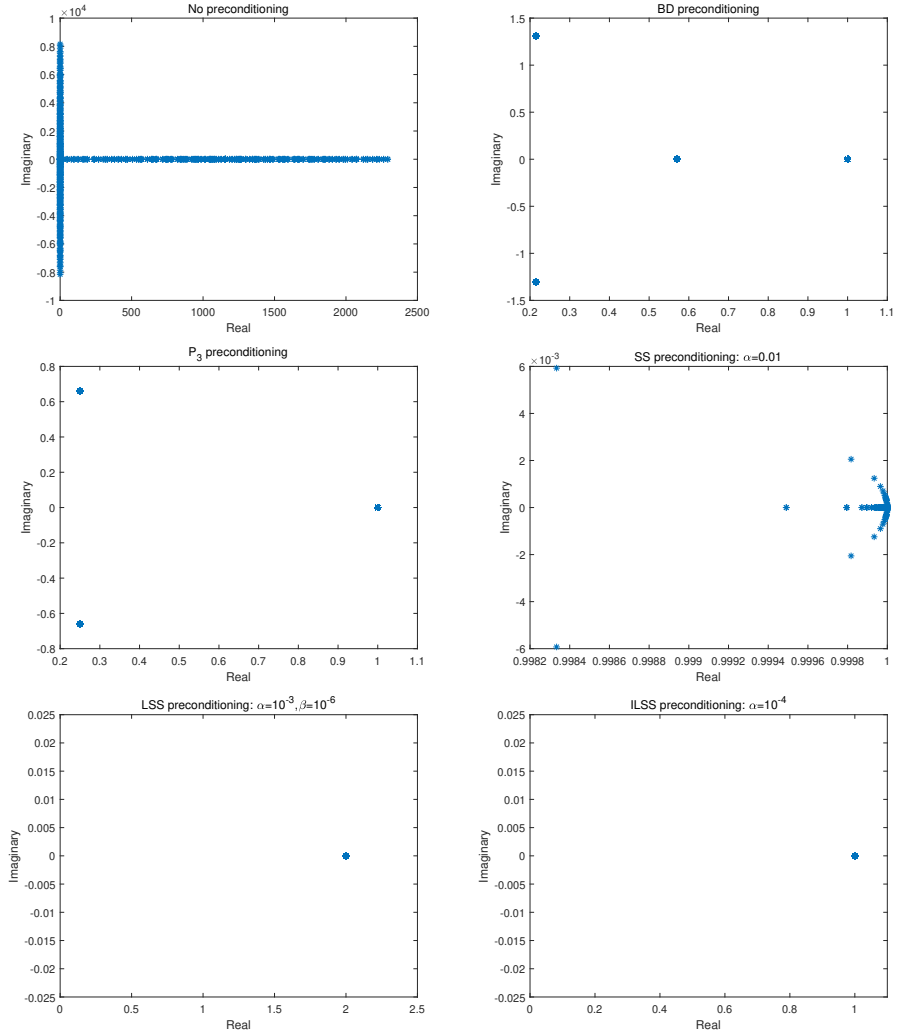

FIGURE 1. The eigenvalue distribution of the preconditioned matrices with $p = 16$ for Example 4.1.

and then accelerate the convergence speed of the GMRES method. Therefore, we list the CPU time of the preconditioned GMRES method with various α in different dimensions in Table 1. We find by Table 1 that when α is from 1 to 10^{-5}, the CPU times of the corresponding ILSS preconditioned GMRES method first decreases and then increases. Based on this phenomenon, we choose $\alpha = 10^{-4}$ when $p = 16, 32$; if $p = 48, 56$, $\alpha = 10^{-3}$, and $\alpha = 10^{-2}$ as $p = 64,\ 80$ for Example 4.1. For Example 4.2, we give the tested optimal parameters α for the ILSS preconditioned GMRES method.

TABLE 3. Numerical results of preconditioned GMRES iteration methods for Example 4.2.

Pre.	I	P_{BD}	P_3	P_{SS}	P_{LSS}	P_{ILSS}
$p = 16$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(0.6, 10^{-2})$	$(10^{7}, -)$
IT	1207	6	3	4	22	40
RES	9.5e-07	7.1e-12	1.5e-11	7.4e-08	8.7e-07	8.7e-07
ERR	1.1e-05	1.9e-11	4.1e-11	1.4e-07	8.8e-06	2.3e-06
CPU	1.0345	0.4359	0.3725	0.2705	0.2587	0.2451
$p = 32$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(0.5, 0.1)$	$(10^{8}, -)$
IT	1452	6	3	3	16	22
RES	9.8e-07	2.1e-14	4.5e-14	4.9e-07	9.2e-07	9.5e-07
ERR	4.5e-05	5.5e-13	1.2e-12	1.2e-06	2.9e-05	1.8e-05
CPU	23.6692	10.8788	8.6491	5.5974	3.9426	3.4267
$p = 48$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(0.5, 0.1)$	$(10^{8}, -)$
IT	–	6	3	3	17	16
RES	–	1.5e-14	9.1e-15	5.9e-07	1.5e-07	7.5e-07
ERR	–	1.8e-12	1.2e-12	1.5e-06	4.5e-05	7.3e-05
CPU	–	119.0142	88.0379	55.1203	24.5532	11.0027
$p = 56$						
(α, β)	–	–	–	$(10^{-2}, -)$	$(0.5, 0.1)$	$(10^{8}, -)$
IT	–	6	3	3	17	16
RES	–	5.6e-15	3.2e-15	5.8e-07	3.2e-07	2.9e-07
ERR	–	1.3e-12	7.4e-13	1.6e-06	5.5e-05	5.3e-05
CPU	–	289.4107	215.7352	159.5430	61.4487	23.5894

In Tables 2 and 3, we list the number of iteration steps IT, the residuals RES, the relative errors ERR and the corresponding CPU times of the preconditioned GMRES iteration methods with I (i.e., no preconditioning), P_{BD}, P_3, P_{SS}, P_{LSS} and P_{ILSS} with different p for Examples 4.1 and 4.2.

From these results in Table 2, we find that the unpreconditioned GMRES method converges very slowly and it is not convergent if $p \geq 32$. Although the P_{BD} and P_3 preconditioned GMRES method has fewer iteration steps, its CPU time increases greatly as the problem size grows. Besides, when comparing SS, LSS and ILSS preconditioners, it is easy to see that the iteration steps of the SS and LSS preconditioned GMRES methods are equal, but the CPU time of SS preconditioner increases greatly as the dimension increases. Although the iteration steps of the ILSS preconditioned GMRES method are more than those of the SS and LSS preconditioned GMRES method, the ILSS method is the least time-consuming, which also proves that our comparison conclusion from the algorithm

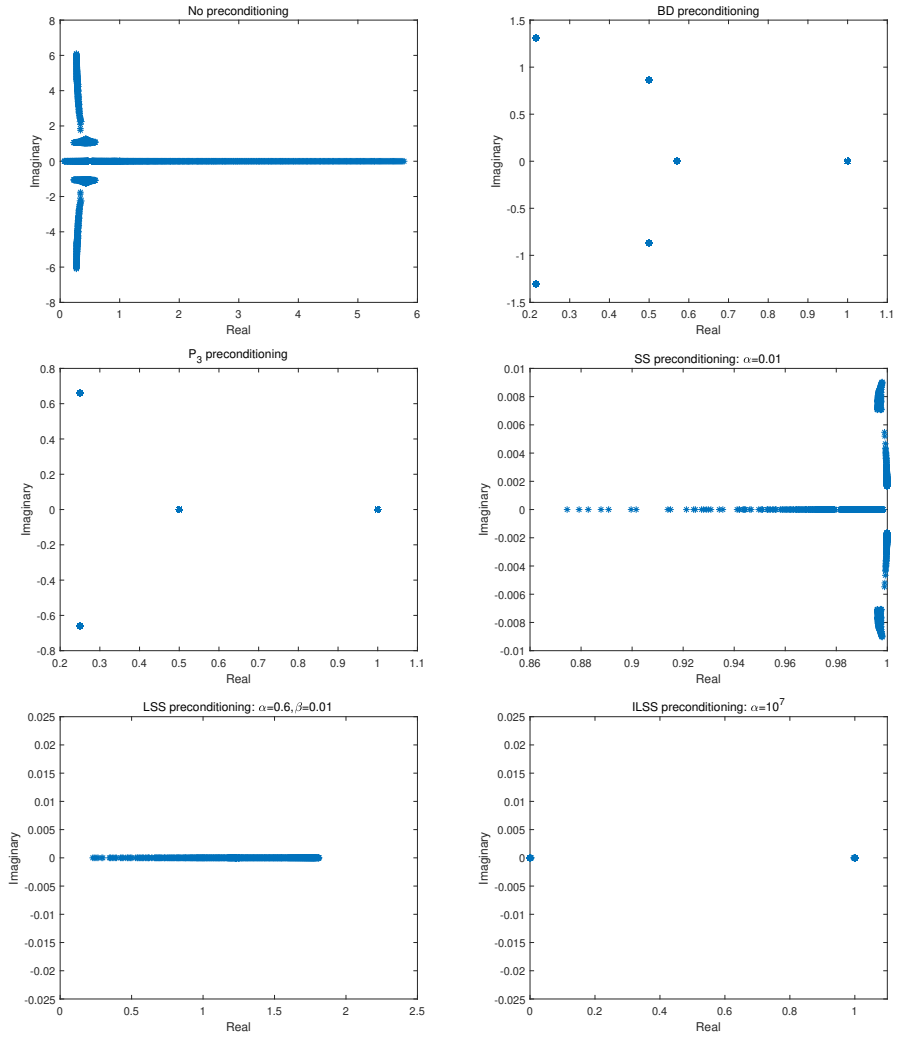

FIGURE 2. The eigenvalue distribution of the preconditioned matrices with $p = 16$ for Example 4.2.

is correct. In addition, the approximate solution obtained by the GMRES method preconditioned by P_{ILSS} is closer to the exact solution than that obtained by the GMRES method preconditioned by P_{SS} and P_{LSS}. This fact can be confirmed by the value of ERR in Table 2. From Table 3, it is easy to see that, although the P_{BD}, P_3 and P_{SS} preconditioned GMRES methods have the fewest iteration steps, their CPU times increase greatly as p increases. On the contrary, the LSS and ILSS preconditioned GMRES methods have more iteration steps, but they take less time

than the other three preconditioners. Besides, when comparing the LSS preconditioner with the ILSS preconditioner, we find that the iteration steps of the ILSS preconditioned GMRES method are more than those of the LSS preconditioned GMRES method when $p \leq 32$; if $p > 32$, it is the other way around. In each case, the CPU time of the ILSS method is always less than that of the LSS method, and the accuracy of the iterative solutions obtained by the two methods is essentially the same. These facts indicate that the parameter we select is effective and that the ILSS preconditioner is the optimal preconditioner for solving the three-by-three block saddle point problems (1.2).

In Figures 1 and 2, we plot the eigenvalue distribution of the original coefficient matrix and the P_{BD}, P_3, P_{SS}, P_{LSS} and P_{ILSS} preconditioned matrices with $p = 16$ for Examples 4.1 and 4.2. It is easy to observe from these figures that all the preconditioners greatly improve the eigenvalue distribution of the original coefficient matrix. In particular, the spectrum of the ILSS preconditioned matrix is clustered at point 1 when $\alpha = 10^{-4}$ in Figure 1. However, the spectrum of the ILSS preconditioned matrix is clustered at points 0 and 1 when $\alpha = 10^{7}$ in Figure 2, which not only supports the validity of the theoretical result but also leads to a fast convergence rate of the GMRES iteration method. See Tables 2 and 3 for detailed results.

5. Conclusion and remarks

In this paper, we establish a new improved shift-splitting (ILSS) preconditioner for the three-by-three block saddle point problems (1.2). We show the unconditional convergence of the ILSS iteration method, and that all the eigenvalues of the preconditioned matrix $P_{\mathrm{ILSS}}^{-1}\mathcal{A}$ are located in a positive real interval. Numerical experiments reveal that this preconditioner has great superiority in suitable parameters compared with other tested preconditioners.

In fact, how to choose the optimal parameters for the ILSS preconditioner is an important problem that needs to be considered. Additionally, the ILSS preconditioned GMRES iteration method with inexact algorithms is also a key point for solving the large linear system (1.2), which is a very practical and interesting problem that requires further research.

References

[1] M. Abdolmaleki, S. Karimi, and D. K. Salkuyeh, A new block-diagonal preconditioner for a class of 3×3 block saddle point problems, *Mediterr. J. Math.* **19** no. 1 (2022), Paper No. 43. DOI MR Zbl

[2] H. Aslani and D. K. Salkuyeh, *Semi-convergence of the APSS method for a class of non-symmetric three-by-three singular saddle point problems*, 2022. arXiv:2208.00814 [math.NA].

[3] H. Aslani, D. K. Salkuyeh, and F. P. A. Beik, On the preconditioning of three-by-three block saddle point problems, *Filomat* **35** no. 15 (2021), 5181–5194. DOI MR

[4] Y. Cao, Shift-splitting preconditioners for a class of block three-by-three saddle point problems, *Appl. Math. Lett.* **96** (2019), 40–46. DOI MR Zbl

[5] P. Degond and P.-A. Raviart, An analysis of the Darwin model of approximation to Maxwell's equations, *Forum Math.* **4** no. 1 (1992), 13–44. DOI MR Zbl

[6] D. Han and X. Yuan, Local linear convergence of the alternating direction method of multipliers for quadratic programs, *SIAM J. Numer. Anal.* **51** no. 6 (2013), 3446–3457. DOI MR Zbl

[7] K. Hu and J. Xu, Structure-preserving finite element methods for stationary MHD models, *Math. Comp.* **88** no. 316 (2019), 553–581. DOI MR Zbl

[8] N. Huang, Variable parameter Uzawa method for solving a class of block three-by-three saddle point problems, *Numer. Algorithms* **85** no. 4 (2020), 1233–1254. DOI MR Zbl

[9] N. Huang, Y.-H. Dai, and Q. Hu, Uzawa methods for a class of block three-by-three saddle-point problems, *Numer. Linear Algebra Appl.* **26** no. 6 (2019), e2265. DOI MR Zbl

[10] N. Huang and C.-F. Ma, Spectral analysis of the preconditioned system for the 3×3 block saddle point problem, *Numer. Algorithms* **81** no. 2 (2019), 421–444. DOI MR Zbl

[11] Q. Liu, C. Chen, and Q. Zhang, Perturbation analysis for total least squares problems with linear equality constraint, *Appl. Numer. Math.* **161** (2021), 69–81. DOI MR Zbl

[12] L. Meng, J. Li, and S.-X. Miao, A variant of relaxed alternating positive semi-definite splitting preconditioner for double saddle point problems, *Japan J. Ind. Appl. Math.* **38** no. 3 (2021), 979–998. DOI MR Zbl

[13] P. Monk, Analysis of a finite element method for Maxwell's equations, *SIAM J. Numer. Anal.* **29** no. 3 (1992), 714–729. DOI MR Zbl

[14] D. K. Salkuyeh, H. Aslani, and Z.-Z. Liang, An alternating positive semidefinite splitting preconditioner for the three-by-three block saddle point problems, *Math. Commun.* **26** no. 2 (2021), 177–195. MR Zbl Available at https://hrcak.srce.hr/261512.

[15] L. Wang and K. Zhang, Generalized shift-splitting preconditioner for saddle point problems with block three-by-three structure, *Open Access Library J.* **6** (2019), e5968. DOI

[16] N.-N. Wang and J.-C. Li, On parameterized block symmetric positive definite preconditioners for a class of block three-by-three saddle point problems, *J. Comput. Appl. Math.* **405** (2022), Paper No. 113959. DOI MR Zbl

[17] X. Xie and H.-B. Li, A note on preconditioning for the 3×3 block saddle point problem, *Comput. Math. Appl.* **79** no. 12 (2020), 3289–3296. DOI MR Zbl

[18] N. Zhang, R.-X. Li, and J. Li, Lopsided shift-splitting preconditioner for saddle point problems with three-by-three structure, *Comput. Appl. Math.* **41** no. 6 (2022), Paper No. 261. DOI MR Zbl

Jun Li✉
School of Science, Lanzhou University of Technology, Lanzhou, Gansu 730050, People's Republic of China
junli026430@163.com

Xiangtuan Xiong
College of Mathematics and Statistics, Northwest Normal University, Lanzhou, Gansu 730070, People's Republic of China
xiongxt@gmail.com

Received: October 30, 2022
Accepted: May 24, 2023
Early view: August 22, 2024