East Asian Journal on Applied Mathematics
doi: 10.4208/eajam.260815.280216a

Vol. **6**, No. 3, pp. 235-252
August 2016

A Block Diagonal Preconditioner for Generalised Saddle Point Problems

Zhong Zheng and Guo Feng Zhang*

School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, P. R. China.

Received 26 August 2015; Accepted (in revised version) 28 February 2016.

Abstract. A lopsided alternating direction iteration (LADI) method and an induced block diagonal preconditioner for solving block two-by-two generalised saddle point problems are presented. The convergence of the LADI method is analysed, and the block diagonal preconditioner can accelerate the convergence rates of Krylov subspace iteration methods such as GMRES. Our new preconditioned method only requires a solver for two linear equation sub-systems with symmetric and positive definite coefficient matrices. Numerical experiments show that the GMRES with the new preconditioner is quite effective.

AMS subject classifications: 15A06, 65F10, 65H10

Key words: Generalised saddle point problem, Krylov subspace methods, alternating direction iteration, preconditioning, convergence.

1. Introduction

Recently, considerable attention has been devoted to solving large linear systems in saddle point form, which arise in a wide variety of scientific computing and engineering applications — e.g. computational fluid dynamics, mixed finite element approximation of elliptic partial differential equations, constrained least-squares problems, and structure analyses [1, 2, 6, 7, 14, 15, 19, 23, 25, 31, 33, 34]. In this article, we consider an iteration solution of the generalised saddle point linear system

$$\mathscr{A}x = \begin{bmatrix} A & B \\ -B^T & C \end{bmatrix} \begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} f \\ -g \end{bmatrix} = b\,, \tag{1.1}$$

where $A \in \mathbb{R}^{n\times n}$ is symmetric and positive definite (SPD), $B \in \mathbb{R}^{n\times m}$ is of full column rank, $C \in \mathbb{R}^{m\times m}$ is symmetric positive semi-definite (SPSD), $y, f \in \mathbb{R}^n$ and $z, g \in \mathbb{R}^m$ $(n \geq m)$.

*Corresponding author. *Email addresses:* zhengzh13@lzu.edu.cn (Z. Zheng), gf_zhang@lzu.edu.cn (G. F. Zhang)

 ©2016 Global-Science Press

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

It is well known that iteration methods are preferable for large problems with sparse large coefficient matrices, and various methods for solving the linear systems such as (1.1) have been developed — e.g. Uzawa-type schemes [12,13,18,22,27], Hermitian and skew-Hermitian Splitting (HSS) method and its variants [5–11,21,35,37], and Krylov subspace methods [2, 4, 26, 30, 32]. However, Krylov methods for example can be very slow or even fail to converge if not conveniently preconditioned, and an acceptable preconditioner should reduce the number of iterations required for convergence but not significantly increase the amount of computation required at each iteration. Many preconditioners for saddle point problems have been proposed, such as the block diagonal preconditoner [23]

$$P_{\mathrm{BDP}} := \begin{bmatrix} P_A & 0 \\ 0 & P_S \end{bmatrix}, \tag{1.2}$$

where P_A and P_S are the respective SPD matrices that approximate A and the Schur complement $S = C + B^T A^{-1} B$; the accelerated Hermitian and skew-Hermitian splitting (AHSS) [3,5] preconditioner

$$P_{\mathrm{AHSS}} := \frac{1}{\alpha+\beta} \begin{bmatrix} \alpha I + A & \\ & \beta I + C \end{bmatrix} \begin{bmatrix} \alpha I & B \\ -B^T & \beta I \end{bmatrix}; \tag{1.3}$$

and the triangular inexact constraint preconditioner [17]:

$$P_{\mathrm{TICP}} := \begin{bmatrix} P_A & \\ & -P_S \end{bmatrix} \begin{bmatrix} I & P_A^{-1} B \\ 0 & I \end{bmatrix}. \tag{1.4}$$

When $C = 0$, the splitting preconditioner (SP) is defined as [20]

$$P_{\mathrm{SP}} := \begin{bmatrix} A + \alpha B B^T & 0 \\ -2B & \frac{1}{\alpha} I \end{bmatrix}. \tag{1.5}$$

Benzi *et al.* [14] have surveyed numerical methods and preconditioners for saddle point problems.

By transforming the coefficient matrix in the linear system (1.1), in Section 2 we construct a lopsided alternating direction iteration (LADI) method that only requires a solver for two linear sub-systems of equations with SPD coefficient matrices, and obtain a new block diagonal preconditioner induced by the LADI method. The convergence of the new iteration method is analysed in Section 3, and some spectral properties of the preconditioned matrix are discussed in Section 4. Numerical experiments in Section 5 show the effectiveness of GMRES with the new preconditioner, and our concluding remarks are in Section 6.

2. The LADI Method

By transforming for the coefficient matrix, we now obtain a lopsided ADI (LADI) method for the generalised saddle point problem (1.1), together with a new block diagonal preconditioner induced by the LADI method.

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

We first introduce the matrix splitting

$$\mathscr{A} = \begin{bmatrix} A & B \\ -B^T & C \end{bmatrix} = \mathscr{M} + \mathscr{N} \tag{2.1}$$

where

$$\mathscr{M} := \begin{bmatrix} A & \\ & C+I \end{bmatrix} \quad \text{and} \quad \mathscr{N} := \begin{bmatrix} 0 & B \\ -B^T & -I \end{bmatrix}, \tag{2.2}$$

and define the matrix

$$\mathscr{P}^{-1} := \begin{bmatrix} -I & -B \\ B^T & 0 \end{bmatrix}.$$

As B is of full column rank, the matrix $\mathscr{P}^{-1}$ exists [14]. Multiplying both sides of Eq. (2.1) from the left with $\mathscr{P}^{-1}$, we obtain

$$\mathscr{P}^{-1}\mathscr{A} = \mathscr{P}^{-1}\mathscr{N} + \mathscr{P}^{-1}\mathscr{M},$$

where

$$\mathscr{P}^{-1}\mathscr{N} = \begin{bmatrix} BB^T & 0 \\ 0 & B^TB \end{bmatrix} \quad \text{and} \quad \mathscr{P}^{-1}\mathscr{M} = \begin{bmatrix} -A & -B(C+I) \\ B^TA & 0 \end{bmatrix}. \tag{2.3}$$

Based on the matrices in (2.2) and (2.3), we propose a lopsided ADI (LADI) method as follows:

The LADI Method: *Given an initial guess $x^{(0)} \in \mathbb{C}^{n+m}$, for $k = 0, 1, 2, \cdots$ until the sequence of iteration $\{x^{(k)}\}_{k=0}^{\infty} \subset \mathbb{C}^{n+m}$ converges, compute the next iterate $x^{(k+1)}$ according to the following procedure:*

$$\begin{cases} \mathscr{M}x^{(k+\frac{1}{2})} = -\mathscr{N}x^{(k)} + b, \\ (\alpha\mathscr{V} + \mathscr{P}^{-1}\mathscr{N})x^{(k+1)} = (\alpha\mathscr{V} - \mathscr{P}^{-1}\mathscr{M})x^{(k+\frac{1}{2})} + \mathscr{P}^{-1}b, \end{cases} \tag{2.4}$$

where α is a given positive parameter and $\mathscr{V} = blkdiag(A, B^TB) \in \mathbb{R}^{(n+m)\times(n+m)}$.

In matrix-vector form, the above LADI method (2.4) can be rewritten as

$$x^{(k+1)} = L(\alpha)x^{(k)} + R(\alpha)b, \quad k = 0,\ 1,\ 2,\ \cdots, \tag{2.5}$$

where the iteration matrix is

$$\begin{aligned} L(\alpha) =& (\alpha\mathscr{V} + \mathscr{P}^{-1}\mathscr{N})^{-1}(\alpha\mathscr{V} - \mathscr{P}^{-1}\mathscr{M})\mathscr{M}^{-1}(-\mathscr{N}) \\ =& \begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1} \begin{bmatrix} BB^T & -\alpha B \\ \alpha B^T & (\alpha+1)I + C \end{bmatrix} \end{aligned} \tag{2.6}$$

and

$$\begin{aligned} R(\alpha) =& \alpha(\alpha\mathscr{V} + \mathscr{P}^{-1}\mathscr{N})^{-1}\mathscr{V}\mathscr{M}^{-1} \\ =& \alpha \begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1}. \end{aligned}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Indeed, Eq. (2.5) may also result from the splitting

$$\mathscr{A} = F(\alpha) - G(\alpha)$$

of the coefficient matrix $\mathscr{A}$, where

$$F(\alpha) = \frac{1}{\alpha}\mathscr{M}\mathscr{V}^{-1}(\alpha\mathscr{V} + \mathscr{P}^{-1}\mathscr{N}) = \frac{1}{\alpha}\begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix} \tag{2.7}$$

and

$$G(\alpha) = \frac{1}{\alpha}\mathscr{M}\mathscr{V}^{-1}(\alpha\mathscr{V} - \mathscr{P}^{-1}\mathscr{M})\mathscr{M}^{-1}(-\mathscr{N}) = \frac{1}{\alpha}\begin{bmatrix} BB^T & -\alpha B \\ \alpha B^T & (\alpha+1)I + C \end{bmatrix}.$$

The matrix $F(\alpha)$ can then be used as a preconditioner for the coefficient matrix $\mathscr{A} \in \mathbb{R}^{(n+m)\times(n+m)}$, which we refer to as the LADI preconditioner.

When implemented to precondition a Krylov subspace method such as GMRES [32], the action of the preconditioning matrix $F(\alpha)$ in Eq. (2.7) is often realised through solving a sequence of generalised residual equations of the form

$$F(\alpha)v = r\,,$$

where $r = (r_1^T, r_2^T)^T \in \mathbb{R}^{n+m}$ is the current residual vector and $v = (v_1^T, v_2^T)^T \in \mathbb{R}^{n+m}$ the generalised residual vector. Given the special structures of the matrix $F(\alpha)$ in Eq. (2.7), the following procedure for computing the vector v can be applied:

(i) Compute $v_1 \in \mathbb{R}^n$ by solving the systems of linear equations: $(\alpha A + BB^T)v_1 = \alpha r_1$;
(ii) Compute $v_2 \in \mathbb{R}^m$ by solving the systems of linear equations: $(I+C)v_2 = \alpha/(\alpha+1)r_2$.

Since both of the matrices $\alpha A + BB^T$ and $I + C$ are SPD, the above two linear systems of equations can be solved either exactly by Cholesky factorization or inexactly by some inner iteration scheme [10].

3. Convergence Analysis of the LADI Method

In this section, before the convergence of the LADI method is discussed let us first note some helpful Lemmas.

Lemma 3.1 (cf. Ref. [36]). *Both roots of the real quadratic equation $\lambda^2 + \eta_1\lambda + \eta_2 = 0$ have modulus less than one if and only if $|\eta_2| < 1$ and $|\eta_1| < 1 + \eta_2$.*

Lemma 3.2 (cf. Ref. [28]). *All roots of the real cubic equation $\lambda^3 + p\lambda^2 + q\lambda + r = 0$ have modulus less than one if and only if*

$$\begin{cases} |r| < 1\,, \\ 1 - p + q - r > 0, \;\; 1 + p + q + r > 0\,, \\ 1 - q + pr - r^2 > 0\,. \end{cases}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Lemma 3.3. *Let $A \in \mathbb{R}^{n\times n}$ be a SPD matrix and $C \in \mathbb{R}^{m\times m}$ be a SPSD matrix, and let $B \in \mathbb{R}^{n\times m}$ be of full column rank ($r(B) = m$). Then no eigenvalue of the iteration matrix $L(\alpha)$ in the LADI method can be* 1.

Proof. Let λ be an eigenvalue of the iteration matrix $L(\alpha)$ and $(u^T, v^T)^T \in \mathbb{R}^{n+m}$ be the corresponding eigenvector. If $\lambda = 1$, from Eq. (2.6) we have

$$\begin{cases} Au + Bv = 0\,, \\ -B^T u + Cv = 0\,. \end{cases} \tag{3.1}$$

Since the coefficient matrix of the linear system (3.1) is nonsingular, we then have the trivial solution $u = 0$ and $v = 0$, contradicting the assumption that $(u^T, v^T)^T$ is an eigenvector of the iteration matrix $L(\alpha)$. □

Lemma 3.4. *Let $A \in \mathbb{R}^{n\times n}$ be a SPD matrix and $C \in \mathbb{R}^{m\times m}$ be a SPSD matrix, and let $B \in \mathbb{R}^{n\times m}$ be of full column rank ($r(B) = m$). Then the generalised eigenvalues λ of*

$$G(\alpha)z = \lambda F(\alpha)z$$

satisfy $BS_0B^Tu = \lambda Au$, where $S_0 = (1-\lambda)I - \alpha^2[(\lambda-1)(\alpha+1)I + (\lambda\alpha+\lambda-1)C]^{-1}$.

Proof. The generalised eigenvalue problem can be rewritten as

$$\begin{bmatrix} BB^T & -\alpha B \\ \alpha B^T & (\alpha+1)I + C \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \lambda \begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix},$$

where the second equation on expanding is $[(\lambda-1)(\alpha+1)I + (\lambda\alpha+\lambda-1)C]v = \alpha B^T u$. Given $[(\lambda-1)(\alpha+1)I + (\lambda\alpha+\lambda-1)C]$ is nonsingular, on eliminating v from the first equation on expanding we have $BS_0B^Tu = \lambda Au$. □

We will specify the SPSD matrix C below as the convergence of the LADI method seems difficult to analyse more generally, and let us first note some results for matrix $A^{-1}BB^T$. Since A is SPD and B has full rank, the eigenvalues of $A^{-1}BB^T$ are positive real numbers — i.e.,

$$A^{-1}BB^T\tilde{z} = \mu\tilde{z}\,, \qquad \tilde{z} \neq 0\,, \tag{3.2}$$

where $\mu > 0$. Eq. (3.2) may be rewritten as $\widehat{B}\widehat{B}^T\hat{z} = \mu\hat{z}$, where $\widehat{B} := A^{-1/2}B$ and $\hat{z} := A^{-1/2}\tilde{z}$. Moreover, there exist orthogonal matrices $U \in \mathbb{C}^{n\times n}$ and $V \in \mathbb{C}^{m\times m}$ such that

$$\widehat{B} = U^* \begin{bmatrix} \Sigma_m \\ 0 \end{bmatrix} V\,, \tag{3.3}$$

where $\Sigma_m = \mathrm{diag}(\sigma_1, \sigma_2, \cdots, \sigma_m) \in \mathbb{R}^{m\times m}$ and $\sigma_i = \sqrt{\mu_i}$. The following theorem then describes the convergence of the LADI method for the system (1.1) for the specified C.

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Theorem 3.1. *Let $A \in \mathbb{R}^{n\times n}$ be a SPD matrix and $C = \beta I \in \mathbb{R}^{m\times m}$ with $\beta = 0$ or 1, and let $B \in \mathbb{R}^{n\times m}$ be of full column rank. Then the LADI method is convergent if one of the following conditions holds:*
Case I: $\mu_{\max} \le \beta + 1$, $\forall\, \alpha > 0$;
Case II: $\mu_{\max} > \beta + 1$, $0 < \alpha \le (\beta+1)/(\mu_{\max} - \beta - 1)$, where $\mu_{\max}$ denotes the maximum eigenvalue from Eq. (3.2).

Proof. The iteration matrix given in (2.6) can be rewritten as

$$\begin{aligned} L(\alpha) &= \begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1} \begin{bmatrix} BB^T & -\alpha B \\ \alpha B^T & (\alpha+1)I + C \end{bmatrix} \\ &= \begin{bmatrix} A^{-\frac{1}{2}} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \alpha I + \widehat{B}\widehat{B}^T & 0 \\ 0 & (\alpha+1)(\beta+1)I \end{bmatrix}^{-1} \begin{bmatrix} \widehat{B}\widehat{B}^T & -\alpha\widehat{B} \\ \alpha\widehat{B}^T & (\alpha+\beta+1)I \end{bmatrix} \begin{bmatrix} A^{\frac{1}{2}} & 0 \\ 0 & I \end{bmatrix}, \end{aligned}$$

and it has the same eigenvalues as the matrix

$$\begin{aligned} \widetilde{L}(\alpha) :&= \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix}^* \begin{bmatrix} \alpha I + \Sigma_m^2 & 0 & 0 \\ 0 & \alpha I & 0 \\ 0 & 0 & (\alpha+1)(\beta+1)\, I \end{bmatrix}^{-1} \begin{bmatrix} \Sigma_m^2 & 0 & -\alpha\Sigma_m \\ 0 & 0 & 0 \\ \alpha\Sigma_m & 0 & (\alpha+\beta+1)\, I \end{bmatrix} \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix} \\ &= \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix}^* \begin{bmatrix} (\alpha I + \Sigma_m^2)^{-1}\Sigma_m^2 & 0 & -\alpha(\alpha I + \Sigma_m^2)^{-1}\Sigma_m \\ 0 & 0 & 0 \\ \alpha(\alpha+1)^{-1}(\beta+1)^{-1}\Sigma_m & 0 & (\alpha+\beta+1)/[(\alpha+1)(\beta+1)]\, I \end{bmatrix} \begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix}, \end{aligned}$$

where the star denotes the complex conjugate transpose. In addition, $\rho(L(\alpha)) = \rho(\widehat{L}(\alpha))$, so that

$$\widehat{L}(\alpha) := \begin{bmatrix} (\alpha I + \Sigma_m^2)^{-1}\Sigma_m^2 & -\alpha(\alpha I + \Sigma_m^2)^{-1}\Sigma_m \\ \alpha(\alpha+1)^{-1}(\beta+1)^{-1}\Sigma_m & (\alpha+\beta+1)/[(\alpha+1)(\beta+1)]\, I \end{bmatrix},$$

and the two eigenvalues of the matrix $\widehat{L}(\alpha)$ are the roots of the quadratic equation

$$\lambda^2(\alpha+1)(\beta+1)(\alpha+\mu) - \lambda[(\alpha+1+\beta)(\alpha+\mu) + (\alpha+1)(\beta+1)\mu] + (\alpha^2+\alpha+1+\beta)\mu = 0\,. \quad (3.4)$$

From Lemma 3.1, both roots λ of this real quadratic equation (3.4) satisfy $|\lambda| < 1$ if and only if

$$\begin{cases} |(\alpha^2+\alpha+1+\beta)\mu| < (\alpha+1)(\beta+1)(\alpha+\mu)\,, \\ |(\alpha+1+\beta)(\alpha+\mu) + (\alpha+1)(\beta+1)\mu| \\ \qquad\qquad < (\alpha^2+\alpha+1+\beta)\mu + (\alpha+1)(\beta+1)(\alpha+\mu)\,, \end{cases} \quad (3.5)$$

and the straightforward solution of (3.5) is

$$\alpha\mu < \alpha(\beta+1) + \beta\mu + \beta + 1\,.$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Consequently, for $0 < \mu \le \mu_{\max}$ and $\alpha > 0$ we have

$$\alpha\mu \le \alpha\mu_{\max} \le \alpha(\beta+1)+\beta+1 < \alpha(\beta+1)+\beta\mu_{\min}+\beta+1 \le \alpha(\beta+1)+\beta\mu+\beta+1\,,$$

such that $\mu_{\max} \le \beta+1\,,\ \forall\alpha>0$ and $\mu_{\max} > \beta+1$ for $0<\alpha\le(\beta+1)/(\mu_{\max}-\beta-1)$. □

The following theorem describes the convergence of the LADI method for the system (1.1) with a nontrivial SPSD matrix $C = \mathrm{blkdiag}(I,0)$, where "blkdiag" means "block diagonal".

Theorem 3.2. *Let $A\in\mathbb{R}^{n\times n}$ be SPD, $C = blkdiag(I,0)\in\mathbb{R}^{m\times m}$ be SPSD with $I\in\mathbb{R}^{k\times k}$ ($k\le m$), and $B=[B_1,\ B_2]\in\mathbb{R}^{n\times m}$ be of full column rank with $B_1\in\mathbb{R}^{n\times k}$ and $B_2\in\mathbb{R}^{n\times(m-k)}$. Then the LADI method is convergent if one of the following conditions holds:*
Case I. $\mu_{\max}\le 2\,,\ \forall\alpha>0$.
Case II. $\mu_{\max}>2$ *and* $0<\alpha\le 2/(\mu_{\max}-2)$.

Proof. Assume that λ is an eigenvalue of the iteration matrix $L(\alpha)$ and $\hat z=(u^T,v^T)^T$ is the corresponding eigenvector. Then $L(\alpha)\hat z=\lambda\hat z$. From (2.6) we have

$$\begin{bmatrix} BB^T & -\alpha B_1 & -\alpha B_2\\ \alpha B_1^T & (\alpha+2)I & 0\\ \alpha B_2^T & 0 & (\alpha+1)I\end{bmatrix}\begin{bmatrix}u\\v_1\\v_2\end{bmatrix}=\lambda\begin{bmatrix}\alpha A+BB^T & 0 & 0\\ 0 & 2(\alpha+1)I & 0\\ 0 & 0 & (\alpha+1)I\end{bmatrix}\begin{bmatrix}u\\v_1\\v_2\end{bmatrix},$$

where $v=(v_1^T,v_2^T)^T$ with $v_1\in\mathbb{R}^k,\ v_2\in\mathbb{R}^{m-k}$ — i.e.,

$$\begin{cases} BB^Tu-\alpha B_1v_1-\alpha B_2v_2=\lambda(\alpha A+BB^T)u\,,\\ \alpha B_1^Tu+(\alpha+2)v_1=2\lambda(\alpha+1)v_1\,,\\ \alpha B_2^Tu+(\alpha+1)v_2=\lambda(\alpha+1)v_2\,.\end{cases}\tag{3.6}$$

From the second equation of (3.6), we have $\alpha B_1^Tu=(2\lambda(\alpha+1)-(\alpha+2))v_1$. Consequently, If $\lambda=(\alpha+2)/(2\alpha+2)$, then a nonzero vector $u\in\mathbb{R}^{m-k}$ exists such that $L(\alpha)\hat z=0$, when $\forall\alpha>0$, $0<|\lambda|<1$ holds. If $\lambda\ne(\alpha+2)/(2\alpha+2)$, from Lemma 3.3 we know that $\lambda\ne1$, and from (3.6) we have

$$v_1=\frac{\alpha}{2\lambda(\alpha+1)-(\alpha+2)}B_1^Tu\,,\qquad v_2=\frac{\alpha}{(\lambda-1)(\alpha+1)}B_2^Tu\,.$$

On eliminating v_1 and v_2 in (3.6) and multiplying both sides of the resulting equation from the left with $u^*/(u^*u)$, we get

$$\begin{aligned}&\frac{u^*BB^Tu}{u^*u}-\frac{\alpha^2}{2\lambda(\alpha+1)-(\alpha+2)}\frac{u^*B_1B_1^Tu}{u^*u}-\frac{\alpha^2}{(\lambda-1)(\alpha+1)}\frac{u^*B_2B_2^Tu}{u^*u}\\ =&\lambda\frac{u^*(\alpha A+BB^T)u}{u^*u}.\end{aligned}\tag{3.7}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Let us conveniently annotate the following two non-negative components as

$$\mu_1 := \frac{u^* B_1 B_1^T u}{u^* A u}\,, \qquad \mu_2 := \frac{u^* B_2 B_2^T u}{u^* A u}\,,$$

and write $\mu = \mu_1 + \mu_2$. Then Eq. (3.7) may be represented explicitly as the cubic equation

$$\delta_1 \lambda^3 + \delta_2 \lambda^2 + \delta_3 \lambda + \delta_4 = 0\,, \tag{3.8}$$

with coefficients

$$\begin{aligned}
&\delta_1 := 2(\alpha+1)^2(\alpha+\mu)\,,\ \delta_2 := -2(\alpha+1)^2(\alpha+\mu) - (\alpha+1)(\alpha+2)(\alpha+\mu) - 2(\alpha+1)^2\mu\,,\\
&\delta_3 := (\alpha+1)(\alpha+2)(\alpha+\mu) + 2(\alpha+1)^2\mu + (\alpha+1)(\alpha+2)\,\mu + (\alpha+1)\,\alpha^2\mu_1 + 2(\alpha+1)\,\alpha^2\mu_2\,,\\
&\delta_4 := -(\alpha+1)(\alpha+2)\,\mu - (\alpha+1)\,\alpha^2\mu_1 - (\alpha+2)\,\alpha^2\mu_2\,.
\end{aligned}$$

From Lemma 3.2, all roots of the real cubic equation (3.8) satisfy $|\lambda| < 1$ if and only if

$$\begin{cases}
|\delta_4| < \delta_1\,,\\
\delta_1 - \delta_2 + \delta_3 - \delta_4 > 0\,,\\
\delta_1 + \delta_2 + \delta_3 + \delta_4 > 0\,,\\
\delta_1^2 - \delta_1\delta_3 + \delta_2\delta_4 - \delta_4^2 > 0\,,
\end{cases} \tag{3.9}$$

so on inserting the $\{\delta_i\}$ from above we have the inequality

$$\alpha^2\mu + \alpha\mu_2 < 2\alpha^2 + 4\alpha + 2 + \mu\,, \tag{3.10}$$

such that

$$\alpha^2\mu + \alpha\mu_2 < \alpha^2\mu_{\max} + \alpha\mu_{\max} \le 2\alpha^2 + 4\alpha + 2 \le 2\alpha^2 + 4\alpha + 2 + \mu\,, \tag{3.11}$$

and hence we obtain the two specified convergence conditions. □

4. Analysis of Preconditioning Properties

We now analyse the spectral properties of the preconditioning matrix $F(\alpha)^{-1}\mathscr{A}$ for various $(2,2)$–block matrices C, where C is SPSD and $C \le I$ — i.e., $I - C$ is symmetric positive semi-definite. The preconditioned matrix $F(\alpha)^{-1}\mathscr{A}$ can be rewritten as

$$\begin{aligned}
K := F(\alpha)^{-1}\mathscr{A} &= \alpha \begin{bmatrix} \alpha A + BB^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1} \begin{bmatrix} A & B \\ -B^T & C \end{bmatrix}\\
&= \alpha \begin{bmatrix} A^{-1/2} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \alpha I + \widehat{B}\widehat{B}^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1} \begin{bmatrix} I & \widehat{B} \\ -\widehat{B}^T & C \end{bmatrix} \begin{bmatrix} A^{\frac{1}{2}} & 0 \\ 0 & I \end{bmatrix},
\end{aligned}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

which has the same eigenvalues as the matrix

$$\begin{aligned}\widetilde{K} &= \alpha\begin{bmatrix} \alpha I + \widehat{B}\widehat{B}^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1/2} \begin{bmatrix} I & \widehat{B} \\ -\widehat{B}^T & C \end{bmatrix} \begin{bmatrix} \alpha I + \widehat{B}\widehat{B}^T & 0 \\ 0 & (\alpha+1)(I+C) \end{bmatrix}^{-1/2} \\ &= \alpha\begin{bmatrix} (\alpha I + \widehat{B}\widehat{B}^T)^{-1} & \frac{1}{\sqrt{\alpha+1}}(\alpha I + \widehat{B}\widehat{B}^T)^{-1/2}\widehat{B}(I+C)^{-1/2} \\ \frac{-1}{\sqrt{\alpha+1}}(I+C)^{-1/2}\widehat{B}^T(\alpha I + \widehat{B}\widehat{B}^T)^{-1/2} & \frac{1}{\alpha+1}(I+C)^{-1/2}C(I+C)^{-1/2} \end{bmatrix} \\ &=: \begin{bmatrix} \widetilde{A} & \widetilde{B} \\ -\widetilde{B}^T & \widetilde{C} \end{bmatrix}, \end{aligned} \tag{4.1}$$

where

$$\begin{aligned} \widetilde{A} &:= \alpha(\alpha I + \widehat{B}\widehat{B}^T)^{-1}, \qquad \widetilde{C} := \frac{\alpha}{\alpha+1}(I+C)^{-1/2}C(I+C)^{-1/2}, \\ \widetilde{B} &:= \frac{\alpha}{\sqrt{\alpha+1}}(\alpha I + \widehat{B}\widehat{B}^T)^{-1/2}\widehat{B}(I+C)^{-1/2}, \end{aligned} \tag{4.2}$$

so we now consider the spectral properties of the matrix $\widetilde{K}$ as follows.

Lemma 4.1 (cf. Refs. [16, 29]). *Let $\widetilde{K}$ be a matrix of the form* (4.1)*, with $\widetilde{A} \in R^{n\times n}$ being SPD and $\widetilde{C} \in R^{m\times m}$ being SPSD ($m \le n$). Let $S_{\widetilde{C}} := \widetilde{C} + \widetilde{B}^T\widetilde{A}^{-1}\widetilde{B}$, and for $\widetilde{C}$ positive definite let $S_{\widetilde{A}} := \widetilde{A} + \widetilde{B}\widetilde{C}^{-1}\widetilde{B}^T$. Then the real eigenvalues λ of the matrix $\widetilde{K}$ satisfy*

$$\min\left\{\lambda_{\min}(\widetilde{A}),\ \lambda_{\min}(S_{\widetilde{C}})\right\} \le\ \lambda\ \le \max\left\{\lambda_{\max}(\widetilde{A}),\ \lambda_{\max}(\widetilde{C})\right\}.$$

Moreover, the eigenvalues λ with nonzero imaginary part satisfy

$$\frac{1}{2}\Big(\lambda_{\min}(\widetilde{A}) + \lambda_{\min}(\widetilde{C})\Big) \le\ Re(\lambda)\ \le \frac{1}{2}\Big(\lambda_{\max}(\widetilde{A}) + \lambda_{\max}(\widetilde{C})\Big), \qquad |Im(\lambda)| \le (\lambda_{\max}(\widetilde{B}\widetilde{B}^T))^{\frac{1}{2}},$$

and $|\lambda - \zeta| \le \zeta$ where

$$\zeta := \begin{cases} \dfrac{\lambda_{\max}(S_{\widetilde{A}})\lambda_{\max}(S_{\widetilde{C}})}{\lambda_{\max}(S_{\widetilde{A}}) + \lambda_{\max}(S_{\widetilde{C}})}, & \text{if } \widetilde{C} \text{ is positive definite}, \\ \lambda_{\max}(S_{\widetilde{C}}), & \text{otherwise}. \end{cases}$$

Consequently, we obtain the following theorem.

Theorem 4.1. *Let $\widetilde{K}$ be as defined in* (4.1)*; and let $\underline{\xi}$, $\overline{\xi}$, $\underline{\chi}$, $\overline{\chi}$, δ and ζ be as defined in Table* 1*. Let $\mu_{\max}$ denote the largest eigenvalue of the generalised eigenvalue problem* (3.2)*, and $\nu_{\min}$ the smallest eigenvalue of $B^TA^{-1}B$. Then the real eigenvalues λ of $F(\alpha)^{-1}\mathscr{A}$ satisfy*

$$\underline{\xi} \le \lambda \le \overline{\xi}, \tag{4.3}$$

and the eigenvalues with nonzero imaginary part satisfy

$$\underline{\chi} \le Re(\lambda) \le \overline{\chi}, \tag{4.4}$$

$$|Im(\lambda)| \le \delta, \tag{4.5}$$

$$|\lambda - \zeta| \le \zeta. \tag{4.6}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Proof. For our chosen matrix C, we have

$$\frac{1}{2}I \le (I+C)^{-1} \le I\,, \tag{4.7}$$

where $X \le Y$ means $Y - X$ is positive semi-definite. Then

$$\lambda_{\min}(\widetilde{C}) = \begin{cases} 0\,, & C = 0 \text{ or } C = \mathrm{blkdiag}(I,0)\,, \\ \frac{\alpha}{2(\alpha+1)}\,, & C = I\,, \end{cases}$$

and

$$\lambda_{\max}(\widetilde{C}) = \begin{cases} 0\,, & C = 0\,, \\ \frac{\alpha}{2(\alpha+1)}\,, & C = I \text{ or } C = \mathrm{blkdiag}(I,0)\,. \end{cases} \tag{4.8}$$

Recalling Eqs. (3.2), (3.3) and (4.2), we have from (4.7)

$$\begin{aligned} \lambda_{\max}(\widetilde{B}\widetilde{B}^T) &= \frac{\alpha^2}{\alpha+1}\lambda_{\max}\left((\alpha I + \widehat{B}\widehat{B}^T)^{-1/2}\widehat{B}(I+C)^{-1}\widehat{B}^T(\alpha I + \widehat{B}\widehat{B}^T)^{-1/2}\right) \\ &\le \frac{\alpha^2}{\alpha+1}\lambda_{\max}\left((I+C)^{-1}\right)\lambda_{\max}\left(\widehat{B}\widehat{B}^T(\alpha I + \widehat{B}\widehat{B}^T)^{-1}\right) \\ &\le \begin{cases} \frac{\alpha^2\mu_{\max}}{(\alpha+1)(\alpha+\mu_{\max})}\,, & C = 0 \text{ or } C = \mathrm{blkdiag}(I,0)\,, \\ \frac{\alpha^2\mu_{\max}}{2(\alpha+1)(\alpha+\mu_{\max})}\,, & C = I\,. \end{cases} \end{aligned} \tag{4.9}$$

For $\lambda_{\max}(\widetilde{A})$ we obtain

$$\lambda(\widetilde{A}) = \alpha\lambda((\alpha I + \widehat{B}\widehat{B}^T)^{-1}) \in \left[\frac{\alpha}{\alpha+\mu_{\max}},\ 1\right]; \tag{4.10}$$

and from

$$S_{\widetilde{C}} = \widetilde{C} + \widetilde{B}^T\widetilde{A}^{-1}\widetilde{B} = \frac{\alpha}{\alpha+1}(I+C)^{-1}(C + \widehat{B}^T\widehat{B}),$$

together with (4.7) we have

$$S_{\widetilde{C}} \le \begin{cases} \frac{\alpha}{\alpha+1}\widehat{B}^T\widehat{B}\,, & C = 0\,, \\ \frac{\alpha}{2(\alpha+1)}I + \frac{\alpha}{(\alpha+1)}\widehat{B}^T\widehat{B}\,, & C = \mathrm{blkdiag}(I,0)\,, \\ \frac{\alpha}{2(\alpha+1)}I + \frac{\alpha}{2(\alpha+1)}\widehat{B}^T\widehat{B}\,, & C = I\,. \end{cases}$$

Consequently,

$$\lambda(S_{\widetilde{C}}) \in \begin{cases} \left[\frac{\alpha\nu_{\min}}{\alpha+1},\ \frac{\alpha\mu_{\max}}{\alpha+1}\right], & C = 0\,, \\ \left[\frac{\alpha(1+2\nu_{\min})}{2(\alpha+1)},\ \frac{\alpha(1+2\mu_{\max})}{2(\alpha+1)}\right], & C = \mathrm{blkdiag}(I,0)\,, \\ \left[\frac{\alpha(1+\nu_{\min})}{2(\alpha+1)},\ \frac{\alpha(1+\mu_{\max})}{2(\alpha+1)}\right], & C = I\,. \end{cases} \tag{4.11}$$

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Table 1: $\underline{\xi}$, $\overline{\xi}$, $\underline{\chi}$, $\overline{\chi}$, δ and ζ for various $(2,2)$-block matrices C.

	$C=0$	$C=\mathrm{blkdiag}(I,0)$	$C=I$
$\underline{\xi}$	$\min\left\{\frac{\alpha}{\alpha+\mu_{\max}}, \frac{\alpha\nu_{\min}}{\alpha+1}\right\}$	$\min\left\{\frac{\alpha}{\alpha+\mu_{\max}}, \frac{\alpha(1+2\nu_{\min})}{2(\alpha+1)}\right\}$	$\min\left\{\frac{\alpha}{\alpha+\mu_{\max}}, \frac{\alpha(1+\nu_{\min})}{2(\alpha+1)}\right\}$
$\overline{\xi}$	1	1	1
$\underline{\chi}$	$\frac{\alpha}{2(\alpha+\mu_{\max})}$	$\frac{\alpha}{2(\alpha+\mu_{\max})}$	$\frac{\alpha(3\alpha+2+\mu_{\max})}{4(\alpha+1)(\alpha+\mu_{\max})}$
$\overline{\chi}$	$1/2$	$\frac{3\alpha+2}{4(\alpha+1)}$	$\frac{3\alpha+2}{4(\alpha+1)}$
δ	$\alpha\sqrt{\frac{\mu_{\max}}{(\alpha+1)(\alpha+\mu_{\max})}}$	$\alpha\sqrt{\frac{\mu_{\max}}{(\alpha+1)(\alpha+\mu_{\max})}}$	$\alpha\sqrt{\frac{\mu_{\max}}{2(\alpha+1)(\alpha+\mu_{\max})}}$
ζ	$\frac{\alpha\mu_{\max}}{\alpha+1}$	$\frac{\alpha(1+2\mu_{\max})}{2(\alpha+1)}$	$\frac{\alpha(1+\mu_{\max})}{3\alpha+2+\mu_{\max}}$

When $C=I$,

$$\lambda_{\max}(S_{\widetilde{A}}) = \alpha\lambda_{\max}\left((I+\widehat{B}\widehat{B}^T)(\alpha I+\widehat{B}\widehat{B}^T)^{-1}\right) = \frac{\alpha(1+\mu_{\max})}{\alpha+\mu_{\max}}, \tag{4.12}$$

so combining (4.11) and (4.12) we obtain

$$\zeta = \begin{cases} \frac{\alpha\mu_{\max}}{\alpha+1}, & C=0, \\ \frac{\alpha(1+2\mu_{\max})}{2(\alpha+1)}, & C=\mathrm{blkdiag}(I,0), \\ \frac{\alpha(1+\mu_{\max})}{3\alpha+2+\mu_{\max}}, & C=I. \end{cases} \tag{4.13}$$

One can check that each of the cases (4.3)–(4.6) hold by combining Lemma 4.1 with the bounds in (4.8)–(4.13) — see also Table 1. □

Remark 4.1. The α such that $\zeta<1$ ensure Theorems 3.1 and 3.2 hold — i.e., the LADI method is convergent.

5. Numerical Experiments

Two examples were considered to examine the effectiveness of our approach. The results show the improvement in the number of iteration steps (denoted by "IT") and elapsed CPU time in seconds (denoted by "CPU") required for the LADI preconditioner in solving the symmetric linear system (1.1) — compared with the AHSS preconditioner defined in (1.3) and the TICP preconditioner defined in (1.4) using the GMRES method [32], and also the BDP preconditioner defined in (1.2) using the MINRES method [26]. In addition, when $C=0$ we compared the SP preconditioner defined in (1.5) using the GMRES

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

method. We chose $P_A = A$ and $P_S = C + B^T \mathrm{diag}(A)^{-1} B$ for the BDP preconditoner, and we chose $P_A = \mathrm{diag}(A)$ and $P_S = C + B^T P_A^{-1} B$ for the TICP preconditioner.

Our initial guess was the zero vector, and the iteration was terminated when either the Euclidean norms of the current residual satisfied

$$\mathrm{ERR} = \frac{||b - \mathscr{A} x^{(k)}||_2}{||b||_2} \leq 10^{-9}$$

or the CPU time exceeded 2000 seconds. In the following tables, "–" means the maximum CPU time was exceeded. All tests were run in MATLAB (version R2013a) double precision, and all experiments were performed on a personal computer with 3.2 GHz CPU (Intel(R) Core(TM)2 i5-3470), 8G memory and the Windows 7 operating system. The right-hand side vector $(f^T, -g^T)^T$ was chosen such that the exact solution of the saddle point problem is $(y^T,\ z^T)^T = (1, 1, \cdots, 1)^T$.

For the LADI preconditioner, we expected $F(\alpha)$ to be as close as possible to $\mathscr{A}$ and $G(\alpha) \approx 0$, so we chose $\alpha > 0$ by minimising the function $\Phi(\alpha) = \|G(\alpha)\|_F$. It is easy to show that $\Phi(\alpha)$ decreases with α, hence the bigger α the better. However, Theorems 3.1 and 3.2 provide a bound on α to ensure the LADI method converges to the unique solution, so we chose either $\alpha = \frac{\beta+1}{\mu_{\max}-\beta-1}$ ($\beta = 0,\ 1$) or $\alpha = \frac{2}{\mu_{\max}-2}$ ($C = \mathrm{blkdiag}(I, 0)$). As $\mu_{\max}$ is difficult to compute, we also used another path — viz. $\alpha_0 = \| BB^T \|_{\inf} / \| A \|_{\inf}$, and the iteration parameter $\alpha_\star = \frac{\beta+1}{|\alpha_0-\beta-1|}$ ($\beta = 0,\ 1$) or $\alpha_\star = \frac{2}{|\alpha_0-2|}$ ($C = \mathrm{blkdiag}(I, 0)$).

Example 5.1 (cf. Ref. [5])**.** Consider the saddle point problem (1.1) when

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

and

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

where $\otimes$ denotes the Kronecker product symbol and $h = 1/(p+1)$ is the discretisation mesh-size.

In Tables 2 to 4, we list the numerical results for various problem sizes and $(2, 2)$–block matrices C, showing the number of iteration steps (IT) and the CPU time (CPU). Fig. 1 depicts the eigenvalue distribution of the LADI preconditioned matrix $F(\alpha)^{-1}\mathscr{A}$ for the $(2, 2)$–block matrices $C = 0$, $\mathrm{blkdiag}(I, 0)$ where $I \in \mathbb{R}^{m/2 \times m/2}$ or $C = I$ for 16×16 grids with the iteration parameters $\alpha = 0.3$ and 0.5, respectively. Here the "*" line denotes the eigenvalues of the preconditioned matrix for p=16, the broken "- -" line defines the region of real eigenvalues and "-" the region of complex eigenvalues, and the "-." line the circle ζ provided in Theorem 4.1 (left. $\alpha = 0.3$, right. $\alpha = 0.5$). We note the following features.

- The LADI preconditioner leads to much better numerical results than the AHSS, TICP and BDP preconditioners, especially when the grid number p becomes large. The

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Table 2: Numerical results for Example 5.1 with various preconditioners and $C = 0$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{SP}			P_{BDP}	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	α	IT	CPU	IT	CPU
32×32	3.3258	39	0.07	1	38	0.06	(1,1)	128	0.32	136	0.32	10	10	0.02	185	0.52
64×64	3.3696	45	0.33	1	44	0.32	(1,1)	210	1.95	237	2.05	10	10	0.09	369	5.15
128×128	3.3918	50	1.76	1	50	1.78	(1,1)	342	16.1	426	18.9	10	11	0.50	742	59.8
256×256	3.4030	54	11.3	1	54	11.5	(1,1)	564	194	–	–	10	12	3.83	1528	640
512×512	3.4086	59	73.4	1	60	75.6	–	–	–	–	–	10	13	20.8	–	–

Table 3: Numerical results for Example 5.1 with various preconditioners and $C = \text{blkdiag}(I, 0)$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{BDP}	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	IT	CPU
32×32	1.5377	44	0.07	1	46	0.08	(1,1)	160	0.47	207	0.73	160	0.47
64×64	1.5423	48	0.37	1	49	0.36	(1,1)	258	2.72	393	5.42	302	4.20
128×128	1.5446	52	1.82	1	54	1.92	(1,1)	417	22.1	726	52.6	591	47.4
256×256	1.5458	53	11.2	1	56	11.8	–	–	–	–	–	1176	491
512×512	1.5463	57	71.7	1	58	72.7	–	–	–	–	–	–	–

Table 4: Numerical results for Example 5.1 with various preconditioners and $C = I$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{BDP}	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	IT	CPU
32×32	1.5377	26	0.05	1	27	0.05	(2,1)	124	0.31	150	0.39	33	0.10
64×64	1.5423	29	0.23	1	28	0.21	(2,1)	205	1.97	294	3.07	35	0.51
128×128	1.5446	30	1.08	1	31	1.11	(2,1)	335	15.7	573	33.1	35	2.90
256×256	1.5458	31	6.47	1	33	6.98	–	–	–	1113	56.9	36	15.6
512×512	1.5463	33	40.8	1	35	43.8	–	–	–	–	–	37	80.3

iteration counts grow much more rapidly with problem size for the AHSS and TICP methods than for the LADI method — although when $C = 0$ the SP method is quite good, and when $C = I$ the iteration counts for the BDP grow quite slowly.

- The LADI preconditioner approach is not sensitive to the value of the parameter α (the iteration count does not change dramatically).
- From the convergence analysis for the LADI method, we know that all eigenvalues of the preconditioned matrix $F(\alpha)^{-1}\mathscr{A}$ are located in a circle with centre $(1,0)$ and radius strictly less than 1 for some positive parameters α. From Fig. 1, we see that Theorem 4.1 predicts the location of both real and non-real eigenvalues.

Example 5.2 (cf. Ref. [3])**.** We considered the incompressible steady Stokes problem

$$\begin{cases} -\nu\triangle u + \nabla p = f\,, \\ \nabla\cdot u = 0\,, \end{cases} \quad \text{in } \Omega$$

with suitable boundary conditions on $\partial\Omega$. Here ν, $\triangle$, u and p denote the viscosity coefficient, the Laplace operator, and the fluid velocity and pressure, respectively.

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

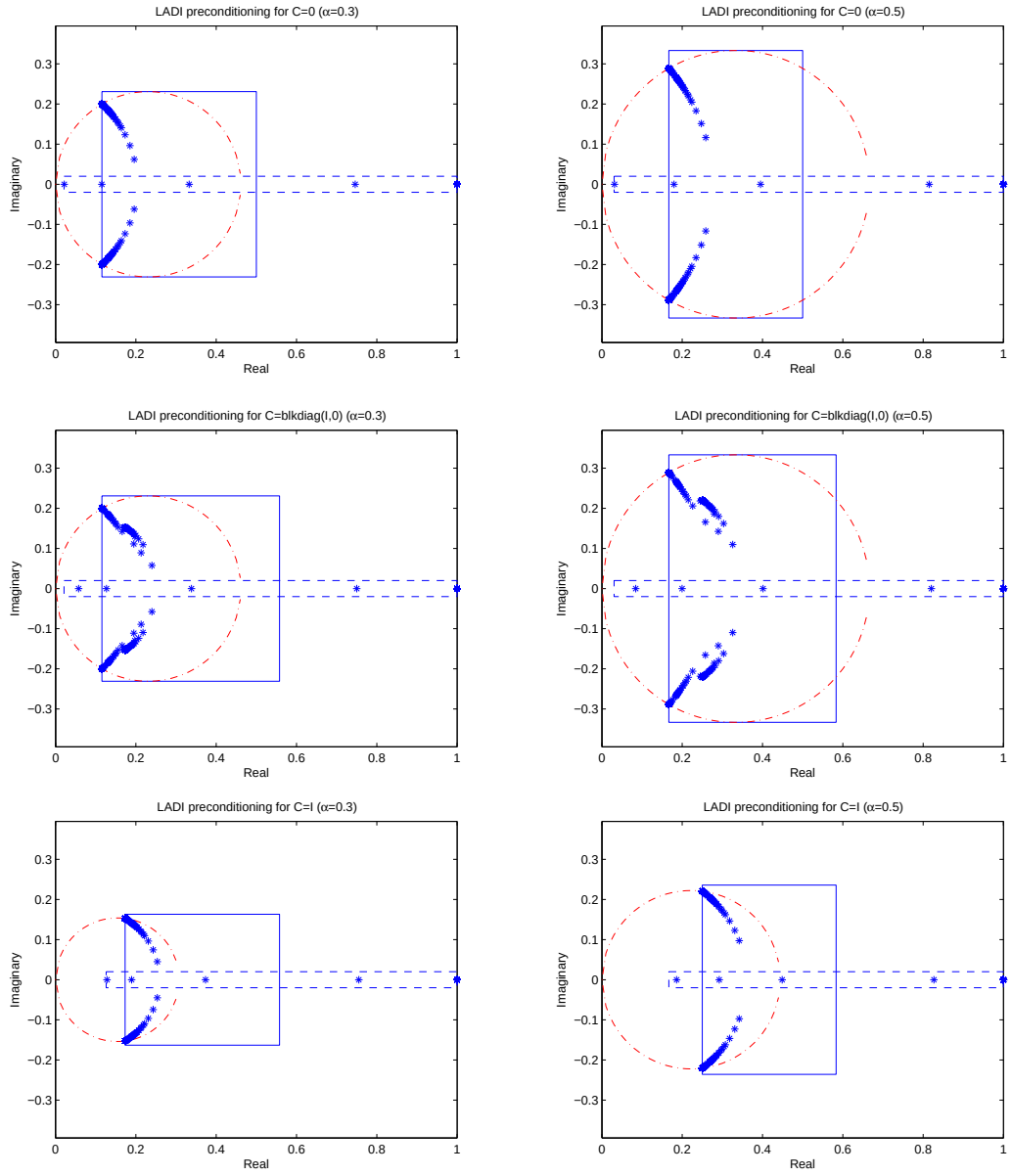


Figure 1: *: eigenvalues of the preconditioned matrix for p=16 for Example 5.1; - -: the region of the real eigenvalues; -: the region of the complex eigenvalues; -.: circle ζ provided in Theorem 4.1 (left $\alpha = 0.3$, right $\alpha = 0.5$).

We chose the 'regularisation' two-dimensional lid-driven cavity problem in a square domain ($0 \le x \le 1,\ 0 \le y \le 1$), and used the IFISS software written by Silvester *et al.* [24] to discretise the Stokes equations. The boundary conditions chosen were $u_x = u_y = 0$ on three fixed walls $x = 0,\ x = 1$ and $y = 0$, and $u_x = 1,\ u_y = 0$ on the moving wall $y = 1$. The finite element subdivision was based on $ne \times ne$ uniform square elements, and the

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

Table 5: Numerical results for Example 5.2 with various preconditioners and $C = 0$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{SP}			P_{BDP}	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	α	IT	CPU	IT	CPU
32×32	1.1772	55	0.15	1	56	0.14	(0.03,0.12)	38	0.07	95	0.20	10	13	0.05	115	1.62
64×64	1.0391	54	0.57	1	55	0.57	(0.012,0.09)	61	0.43	180	1.65	10	17	0.22	232	16.4
128×128	1.0095	51	2.78	1	51	2.76	(0.005,0.09)	94	4.21	340	16.6	10	22	1.35	467	154
256×256	1.0024	47	13.5	1	47	13.3	(0.0015,0.01)	141	37.8	620	221	10	23	7.20	975	1562
512×512	1.0006	44	76.6	1	44	75.8	(0.001,0.001)	218	354	–	–	10	23	42.3	–	–

Table 6: Numerical results for Example 5.2 with various preconditioners and $C = \text{blkdiag}(I, 0)$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{BDP}	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	IT	CPU
32×32	1.0814	53	0.13	1	53	0.14	(0.05,0.5)	71	0.15	163	0.48	121	1.71
64×64	1.0192	61	0.64	1	62	0.65	(0.015,0.06)	106	1.02	306	3.90	223	15.7
128×128	1.0047	62	3.52	1	62	3.35	(0.008,0.02)	164	8.09	577	39.7	433	143
256×256	1.0012	57	16.2	1	57	16.2	(0.003,0.006)	245	74.5	1070	564	867	1463
512×512	1.0003	51	88.5	1	51	87.7	–	–	–	–	–	–	–

Table 7: Numerical results for Example 5.2 with various preconditioners and $C = I$.

Grids	P_{LADI}			P_{LADI}			P_{AHSS}			P_{TICP}		P_{BDP}.	
	$\alpha_\star$	IT	CPU	α	IT	CPU	(α,β)	IT	CPU	IT	CPU	IT	CPU
32×32	1.0814	21	0.07	1	21	0.06	(0.2,4)	53	0.10	82	0.15	21	0.33
64×64	1.0192	14	0.19	1	14	0.18	(0.1,2)	72	0.64	137	1.07	14	1.08
128×128	1.0047	10	0.75	1	10	0.72	(0.04,5)	82	3.65	250	10.4	10	3.72
256×256	1.0012	7	3.00	1	7	2.93	(0.02,5)	102	27.2	436	127	8	19.6
512×512	1.0003	6	15.3	1	6	15.3	(0.01,5)	137	199	717	1684	7	120

mixed finite element used the bilinear-constant velocity-pressure $Q2-P1$ pair with local stabilisation (stabilisation parameter 0.25). The generalised saddle point problem (1.1) is thus obtained, with $\nu A \in \mathbb{R}^{2(l+1)^2\times 2(l+1)^2}$ and $B \in \mathbb{R}^{2(l+1)^2\times \frac{3}{4}l^2}$, where l is ne and $\nu = 0.05$.

In Tables 5-7 we list the IT and CPU for different grids with various $(2,2)$–block matrices C, respectively. Fig. 2 depicts the eigenvalue distribution of the LADI preconditioned matrix $F(\alpha)^{-1}\mathscr{A}$ with $C = 0$, $C = \text{blkdiag}(I, 0)$ where $I \in \mathbb{R}^{m/2\times m/2}$ and $C = I$ for 16×16 grids and two parameters $\alpha = 0.3$ and 0.5, respectively. We note the following features.

- The LADI preconditioner leads to much better numerical results than the AHSS, TICP and BDP preconditioners, especially when the grid number l becomes large. The iteration counts with the AHSS and TICP methods grow rapidly with problem size, whereas the LADI method shows only slight variations. When $C = 0$, the SP method is also very good, and when $C = I$ the iteration counts of the LADI and BDP grow more slowly.
- The LADI preconditioner approach is not sensitive to the value of the parameter α (the iteration count does not change dramatically), whereas the AHSS preconditioner

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

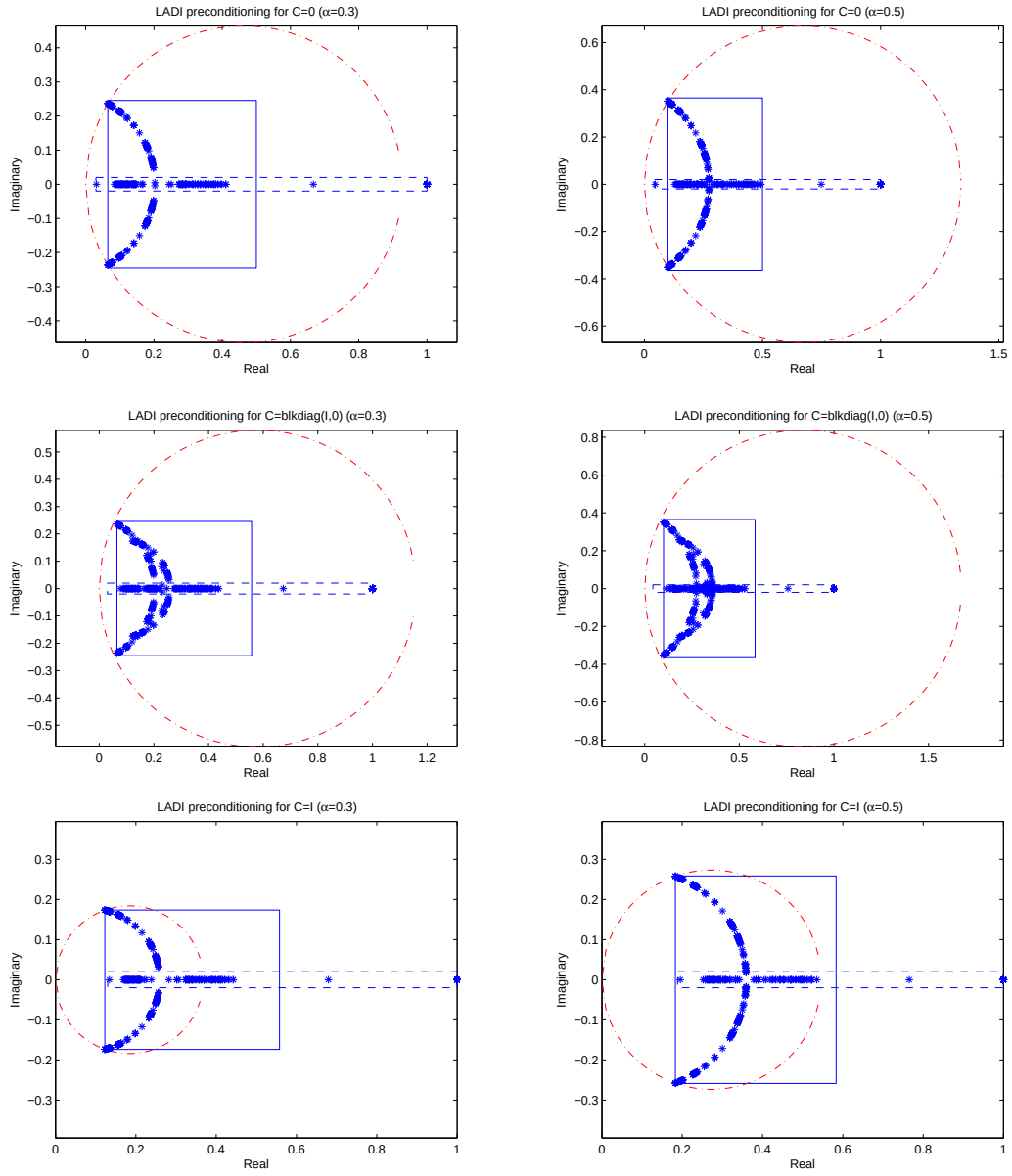


Figure 2: *: eigenvalues of the preconditioned matrix for l=16 for Example 5.2; - -: the region of the real eigenvalues; -: the region of the complex eigenvalues; -.: circle ζ provided in Theorem 4.1 (left $\alpha = 0.3$, right $\alpha = 0.5$).

method is very sensitive to the values of the parameters α and β.

- From the convergence analysis for the LADI method, we know that all eigenvalues of the preconditioned matrix $F(\alpha)^{-1}\mathscr{A}$ are located in a circle centre $(1,0)$ and radius strictly less than 1 for some positive parameters α. From Fig. 2, we see that Theorem 4.1 predicts the location of both real and complex eigenvalues.

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

6. Concluding Remarks

We have constructed and analysed a new LADI-preconditioner method for the generalised saddle point problem (1.1). Both theory and numerical experiments demonstrate that this new approach can significantly improve the convergence property of the GMRES method in solving the generalised saddle point problem (1.1), and the preconditioned GMRES method proved more effective than some traditional methods. However, we have considered only three $(2, 2)$–block cases involving symmetric and positive semi-definite matrices C, and in future we intend to extend the discussion further.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (11271174).

References

[1] O. Axelsson and M. Neytcheva, *Eigenvalue estimates for preconditioned saddle point matrices*, Numer. Linear Algebra Appl. **13**, 339-360 (2006).
[2] Z.Z. Bai, *Structured preconditioners for nonsingular matrices of block two-by-two structures*, Math. Comput. **75**, 791-815 (2006).
[3] Z.Z. Bai, *Optimal parameters in the HSS–like methods for saddle–point problems*, Numer. Linear Algebra Appl. **16**, 447-479 (2009).
[4] Z.Z. Bai, *Motivations and realizations of Krylov subspace methods for large sparse linear systems*, J. Comp. Appl. Math. **283**, 71-78 (2015).
[5] 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).
[6] Z.Z. Bai, G.H. Golub and C.K. Li, *Optimal parameter in Hermitian and skew-Hermitian splitting method for certain two-by-two block matrices*, SIAM J. Sci. Comput. **28**, 583-603 (2006).
[7] Z.Z. Bai, G.H. Golub and C.K. Li, *Convergence properties of preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite matrices*, Math. Comput. **76**, 287-298 (2007).
[8] Z.Z. Bai, G.H. Golub and J.Y. Pan, *Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems*, Numer. Math. **98**, 1-32 (2004).
[9] Z.Z. Bai, G.H. Golub and M.K. Ng, *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl. **24**, 603-626 (2003).
[10] Z.Z. Bai, G.H. Golub and M.K. Ng, *On inexact Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, Linear Algebra Appl. **428**, 413-440 (2008).
[11] Z.Z. Bai, G.H. Golub and M.K. Ng, *On successive-overrelaxation acceleration of the Hermitian and skew-Hermitian splitting iterations*, Numer. Linear Algebra Appl. **14**, 319-335 (2007).
[12] 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).
[13] Z.Z. Bai and Z.Q. Wang, *On parameterized inexact Uzawa methods for generalized saddle point problems*, Linear Algebra Appl. **428**, 2900-2932 (2008).
[14] M. Benzi, G.H. Golub and J. Liesen, *Numerical solution of saddle point problems*, Acta Numer. **14**, 1-137 (2005).

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press

[15] M. Benzi and J. Liu, *Block preconditioning for saddle point systems with indefinite (1, 1) block*, Int. J. Comput. Math. **84**, 1117-1129 (2007).
[16] M. Benzi and V. Simoncini, *On the eigenvalues of a class of saddle point matrices,* Numer. Math. **103**, 173-196 (2006).
[17] L. Bergamaschi, *On eigenvalue distribution of constraint–preconditioned symmetric saddle point matrices*, Numer. Linear Algebra Appl. **19**, 754-772 (2012).
[18] J.H. Bramble, J.E. Pasciak and A.T. Vassilev, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal. **34**, 1072-1092 (1997).
[19] Z.H. Cao, *Augmentation block preconditioners for saddle point-type matrices with singular (1, 1) blocks,* Numer. Linear Algebra Appl. **15**, 515-533 (2008).
[20] Y. Cao, M.Q. Jiang and Y.L. Zheng, *A splitting preconditioner for saddle point problems*, Numer. Linear Algebra Appl. **18**, 875-895 (2011).
[21] L.C. Chan, M.K. Ng and N.K. Tsing, *Spectral Analysis for HSS Preconditioners*, Numer. Math. Theor. Meth. Appl. **1**, 57-77 (2008).
[22] F. Chen and Y.L. Jiang, *A generalization of the inexact parameterized Uzawa methods for saddle point problems*, Appl. Math. Comput. **206**, 765-771 (2008).
[23] H.C. Elman, D.J. Silvester and A.J. Wathen, *Iterative methods for problems in computational fluid dynamics*, in *Iterative Methods in Scientific Computing*, R. H. Chan, C. T. Chan, and G. H. Golub (eds.), pp. 271-327, Springer, Singapore (1997).
[24] H.C. Elman, A. Ramage and D.J. Silvester, *Algorithm 866: IFISS: A Matlab toolbox for modelling incompressible flow*, ACM Trans, Math. Software (TOMS) **33**, 14 (2007).
[25] H.C. Elman, D.J. Silvester and A.J. Wathen, *Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics*, Oxford University Press, Oxford, (2005).
[26] H.H. Harman and W.H. Jones, *Factor analysis by minimizing residuals*, Psychometrika **30**, 351-368 (1963).
[27] C.J. Li, B.J. Li and D.J. Evans, *A generalized successive overrelaxation method for least squares problems*, BIT Numer. Math. **28**, 347-355 (1998).
[28] S.L. Ma and Z.Y. Xu, *Neccessary and sufficient conditions for a polynomial of degree 3 or 4 with real coefficients to be a Von Neumann polynomial*, Numer. Math. J. Chinese Universities. **3**, 274-280 (1984).
[29] Y. Notay, *A new analysis of block preconditioners for saddle point problems*, SIAM J. Matrix Anal. Appl. **35**, 143-173 (2014).
[30] J.Y. Pan, M.K. Ng and Z.Z. Bai, *New preconditioners for saddle point problems*, Appl. Math. Comput. **172**, 762-771 (2006).
[31] I. Perugia and V. Simoncini, *New Block-diagonal and indefinite symmetric preconditioners for mixed finite element formulations*, Numer. Linear Algebra Appl. **7**, 585-616 (2000).
[32] Y. Saad and M.H. Schultz, *GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems*, SIAM J. Sci. Stat. Comput. **7**, 856-869 (1986).
[33] X.H. Shao, Z. Li and C.J. Li, *Modified SOR-like method for the augmented system*, Int. J. Comput. Math. **84**, 1653-1662 (2007).
[34] S. Wright, *Stability of augmented system factorizations in interior-point methods*, SIAM J. Matrix Anal. Appl. **18**, 191-222 (1997).
[35] S.S. Wang and G.F. Zhang, *Preconditioned AHSS iteration method for singular saddle point problems*, Numer. Algor. **63**, 521-535 (2013).
[36] D.M. Young, *Iterative Solution of Large Linear Systems*, Academic Press (1971).
[37] M.Z. Zhu, G.F. Zhang, Z. Zheng and Z.Z. Liang *On HSS-based sequential two-stage method for non-Hermitian saddle point problems*, Appl. Math. Comput. **242**, 907-916 (2014).

https://doi.org/10.4208/eajam.260815.280216a Published online by Cambridge University Press