J. Appl. Math. & Informatics Vol. **36***(2018), No. 5 - 6, pp. 459 - 474*
https://doi.org/10.14317/jami.2018.459

ON A SPLITTING PRECONDITIONER FOR SADDLE POINT PROBLEMS[†]

DAVOD KHOJASTEH SALKUYEH*, MARYAM ABDOLMALEKI, SAEED KARIMI

Abstract. Cao et al. in (Numer. Linear. Algebra Appl. 18 (2011) 875-895) proposed a splitting method for saddle point problems which unconditionally converges to the solution of the system. It was shown that a Krylov subspace method like GMRES in conjunction with the induced preconditioner is very effective for the saddle point problems. In this paper we first modify the iterative method, discuss its convergence properties and apply the induced preconditioner to the problem. Numerical experiments of the corresponding preconditioner are compared to the primitive one to show the superiority of our method.

AMS Mathematics Subject Classification : 65F10, 65F50, 65N22..
Key words and phrases : Saddle point problem, modification, Splitting preconditioner, Indefinite system.

1. Introduction

We consider the large and sparse saddle point problems of the form

$$\mathcal{A}u \equiv \begin{pmatrix} A & B^T \\ -B & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ -g \end{pmatrix} \equiv b, \tag{1}$$

where $A \in \mathbb{R}^{n\times n}$ is symmetric positive definite (SPD) and $B \in \mathbb{R}^{m\times n}$ with $m \leqslant n$, is of full rank. In addition, $x, f \in \mathbb{R}^n$ and $y, g \in \mathbb{R}^m$. It is not difficult to prove that under the above conditions the matrix $\mathcal{A}$ is nonsingular and as a result the system (1) has a unique solution (see [9, Lemma 1.1]). Systems of the form (1) arise in many applications, such as constrained optimization, finite diffrence and mixed finite element discretization of the Navier-stokes equations (see [8, 11, 16]).

Received January 19, 2018. Revised April 27, 2018. Accepted May 2, 2018. *Corresponding author.

[†]The work of Davod Khojateh Salkuyeh is partially supported by University of Guilan.

Recent years, several iterative methods (stationary and nonstationary) have been proposed to solve (1) in the literature. Among them, although the stationary iterative methods are not too effective to be competitive with Krylov-subspace methods like GMRES [21], they serves efficient preconditioners for the system (1). Bai et al. in [4], proposed the Hermitian and skew-Hermitian splitting (HSS) method for solving non-Hermitian positive definite linear systems. Convergence of the HSS method for the saddle point problems was investigated by Benzi and Golub in [9]. As they mentioned the method is typically too slow to be competitive and for this reason they used the induced preconditioner (HSS preconditioner) to accelerate the convergence of a nonsymmetric Krylov subspace method like GMRES, or its restarted version GMRES(ℓ) [21]. Several variants of the HSS preconditioner as well as their relaxed versions have been presented up to now (for example see [1, 2, 6, 7, 24]). Bai et al. in [5] proposed the shift-splitting method to solve non-Hermitian positive definite. Cao et al. in [12], using the idea of [5], proposed the shift-splitting preconditioner for the saddle point problems. Several extensions and improvements have been presented for diffrent types of saddle point problems (see [12, 15, 25]). When $(1,1)$-block A is ill-conditioned or even singular, the augmented Lagrangian method can be employed [8, 18]. In fact, the saddle point system (1) is replaced by the equivalent system

$$\begin{pmatrix} A+\gamma B^TB & B^T \\ -B & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f+\gamma B^Tg \\ -g \end{pmatrix}, \tag{2}$$

where $\gamma \geqslant 0$. As was mentioned in [8], by taking

$$\gamma = \frac{\|A\|_2}{\|B\|_2^2}, \tag{3}$$

the condition number of both the $(1,1)$-block and of the coefficient matrix in (2) are approximately minimized.

In [3], Bai and Wang studied the parameterized inexact Uzawa method (PIU) for solving the saddle point problem (1). Then Chen and Jiang generalized the PIU (GPIU) method for solving the same problem [14]. In [13], Cao et al. studied a special case of the GPIU method and proposed the following stationary iterative method

$$\begin{pmatrix} A+tB^TB & 0 \\ -2B & \frac{1}{t}I \end{pmatrix} \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \begin{pmatrix} tB^TB & -B^T \\ -B & \frac{1}{t}I \end{pmatrix} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \begin{pmatrix} f \\ -g \end{pmatrix}, \tag{4}$$

where $t > 0$. This method can be written as (hereafter we call it GPIU1)

$$u^{(k+1)} = \Gamma_t u^{(k+1)} + c_t, \tag{5}$$

where

$$\Gamma_t = \begin{pmatrix} A+tB^TB & 0 \\ -2B & \frac{1}{t}I \end{pmatrix}^{-1} \begin{pmatrix} tB^TB & -B^T \\ -B & \frac{1}{t}I \end{pmatrix},$$

and

$$c_t = \begin{pmatrix} A + tB^T B & 0 \\ -2B & \frac{1}{t}I \end{pmatrix}^{-1} \begin{pmatrix} f \\ -g \end{pmatrix}.$$

In [13], it was proved that for every $t > 0$ it holds that

$$\rho(\Gamma_t) = \frac{1}{1 + t\sigma_m^2} < 1, \tag{6}$$

where σ_m is the smallest nonzero singular value of the matrix $BA^{-\frac{1}{2}}$ and $\rho(.)$ stands for the spectral radius of the matrix. It means that the iterative method (5) unconditionally converges to the solution of (1) for every initial guess. As usual, the matrix

$$\mathcal{P}_t = \begin{pmatrix} A + tB^T B & 0 \\ -2B & \frac{1}{t}I \end{pmatrix},$$

for every $t > 0$, can be used as a preconditioner for the saddle point problem (1). Numerical results presented in [13] show that the GMRES method or its restarted version, GMRES(ℓ) [21], is very efficient to solve the preconditioned system

$$\mathcal{P}_t^{-1}\mathcal{A}u = \mathcal{P}_t^{-1}b.$$

It was proved that the preconditioned matrix $\mathcal{P}_t^{-1}\mathcal{A}$ has an eigenvalue 1 with multiplicity n and the remaining eigenvalues are equal to

$$\frac{t\sigma_i^2}{1 + t\sigma_i^2}, \quad 1 \leqslant i \leqslant m,$$

where σ_is are the positive singular values of the matrix $BA^{-\frac{1}{2}}$.

From (6), we see that $\rho(\Gamma_t) \to 0$ as t tends to infinity. As mentioned in [8, 13], for large values of t the block $A + tB^T B$ becomes severely ill-conditioned. On the other hand, the $(2, 2)$-block in $\mathcal{P}_t$ is equal to $(1/t)I$, which in this case, approaches to zero and will affect the numerical results when $\mathcal{P}_t$ is used as a preconditioner. Therefore, for large values of t the matrix $\mathcal{P}_t$ would be near to singular. In fact, small values of t affect the convergence of the outer iteration and large values of t make the preconditioned matrix ill-conditioned. Therefore, choosing t so as to minimize the spectral radius of the iteration matrix is not necessarily the best choice, which also holds when the induced preconditioner is utilized to accelerate a Krylov subspace iterative method. Hence, in general finding a good value for t is difficult. In order to improve the efficiency of the method a modification of the stationary iterative method (7) is presented and its convergence properties is studied. We also verify the properties of the induced preconditioner.

This paper is organized as follows. Section 2 is devoted to presenting a modification of the GPIU1 iterative method, its convergence properties and the corresponding preconditioner. In Section 3, we present a strategy for selecting the parameters of proposed method. In Section 4, practical implementation of the extracted preconditioner are presented. Numerical experiments are presented in Section 5 to examine the effectiveness of the proposed method. Finally, the paper is ended by some concluding remarks in Section 6.

2. Modification of the method

We propose the following stationary iteration method

$$\begin{pmatrix} A+\eta\theta B^TB & 0 \\ -(1+\theta)B & \frac{1}{\eta}I \end{pmatrix} \begin{pmatrix} x^{(k+1)} \\ y^{(k+1)} \end{pmatrix} = \begin{pmatrix} \eta\theta B^TB & -B^T \\ -\theta B & \frac{1}{\eta}I \end{pmatrix} \begin{pmatrix} x^{(k)} \\ y^{(k)} \end{pmatrix} + \begin{pmatrix} f \\ -g \end{pmatrix}, \tag{7}$$

for solving the system (1), where $\eta, \theta > 0$. This splitting leads to the following stationary iterative method (hereafter we denote it by GPIU2)

$$u^{(k+1)} = \Upsilon_{\eta,\theta} u^{(k)} + d_{\eta,\theta}, \tag{8}$$

where

$$\Upsilon_{\eta,\theta} = \begin{pmatrix} A+\eta\theta B^TB & 0 \\ -(1+\theta)B & \frac{1}{\eta}I \end{pmatrix}^{-1} \begin{pmatrix} \eta\theta B^TB & -B^T \\ -\theta B & \frac{1}{\eta}I \end{pmatrix}$$

and

$$d_{\eta,\theta} = \begin{pmatrix} A+\eta\theta B^TB & 0 \\ -(1+\theta)B & \frac{1}{\eta}I \end{pmatrix}^{-1} \begin{pmatrix} f \\ -g \end{pmatrix}.$$

Obviously, for $\theta = 1$ the GPIU2 method coincides with GPIU1. The next theorem verifies the convergence of the method.

Theorem 2.1. *Let $A \in \mathbb{R}^{n\times n}$ be symmetric positive definite and $B \in \mathbb{R}^{m\times n}$ be of full rank. Then $\Upsilon_{\eta,\theta}$ has an eigenvalue 0 of multiplicity n, i.e. $\mu_i^{(1)} = 0$, $i = 1, 2, \ldots, n$, and remaining other m eigenvalues are of the form*

$$\mu_i^{(2)} = 1 - \frac{\eta\sigma_i^2}{1+\eta\theta\sigma_i^2}, \quad i = 1, 2, \ldots, m, \tag{9}$$

where $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_m$, are the nonzero singular values of $BA^{-\frac{1}{2}}$. Moreover, $\rho(\Upsilon_{\eta,\theta}) < 1$ provided that $\eta > 0$ and

$$\theta > \max\{0, \frac{1}{2} - \frac{1}{\eta{\sigma_1}^2}\}.$$

Proof. The proof of this theorem is similar to that of Theorem 3.1 in [13]. Therefore, we present a sketch of the proof. It is not difficult to show that

$$\Upsilon_{\eta,\theta} = \begin{pmatrix} \eta\theta\mathcal{S}^{-1}B^TB & -\mathcal{S}^{-1}B^T \\ \eta^2\theta(1+\theta)B\mathcal{S}^{-1}B^TB - \eta\theta B & I - \eta(1+\theta)B\mathcal{S}^{-1}B^T \end{pmatrix},$$

wherein $\mathcal{S} = A + \eta\theta B^TB$. We set $\tilde{\Upsilon}_{\eta,\theta} = F_0\Upsilon_{\eta,\theta}F_0^{-1}$ where

$$F_0 = \begin{pmatrix} A^{\frac{1}{2}} & 0 \\ 0 & I \end{pmatrix}.$$

Let $BA^{-\frac{1}{2}} = U^T[\Sigma\ 0]V$ be the singular value decomposition of $BA^{-\frac{1}{2}}$, where U and V are orthogonal matrices and $\Sigma = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_m)$. It can be shown that

$$\tilde{\Upsilon}_{\eta,\theta} = \begin{pmatrix} V & 0 \\ 0 & U \end{pmatrix}^T \hat{\Upsilon}_{s,t} \begin{pmatrix} V & 0 \\ 0 & U \end{pmatrix},$$

where

$$\begin{aligned}
\hat{\Upsilon}_{\eta,\theta} &= \begin{pmatrix} \eta\theta(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 & 0 & -(I+\eta\theta\Sigma^2)^{-1}\Sigma \\ 0 & 0 & 0 \\ \eta^2\theta(1+\theta)\Sigma(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 - \eta\theta\Sigma & 0 & I-\eta(1+\theta)\Sigma(I+\eta\theta\Sigma^2)^{-1}\Sigma \end{pmatrix} \\
&= F_1 \begin{pmatrix} \eta\theta(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 & 0 & -(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 \\ 0 & 0 & 0 \\ \eta^2\theta(1+\theta)(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 - \eta\theta I & 0 & I-\eta(1+\theta)(I+\eta\theta\Sigma^2)^{-1}\Sigma^2 \end{pmatrix} F_1^{-1} \\
&= F_1 \bar{\Upsilon}_{\eta,\theta} F_1^{-1},
\end{aligned}$$

in which

$$F_1 = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & \Sigma \end{pmatrix}.$$

Therefore, we deduce that the matrices $\Upsilon_{\eta,\theta}$ and $\bar{\Upsilon}_{\eta,\theta}$ are similar. Hence, all we need is to compute the eigenvalues of $\bar{\Upsilon}_{\eta,\theta}$. Since the nonzero blocks of this matrix are diagonal, we only need to compute the eigenvalues of the matrices

$$\begin{aligned}
\mathcal{H}_{\eta,\theta}(\sigma_i) &= \begin{pmatrix} \eta\theta(1+\eta\theta\sigma_i^2)^{-1}\sigma_i^2 & -(1+\eta\theta\sigma_i^2)^{-1}\sigma_i^2 \\ \eta^2\theta(1+\theta)(1+\eta\theta\sigma_i^2)^{-1}\sigma_i^2 - \eta\theta & 1-\eta(1+\theta)(1+\eta\theta\sigma_i^2)^{-1}\sigma_i^2 \end{pmatrix} \\
&= \frac{\sigma_i^2}{1+\eta\theta\sigma_i^2} \begin{pmatrix} \eta\theta & -1 \\ \eta\theta(\eta - \frac{1}{\sigma_i^2}) & \frac{1}{\sigma_i^2} - \eta \end{pmatrix},
\end{aligned}$$

for $i = 1, 2, \ldots, m$. It is straightforward to see that the eigenvalues of $\mathcal{H}_{\eta,\theta}(\sigma_i)$ are $\mu_i^{(1)} = 0$ and

$$\mu_i^{(2)} = 1 - \frac{\eta\sigma_i^2}{1+\eta\theta\sigma_i^2}, \quad i = 1, 2, \ldots, m. \tag{10}$$

Summarizing the above results we conclude that the matrix $\Upsilon_{\eta,\theta}$ has an eigenvalue 0 of multiplicity n and the remaining m eigenvalues are of the form (10). Now, for $\eta > 0$ it is easy to see that $|\mu_i^{(2)}| < 1$ if and only if

$$\theta > \frac{1}{2} - \frac{1}{\eta\sigma_i^2}.$$

Therefore, if $\eta > 0$ and $\theta > \max\{0, 1/2 - 1/(\eta\sigma_1^2)\}$, then $\rho(\Upsilon_{\eta,\theta}) < 1$. □

Remark 2.1. Let A be symmetric positive definite and B be of full rank. For every $\eta > 0$ and $\theta \geqslant 1/2$, from Theorem 2.1 it follows that $\rho(\Upsilon_{\eta,\theta}) < 1$.

In general, convergence of the stationary iterative methods like (8) are typically too slow for the method to be competitive. However, the iterative method (8) serves the preconditioner

$$\mathcal{Q}_{\eta,\theta} = \begin{pmatrix} A + \eta\theta B^T B & 0 \\ -(1+\theta)B & \frac{1}{\eta} I \end{pmatrix},$$

for the system $\mathcal{A}u = b$ and the preconditioned system $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}u = \mathcal{Q}_{\eta,\theta}^{-1}b$ (or $\mathcal{A}\mathcal{Q}_{\eta,\theta}^{-1}v = b$ with $u = \mathcal{Q}_{\eta,\theta}^{-1}v$) can be solved by a Krylov subspace method like GMRES or its restarted version. It is noted that the eigenvalue distribution of matrices $\mathcal{A}\mathcal{Q}_{\eta,\theta}^{-1}$ and $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}$ are the same. In contrast to the GIPU1 method, we can independently choose the coefficient of the matrices $B^T B$ in $(1,1)$-block and I in $(2,2)$-block. In fact, two parameters η and θ are chosen so that they balance the coefficient of these matrices in the preconditioner $\mathcal{Q}_{\eta,\theta}$.

Setting $\mathcal{R}_{\eta,\theta} = \mathcal{Q}_{\eta,\theta} - \mathcal{A}$, we have $\mathcal{A} = \mathcal{Q}_{\eta,\theta} - \mathcal{R}_{\eta,\theta}$. Therefore,

$$\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A} = I - \mathcal{Q}_{\eta,\theta}^{-1}\mathcal{R}_{\eta,\theta} = I - \Upsilon_{\eta,\theta}.$$

This shows that the eigenvalues of $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}$ are of the form $\mu = 1 - \lambda$, where $\lambda \in \sigma(\Upsilon_{\eta,\theta})$, in which $\sigma(.)$ stands for the spectrum of the matrix. Hence, from (9) we conclude that the preconditioned matrix $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}$ has eigenvalue 1 with multiplicity n and the remaining eigenvalues are

$$\lambda_i^{(2)} = \frac{\eta\sigma_i^2}{1 + \eta\theta\sigma_i^2}, \quad i = 1, 2, \ldots, m. \tag{11}$$

3. Selection of the parameters

As we see the scalar coefficient of the matrix $B^T B$ in $(1,1)$-block of the GPIU1 and GPIU2 preconditioners are t and $\eta\theta$, respectively. In the implementation of these preconditioners we need to solve systems of form

$$(A + \delta B^T B)z = r, \tag{12}$$

where $\delta = t$ and $\delta = \eta\theta$ in the GPIU1 and the GPIU2 preconditioners, respectively. For large problems these systems are solved by the CG method or its preconditioned version, PCG. As already mentioned, the value

$$\delta^* = \frac{\|A\|_2}{\|B\|_2^2},$$

is a good choice for δ. Therefore, in the GPIU1 preconditioner we set $t = \delta^*$. In the GPIU2 preconditioner we set $\eta\theta = \delta^*$ and compute the values of η and θ such that the spectral radius of the GPIU2 iteration matrix is minimized. In fact, using this strategy not only the iteration counts for solving the system (12) is approximately minimized but also the outer iteration of the method is also improved.

From Eq. (9), we see that if we set $\eta\theta = \delta^*$, then the nonzero eigenvalues of the GPIU2 iteration matrix would be of the form

$$\mu_i^{(2)} = 1 - \frac{\eta\sigma_i^2}{1+\delta^*\sigma_i^2}, \quad i = 1, 2, \ldots, m. \tag{13}$$

Now all we need is to seek the parameter η^* such that

$$\eta^* = \underset{\eta}{\operatorname{argmin}} \max_{\sigma_i} \left| 1 - \frac{\eta\sigma_i^2}{1+\delta^*\sigma_i^2} \right|. \tag{14}$$

In fact for fixed $\eta\theta = \delta^*$ the value η^* makes the spectral radius of the GPIU2 iteration matrix minimized. In other words, from Eq. (11), we see that the value η^* is chosen such a way that the eigenvalues of the preconditioned matrix $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}$ are well-clustered around the point $(1, 0)$. The following theorem gives the value of η^*.

Theorem 3.1. *Let the conditions of Theorem 2.1 hold and $\Upsilon_{\eta,\theta}$ be the iteration matrix of GPIU2 iterative method. If $\eta\theta = \delta^*$ be a positive constant, then the optimal iteration parameters η^* and θ^* of the GPIU2 iteration method that minimize the spectral radius $\rho(\Upsilon_{\eta,\theta})$ are given by*

$$\eta^* = \frac{2(1+\delta^*\sigma_1^2)(1+\delta^*\sigma_m^2)}{\sigma_1^2(1+\delta^*\sigma_m^2)+\sigma_m^2(1+\delta^*\sigma_1^2)} \quad \text{and} \quad \theta^* = \frac{\delta^*}{\eta^*}, \tag{15}$$

wherein σ_1 and σ_m are the largest and smallest singular values of $BA^{-\frac{1}{2}}$, respectively. Moreover, the corresponding optimal spectral radius is given by

$$\rho(\Upsilon_{\eta^*,\theta^*}) = \frac{1-k}{1+k} < 1, \tag{16}$$

where

$$k = \frac{\sigma_m^2(1+\delta^*\sigma_1^2)}{\sigma_1^2(1+\delta^*\sigma_m^2)}.$$

Proof. For the sake of the simplicity, let

$$\omega_i = \frac{\sigma_i^2}{1+\delta^*\sigma_i^2}, \quad i = 1, 2, \ldots, m. \tag{17}$$

It is easy to see that $\omega_1 \geqslant \omega_2 \geqslant \cdots \geqslant \omega_m > 0$. Hence,

$$\max_{\sigma_i} \left| 1 - \eta\frac{\sigma_i^2}{1+\delta^*\sigma_i^2} \right| = \max\left\{ |1-\eta\omega_1|, |1-\eta\omega_m| \right\}.$$

Hence, from Eq. (14) we deduce that

$$\eta^* = \underset{\eta}{\operatorname{argmin}} \max\left\{ |1-\eta\omega_1|, |1-\eta\omega_m| \right\}. \tag{18}$$

From the latter equation it is not difficult to see that the optimal value of the η satisfies

$$1 - \eta^*\omega_1 = -\left(1 - \eta^*\omega_m\right).$$

Hence,

$$\eta^* = \frac{2}{\omega_1 + \omega_m}.$$

Substituting ω_1 and ω_m from Eq. (17) in the latter equation, the first relation in (15) is obtained. The second relation in (15) follows from $\eta^*\theta^* = \delta^*$. Finally, we have

$$\rho(\Upsilon_{\eta^*,\theta^*}) = \left|1 - \eta^* \frac{\sigma_1^2}{1 + t^*\sigma_1^2}\right| = \left|1 - \eta^* \frac{\sigma_1^2}{1 + \eta^*\theta^*\sigma_1^2}\right|,$$

which proves Eq. (16). □

From Theorem 3.1 we see that for computing the values of η^* and $\theta*$ we need to compute the largest and smallest singular values of $BA^{-\frac{1}{2}}$ which is impractical for large problems. In the sequel, we compare the spectral radius of the GPIU1 and GPIU2 iteration matrices for arbitrary choices of η and θ. Let $\delta = \eta\theta$ be a positive constant and $\eta > 0$. As we mentioned, for $\theta = 1$ the GPIU2 method coincides with GPIU1, i.e., $\rho(\Upsilon_{\eta,1}) = \rho(\Gamma_t)$. According to Theorem 3.1, we have

$$\rho(\Upsilon_{\eta,\theta}) = \frac{{\sigma_1}^2(1 + \delta{\sigma_m}^2) - {\sigma_m}^2(1 + \delta{\sigma_1}^2)}{{\sigma_1}^2(1 + \delta{\sigma_m}^2) + {\sigma_m}^2(1 + \delta{\sigma_1}^2)}.$$

It is not difficult to prove the following remark.

Remark 3.1. (i) If $0.5 \leqslant \theta < 1$, then $\rho(\Upsilon_{\eta,\theta}) < \rho(\Gamma_t) = \rho(\Upsilon_{\eta,1})$.
(ii) If $\theta > 1$, then $\rho(\Upsilon_{\eta,1}) < \rho(\Upsilon_{\eta,\theta})$.
(iii) If $0 < \theta < 0.5$ then the superiority of the GPIU2 method to GPIU1 can not be deduced theoretically. However, the numerical results show that GPIU2 outperforms the GPIU1 method for the values η^* and θ^* defined in Theorem 3.1.

4. Practical implementation

In the section of the numerical experiments we solve the system $\mathcal{A}u = b$ by the preconditioned GMRES(ℓ) in conjunction with preconditioner $\mathcal{Q}_{\eta,\theta}$. We use right preconditioning. It is noted that the eigenvalues of $\mathcal{Q}_{\eta,\theta}^{-1}\mathcal{A}$ and $\mathcal{A}\mathcal{Q}_{\eta,\theta}^{-1}$ are the same. Indeed, we apply the GMRES(ℓ) method for solving the preconditioned system $\mathcal{A}\mathcal{Q}_{\eta,\theta}^{-1}v = b$ with $u = \mathcal{Q}_{\eta,\theta}^{-1}v$. We use a zero vector as an initial guess and the stopping criterion

$$R_k = \frac{\|r^{(k)}\|_2}{\|r^{(0)}\|_2} < tol, \tag{19}$$

where $r^{(k)} = b - \mathcal{A}u^{(k)}$ ($u^{(k)}$ is the computed solution) and $r^{(0)} = b - \mathcal{A}u^{(0)}$. The maximum number of restarts is set to *maxrest*. In each restart of the method, we need to compute an orthonormal basis for a Krylov subspace and during this process we need to compute some vectors of the form $(z_1; z_2) = \mathcal{Q}_{\eta,\theta}^{-1}(r_1; r_2)$, where $r_1 \in \mathbb{R}^m$ and $r_2 \in \mathbb{R}^n$. To do so, it is enough to solve the system $\mathcal{Q}_{\eta,\theta}(z_1; z_2) = (r_1; r_2)$ for $(z_1; z_2)$ and this can be done by the following algorithm.

Algorithm 1. Computation of $(z_1; z_2) = \mathcal{Q}_{\eta,\theta}^{-1}(r_1; r_2)$.

```
(1) Solve $(A + \eta\theta B^T B)z_1 = r_1$ by the CG method
(2) Compute $z_2 = \eta(r_2 + (1 + \theta)Bz_1)$
```

In solving the system $(A + \eta\theta B^T B)z_1 = r_1$ by the CG method, the initial guess is set to be a zero vector and the iteration is terminated as soon as

$$\frac{\|r_1 - (A + \eta\theta B^T B)z_1^{(k)}\|_2}{\|r_1\|_2} < tol_{in},$$

or the number of the iterations exceeds "$maxit_{in}$". Since, for $\theta = 1$ the GPIU2 reduces to the GPIU1 method, the same strategy is used in the implementation of the GPIU1 method.

To compute the value of δ^* we use `normest(A,0.001)` and `normest(B,0.001)` commands of Matlab, respectively, for estimating $\|A\|_2$ and $\|B\|_2$. Also, from (15), for computing η^* we need to compute the smallest and the largest singular value of $BA^{-\frac{1}{2}}$. To do so, we need to compute the extreme eigenvalues of the matrix $A^{-\frac{1}{2}}B^T BA^{-\frac{1}{2}}$ which is similar to $A^{-1}B^T B$. Therefore, since their eigenvalues are the same we compute the largest and smallest eigenvalues of the matrix $A^{-1}B^T B$ and their square root give the extreme singular values of $BA^{-\frac{1}{2}}$.

5. Numerical Experiments

We present some numerical experiments to compare the effectiveness of the GPIU1 and the GPIU2 preconditioners. All the numerical experiments were computed in double precision using some Matlab codes on a Laptop with Intel Core i5 CPU 2.40 GHz, 8GB RAM. In all experiments, the right-hand side vector b is set to $b = Au^*$ where u^* is a vector of all ones. According to the implementation method described in Section 4, we always set

$$tol = 10^{-9}, \quad maxrest = 10000, \quad tol_{in} = 10^{-6}, \quad \text{and} \quad maxit_{in} = 200,$$

unless it is mentioned. Numerical results are presented in the tables. In the tables, "Iters" and "CPU" stand for the number of iterations and the CPU time (in seconds) for the convergence of the methods, respectively. Also, the values of R_k and E_k are presented where R_k was defined in (19) and $E_k = \|u^{(k)} - u^*\|_\infty$. Moreover, we use the `gmres` function of Matlab with $restart = 5$. Let "$It = (i_1, i_2)$" be the number of restarts returned by the GMRES method. In this case, the value of Iters presented in the table are computed via $Iters = restart \times (i_1 - 1) + i_2$.

Example 5.1. We consider the Stokes problem

$$\begin{cases} -\nu\triangle\mathbf{u}+\nabla p=\mathbf{f}, & \text{in } \Omega,\\ \text{div }\mathbf{u}=g, & \text{in } \Omega,\\ \mathbf{u}=0, & \text{on } \partial\Omega,\\ \displaystyle\int_\Omega p dx=0, & \end{cases} \tag{20}$$

where $\Omega=(-1,1)\times(-1,1)$, $\partial\Omega$ is the boundary of the domain Ω, $\triangle$ is the componentwise Laplace operator, $\mathbf{u}$ is a vector-valued function representing the velocity, p is a scalar function representing the pressure and $\nu>0$ the viscosity constant. By discretizing with the upwind scheme, we obtain the saddle point problem (1) in which (see [13])

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

where

$$T=\frac{\nu}{h^2}\cdot\text{tridiag}(-1,2,-1)\in\mathbb{R}^{q\times q},$$
$$F=\frac{1}{h}\cdot\text{tridiag}(-1,1,0)\in\mathbb{R}^{q\times q}.$$

Here $\otimes$ denotes the Kronecker product and $h=1/(q+1)$. In this case $n=2q^2$ and $m=q^2$. We set $\nu=0.001$ and the numerical results are given for $q=16,\ 32,\ 64$. In Table 1 we present the values of n, m, η^*, θ^* and t^* for Example 5.1.

Numerical results are presented in Table 2. As we observe both of the GPIU1 and GPIU2 preconditioners are effective in reducing the number of iterations and CPU timing. However, we see that the GPIU2 preconditioner outperforms the GPIU1 preconditioner from both the iteration counts and the CPU time.

Figure 1 displays the eigenvalue distribution of the matrices $\mathcal{P}_{t^*}^{-1}\mathcal{A}$ and $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$. As we observe the eigenvalues of $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$ are more clustered around $(1,0)$ than the matrix $\mathcal{P}_{t^*}^{-1}\mathcal{A}$. From the theoretical results presented in the previous section we saw that the eigenvalues of these matrices are all real. However, from Figure 1, we see that the imaginary part of some the eigenvalues are not zero. This is due to roundoff errors in the computational process.

Example 5.2. Consider the ossen problem which is obtained from the linearization of the following steady$-$state Navier$-$Stokes equation by the Picard iteration with suitable boundary condition on $\partial\Omega$ (see [19])

$$\begin{cases} -\nu\Delta\mathbf{u}+(\mathbf{w}.\nabla)\mathbf{u}+\nabla p &=\mathbf{f}, \quad \text{in } \Omega,\\ \text{div}\mathbf{u} &=0, \quad \text{in } \Omega,\\ \mathbf{u} &=\mathbf{g}, \quad \text{on } \partial\Omega,\end{cases} \tag{21}$$

TABLE 1. n, m, η^*, θ^* and t^* for Example 5.1

q			GPIU2 Preconditioner		GPIU1 Preconditioner
	n	m	η^*	θ^*	t^*
16	512	256	0.003	0.293	0.001
32	2048	1024	0.004	0.277	0.001
64	8192	4096	0.004	0.266	0.001

TABLE 2. Numerical results of GMRES(5) without preconditioning and with the GPIU1 and the GPIU2 preconditioners for Example 5.1

q		GPIU2 preconditioner	GPIU1 preconditioner	GMRES(5)
16	Iters	24	25	15195
	CPU	0.14	0.20	1.62
	R_k	8.01×10^{-10}	1.42×10^{-10}	1.00×10^{-9}
	E_k	1.54×10^{-9}	3.69×10^{-10}	1.71×10^{-7}
32	Iters	25	28	26650
	CPU	0.43	0.56	7.39
	R_k	5.93×10^{-10}	9.67×10^{-10}	1.00×10^{-9}
	E_k	2.09×10^{-9}	4.77×10^{-9}	1.90×10^{-7}
64	Iters	29	44	49524
	CPU	2.46	3.92	37.71
	R_k	6.25×10^{-10}	5.20×10^{-10}	1.00×10^{-9}
	E_k	3.84×10^{-9}	3.27×10^{-9}	1.66×10^{-7}

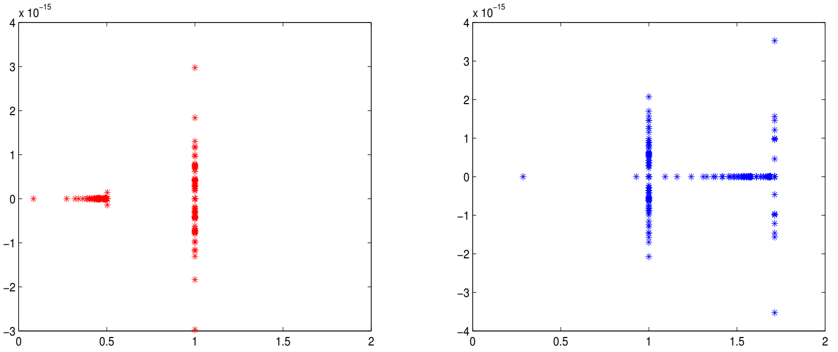

FIGURE 1. Eigenvalue distribution of the matrices $\mathcal{P}_{t^*}^{-1}\mathcal{A}$ (left) and $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$ (right) for Example 5.1 with $q = 16$

Table 3. n, m, η^*, θ^* and t^* for Example 5.2

q			GPIU2 Preconditioner		GPIU1 Preconditioner
	n	m	η^*	θ^*	t^*
16	578	189	77.228	0.468	36.145
32	2178	765	322.085	0.449	144.495
64	8450	3069	2079.109	0.477	991.889

where $\nu > 0$, Δ, ∇, div, $\mathbf{u}$, and p stand for the Laplace operator, the gradient operator, the divergence, the velocity and pressure of the fluid, respectively. Here the vector field $\mathbf{w}$ is the approximation of $\mathbf{u}$ from the previous Picard iteration. Many approximation schemes can be applied to discretize the Oseen problem (21) leading to a saddle point system of type (1). We consider a leaky two-dimensional lid-driven cavity problem discretized by Q2-P1 finite element on uniform grids on the unit square. The test problem was generated by using the IFISS software package written by Elman et al. [17]. We use the viscosity value $\nu = 1$ to generate linear systems corresponding to $16 \times 16, 32 \times 32$ and 64×64 meshes. The values of n, m, η^*, θ^* and t^* for Example 5.2 are presented in Table 3.

Numerical results are presented in Table 4. As seen, in all cases the GPIU2 preconditioner behave better than the GPIU1 preconditioner from both of iterations and CPU time point of view. The cause of such performance is predictable by considering the way of choosing the parameters of GPIU2. It should be mentioned that for $q = 32$ and $q = 64$ the GMRES(5) method without preconditioning fails to converge in 10000 restarts.

Figure 5.2 shows the eigenvalue distribution of the matrices $\mathcal{P}_{t^*}^{-1}\mathcal{A}$ and $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$. From this figure we see that the eigenvalues of $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$ are more clustered around $(1, 0)$ than the matrix $\mathcal{P}_{t^*}^{-1}\mathcal{A}$.

According to the Remark 3.1, we expect that the GPIU2 outperforms GPIU1 for $\theta \in [0.5, 1)$. To corroborate this matter numerically we consider the probems of Examples 5.1 and 5.2 with $q = 128$. Here, we have $n = 32768$ and $m = 16384$. In Example 5.2 the number inner iterations is set to be 1000. We use $\delta = \|A\|_2/\|B\|_2^2$ and compute $\eta = \delta/\theta$ with various values of θ in the interval $[0.5, 1)$. Numerical results of Flexible version of GMRES(5) (see [22, 23]) in conjunction with the GPIU1 and GPIU2 preconditioners are shown in Table 5. As we see, in all the cases, the GPIU2 preconditioner has provided quite suitable results and this results are in good agreement with what we claimed in Remark 3.1.

6. Conclusion

Wang and Bai [3] proposed the parameterized inexact Uzawa method (PIU) for solving the saddle point problems. Then a modification of the PIU method

TABLE 4. Numerical results of GMRES(5) without preconditioning and with the GPIU1 and the GPIU2 preconditioners for Example 5.2

q		GPIU2 preconditioner	GPIU1 preconditioner	GMRES(5)
16	Iters	41	45	13361
	CPU	0.23	0.32	1.50
	R_k	5.37×10^{-10}	7.34×10^{-10}	1.00×10^{-9}
	E_k	7.55×10^{-7}	1.23×10^{-6}	8.14×10^{-6}
32	Iters	30	34	†
	CPU	0.58	0.76	-
	R_k	9.87×10^{-10}	8.33×10^{-10}	-
	E_k	1.67×10^{-5}	3.28×10^{-5}	-
64	Iters	45	75	†
	CPU	5.69	10.03	-
	R_k	6.02×10^{-10}	8.26×10^{-10}	-
	E_k	5.97×10^{-5}	3.73×10^{-4}	-

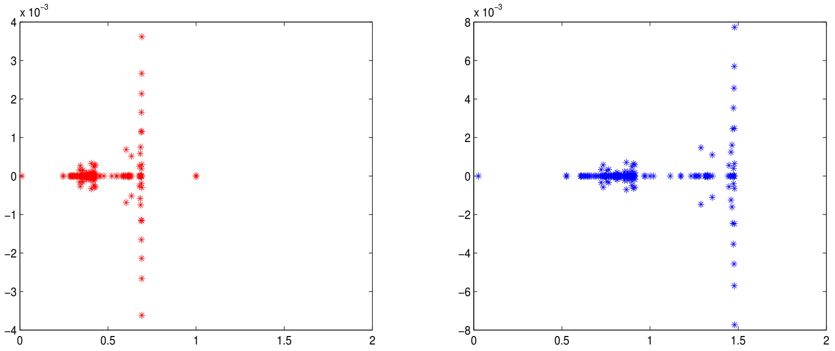

FIGURE 2. Eigenvalue distribution of the matrices $\mathcal{P}_{t^*}^{-1}\mathcal{A}$ (left) and $\mathcal{Q}_{\eta^*,\theta^*}^{-1}\mathcal{A}$ (right) for Example 5.2 with $q = 16$

say (GPIU) was presented for solving the same problem [14]. Cao et al. in [13] studied a special case of the GPIU method and showed that the induced preconditioner is a very efficient. We have presented a modification of the GPIU1 method say GPIU2 and investigated its convergence properties. We proved that the proposed method is always convergent under some mild conditions. Corresponding to any parameter t^* of the GPIU1, we have presented a strategy to choose the parameters of the GPIU2 preconditioner. We have examined the GPIU1 and the GPIU2 preconditioners to accelerate the converge speed of the GMRES(ℓ) or its Flexible version. The presented numerical experiments show

Table 5. Numerical results of Flexible GMRES(5) with the GPIU1 and GPIU2 preconditioners for the Examples 5.1 and 5.2 for $q = 128$

Example 5.1					
GPIU2					
θ	η	Iters	CPU	R_k	E_k
0.5	0.0020	50	6.75	6.17×10^{-10}	1.95×10^{-6}
0.6	0.0017	54	7.88	9.36×10^{-10}	2.95×10^{-6}
0.7	0.0015	66	9.57	8.64×10^{-10}	2.73×10^{-6}
0.8	0.0013	80	10.82	5.09×10^{-10}	1.61×10^{-6}
0.9	0.0011	82	11.03	8.16×10^{-10}	2.57×10^{-6}
GPIU1					
	t	Iters	CPU	R_k	E_k
	0.0010	101	14.88	7.05×10^{-10}	2.22×10^{-6}
Example 5.2					
GPIU2					
θ	η	Iters	CPU	R_k	E_k
0.5	7945.5212	60	35.20	3.71×10^{-10}	2.09×10^{-8}
0.6	6621.2676	50	30.40	8.36×10^{-10}	4.70×10^{-8}
0.7	5675.3723	51	29.89	9.38×10^{-10}	5.28×10^{-8}
0.8	4965.9507	60	36.41	7.12×10^{-10}	4.01×10^{-8}
0.9	4414.1784	87	54.13	9.26×10^{-10}	5.21×10^{-8}
GPIU1					
	t	Iters	CPU	R_k	E_k
	3972.7606	97	62.09	9.43×10^{-10}	5.30×10^{-8}

that the GPIU2 preconditioner outperforms the GPIU1 preconditioner for the presented test examples.

Acknowledgments

The authors would like to thank the referee for his/her helpful comments and suggestions.

References

1. Z.Z. Bai, *Optimal parameters in the HSS-like methods for saddle-point problems*, Numer. Linear Algebra Appl. **16** (2009), 447-479.
2. Z.Z. Bai and G.H. Golub, *Accelerated Hermitian and skew-Hermitian splitting methods for saddle-point problems*, IMA J. Numer. Anal. **27** (2007), 1-23.
3. Z.Z. Bai and Z.Q. Wang, *On parameterized inexact Uzawa methods for generalized saddle point problems*, Linear Algebra Appl. **428** (2008), 2900-2932.
4. Z.Z. Bai, G.H. Golub and M.K. Ng, *Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems*, SIAM J. Matrix Anal. Appl. **24** (2003), 603-626.
5. Z.Z. Bai, J.F. Yin and Y.F. Su, *A shift-splitting preconditioner for non-Hermitian positive definite matrics*, J. Comput. Math. **24** (2006), 539-552.
6. M. Benzi and X.P. Guo, *A dimensional split preconditioner for Stokes and linearized Navier-Stokes equations*, Appl. Numer. Math. **61** (2011), 66-76.
7. M. Benzi, M.K. Ng, Q. Niu and Z. Wang, *A relaxed dimensional fractorization preconditioner for the incompressible Navier-Stokes equations*, J. Comput. Phys. **230** (2011), 6185-6202.
8. M. Benzi, G.H. Golub and J. Liesen, *Numerical solution of saddle point problems*, Acta Numerica **14** (2005), 1-137.
9. M. Benzi and G.H. Golub, *A preconditioner for generalized saddle point problems*, SIAM J. Matrix Anal. Appl. **26** (2004), 20-41.
10. H. Bramble, J.E. Pasciak and A.T. Vassilev, *Analysis of the inexact Uzawa algorithm for saddle point problems*, SIAM J. Numer. Anal. **34** (1997), 1072-1092.
11. F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Element Methods, Springer*, New York, 1991.
12. Y. Cao, J. Du and Q. Niu, *Shift-splitting preconditioners for saddle point problems*, J. Comput. Appl. Math. **272** (2014), 239-250.
13. Y. Cao, M.Q. Jiang and Y.L. Zheng, *A splitting preconditioner for saddle point problems*, Numer. Linear Algebra Appl. **18** (2011), 875-895.
14. F. Chen and Y.-L. Jiang, *A generalization of the inexact parameterized Uzawa methods for saddle point problems*, Appl. Math. Comput. **206** (2008), 765-771.
15. C. Chen and C. Ma, *A generalized shift-splitting preconditioner for saddle point problems*, Appl. Math. Lett. **43** (2015), 49-55.
16. H.C. Elman, D.J. Silvester and A.J. Wathen, *Finite Elements and Fast Iterative Solvers*, Oxford University Press, Oxford, 2003.
17. H.C. Elman, A. Ramage and D.J. Silvester, *A* MATLAB *toolbox for modelling incompressible How*, ACM Trans. Math. Software. **33** (2007), Article 14.
18. G.H. Golub and C. Greif, *On solving block-structured indefinite linear systems*, SIAM J. Sci. Comput. **24** (2003), 2076–2092.
19. D. Hezari, V. Edalatpour, H. Feyzollahzadeh and D.K. Salkuyeh, *On the generalized deteriorated positive semi-definite and skew-hermitian spliting preconditioner*, Journal of Computational Mathematics, to appear, 2018.
20. Q. Hu and J. Zou,*An iterative method with variable relaxation parameters for saddle-point problems*, SIAM J. Matrix Anal. Appl. **23** (2001), 317–338.
21. Y. Saad and M.H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. **7** (1986), 856–869.
22. Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Comput. **14** (1993), 461–469.
23. Y. Saad, *Iterative methods for sparse linear systems*, Second edition PWS, New York, 1995.
24. D.K. Salkuyeh and M. Masoudi, *A new relaxed HSS preconditioner for saddle point problems*, Numer. Algor. **74** (2017), 781795.

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

Davod Khojasteh Salkuyeh received his Ph.D degree from Ferdowsi University of Mashhad in 2003 under supervision of professor Faezeh Toutounian. He is currently a professor of Mathematics at University of Guilan. His research interests are mainly iterative methods for large sparse linear systems of equations and preconditioning.

Faculty of Mathematical Sciences, University of Guilan, Rasht, Iran
e-mail: khojasteh@guilan.ac.ir

Maryam Abdolmaleki received her M.Sc. degree from University of Guilan. She is currently a Ph.D student at Persian Gulf University under supervision of Dr. Saeed Karimi and Dr. Davod Khojasteh Salkuyeh. Her research interests are mainly iterative methods for large sparse linear systems of equations and preconditioning.

Department of Mathematics, Persian Gulf University, Bushehr, Iran
e-mail: maleki.un@gmail.com

Saeed Karimi received his Ph.D degree under supervision of professor Faezeh Toutounian at Ferdowsi University of Mashhad in 2006. He is currently an associate professor of Mathematics at Persian Gulf University, Bushehr, Iran. His research interests are mainly iterative methods for sparse linear system of equations and finite element method.

Department of Mathematics, Persian Gulf University, Bushehr, Iran
e-mail: karimi@pgu.ac.ir