Performance Analysis of a Special GPIU Method for Singular Saddle Point Problems

Jae Heon Yun, *Member, IAENG*

***Abstract*—In this paper, we first provide semi-convergence analysis for a special GPIU(Generalized Parameterized Inexact Uzawa) method with singular preconditioners for solving singular saddle point problems. We next provide a methodology of how to choose nearly quasi-optimal parameters of the special GPIU method. Lastly, numerical experiments are carried out to examine the effectiveness of the special GPIU method with singular preconditioners by comparing its performance with that of other existing iterative methods for solving singular saddle point problems.**

***Index Terms*—singular saddle point problem, GPIU method, semi-convergence, singular splitting, Moore-Penrose inverse.**

I. Introduction

We consider the following large sparse augmented linear system

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

where $A \in \mathbb{R}^{m\times m}$ is a symmetric positive definite matrix, and $B \in \mathbb{R}^{m\times n}$ is a rank-deficient matrix with $m \geq n$. In this case, the coefficient matrix of (1) is singular and so the problem (1) is called a singular saddle point problem. This type of problem appears in many different scientific applications, such as constrained optimization problems [13], [18], [30], the finite element approximation for solving the Navier-Stokes equation [10], the constrained least squares problems and generalized least squares problems [1], [24], and so on.

In case of B being of full rank, many relaxation iterative methods have been proposed for solving the augmented linear system (1), e.g., SOR-like method [11], GSOR (Generalized SOR) method [2], PIU (Parameterized Inexact Uzawa) method [3], GPIU (Generalized Parameterized Inexact Uzawa) method [7], the modified SOR-like method [16], SSOR-like method [8], the modified SSOR-like method [17], Uzawa-SAOR method [21], GSSOR (Generalized SSOR) method [25], and MIAOR (Modified inexact AOR) method [22].

Recently, several authors have presented *semi-convergence analysis* of relaxation iterative methods for solving the singular saddle point problem (1). Zheng et al [28] studied semi-convergence of the PU (Parameterized Uzawa) method with nonsingular preconditioners, Li and Huang [14] examined semi-convergence of the GSSOR method with nonsingular preconditioners, Zhang et al. [26] provided semi-convergence analysis of the inexact Uzawa method with nonsingular preconditioners, Zhang and Wang [27] studied semi-convergence of the GPIU method with nonsingular preconditioners, Chao and Chen [6] provided semi-convergence analysis of the Uzawa-SOR method with nonsingular preconditioners, Zhou and Zhang [29] studied semi-convergence of the GMSSOR (Generalized Modified SSOR) method with nonsingular preconditioners, Yun [23] studied acceleration of one-parameter relaxation methods with nonsingular preconditioners, Liang and Zhang [15] presented semi-convergence analysis of the Uzawa-SAOR method with singular or nonsingular preconditioners, Yang et al. [20] presented semi-convergence analysis of the Uzawa-HSS method with singular preconditioners, Yang et al. [19] presented semi-convergence analysis of the PIU method with singular preconditioners, and so on.

The purpose of this paper is to provide performance analysis of a special case of the GPIU method with singular preconditioners for solving the singular saddle point problem (1). This paper is organized as follows. In Section 2, we provide preliminary results for semi-convergence analysis of the basic iterative methods. In Section 3, we provide semi-convergence results for a special case of the GPIU method with singular preconditioners. In Section 4, we first provide a methodology of how to choose nearly quasi-optimal parameters of the special GPIU method, and then we provide numerical experiments in order to examine the effectiveness of the special GPIU method with singular preconditioners. Lastly, some conclusions are drawn.

II. Preliminaries for Semi-convergence analysis

For simplicity of exposition, some notation and definitions are presented. For a vector x, x^* denotes the complex conjugate transpose of the vector x. $\lambda_{min}(H)$ and $\lambda_{max}(H)$ denote the minimum and maximum eigenvalues of the Hermitian matrix H, respectively. For a square matrix G, $R(G)$ denotes the range space of G, $N(G)$ denotes the null space of G, $\sigma(G)$ denotes the set of all eigenvalues of G, and $\rho(G)$ denotes the spectral radius of G.

Let us recall some useful results on iterative methods for solving singular linear systems based on matrix splitting. For a matrix $E \in \mathbb{R}^{n\times n}$, the smallest nonnegative integer k such that $rank(E^k) = rank(E^{k+1})$ is called the *index* of E, and denoted by $k = index(E)$. In other words, $index(E)$ is the size of the largest Jordan block corresponding to the zero eigenvalue of E. For a square matrx T, the *pseudo-spectral radius* $\nu(T)$ is defined by

$$\nu(T) = \max\{|\lambda| \mid \lambda \in \sigma(T) - \{1\}\}$$

where $\sigma(T)$ is the set of eigenvalues of T.

The *Moore-Penrose inverse* [4] of a singular matrix $E \in \mathbb{R}^{n\times n}$ is defined by the unique matrix $E^\dagger$ which satisfies the

Manuscript received April 7, 2017; revised June 15, 2017. This work was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2016R1D1A1A09917364).

Jae Heon Yun (Corresponding author) is with the Department of Mathematics, College of Natural Sciences, Chungbuk National University, Cheongju 28644, Korea e-mail: gmjae@cbnu.ac.kr.

following equations

$$E = EE^\dagger E,\ E^\dagger = E^\dagger EE^\dagger,\ (EE^\dagger)^T = EE^\dagger,\ (E^\dagger E)^T = E^\dagger E.$$

Let $A = M - N$ be a splitting of a singular matrix A, where M is singular. Then an iterative method corresponding to this singular splitting for solving a singular linear system $Ax = b$ is given by

$$x_{i+1} = (I - M^\dagger A)x_i + M^\dagger b \quad \text{for } i = 0, 1, \ldots. \tag{2}$$

Definition 2.1: The iterative method (2) is *semi-convergent* if for any initial guess x_0, the iteration sequence $\{x_i\}$ produced by (2) converges to a solution x_* of the singular linear system $Ax = b$.

Notice that a matrix T is called *semi-convergent* if $\lim_{k\to\infty} T^k$ exists, or equivalently $index(I - T) = 1$ and $\nu(T) < 1$ [4].

Theorem 2.2 ([5]): The iterative method (2) is semi-convergent if and only if $index(M^\dagger A) = 1$, $\nu(I - M^\dagger A) < 1$, and $N(M^\dagger A) = N(A)$, i.e., $I - M^\dagger A$ is semi-convergent and $N(M^\dagger A) = N(A)$.

III. Semi-convergence analysis of a special GPIU method

In this section, we provide semi-convergence analysis for a special case of the GPIU method with *singular* preconditioners for solving the singular saddle point problem (1). Notice that Chen and Jiang [7] presented convergence analysis of the GPIU method for nonsingular saddle point problems, and Zhang and Wang [27] provided semi-convergence analysis of the GPIU method with *nonsingular* preconditioners for the singular saddle point problem (1).

Assume that the coefficient matrix $\mathcal{A}$ of (1) is split as

$$\mathcal{A} = \begin{pmatrix} A & B \\ -B^T & 0 \end{pmatrix} = \mathcal{D} - \mathcal{L} - \mathcal{U}, \tag{3}$$

where

$$\mathcal{D} = \begin{pmatrix} P & 0 \\ 0 & Q \end{pmatrix},\ \mathcal{L} = \begin{pmatrix} 0 & 0 \\ B^T & 0 \end{pmatrix},\ \mathcal{U} = \begin{pmatrix} P - A & -B \\ 0 & Q \end{pmatrix}, \tag{4}$$

where $P \in \mathbb{R}^{m\times m}$ is a symmetric positive definite (SPD) matrix which approximates A, and $Q \in \mathbb{R}^{n\times n}$ is a singular symmetric positive semi-definite matrix which approximates the approximated Schur complement matrix $B^T P^{-1} B$. Let s be a real parameter and Q be chosen as $Q = B^T M^{-1} B$, where M is a SPD matrix which approximates P. Then a special case of the GPIU method with the singular preconditioning matrix Q, which is called the SGPIU method, is defined by

$$\begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = H_4(s) \begin{pmatrix} x_k \\ y_k \end{pmatrix} + M_4(s) \begin{pmatrix} f \\ -g \end{pmatrix}, \quad k = 0, 1, 2, \ldots, \tag{5}$$

where

$$H_4(s) = I - (\mathcal{D} + (s-1)\mathcal{L})^\dagger \mathcal{A}$$
$$M_4(s) = (\mathcal{D} + (s-1)\mathcal{L})^\dagger.$$

By some manipulation, one obtains

$$M_4(s) = \begin{pmatrix} P^{-1} & 0 \\ (1-s)Q^\dagger B^T P^{-1} & Q^\dagger \end{pmatrix} \tag{6}$$

and

$$H_4(s) = \begin{pmatrix} I_m - P^{-1}A & -P^{-1}B \\ Q^\dagger B^T (I_m + (s-1)P^{-1}A) & I_n + (s-1)Q^\dagger B^T P^{-1} B \end{pmatrix}. \tag{7}$$

From (5), (6) and (7), the SGPIU method with the singular preconditioner Q for solving the singular saddle point problem (1) can be rewritten as

Algorithm 1: SGPIU method with singular Q
Choose s and initial vectors x_0, y_0
For $k = 0, 1, \ldots,$ until convergence
$\quad x_{k+1} = x_k + P^{-1}(f - Ax_k - By_k)$
$\quad y_{k+1} = y_k + Q^\dagger(B^T x_{k+1} - g - sB^T(x_{k+1} - x_k))$
$\quad\quad = y_k + Q^\dagger(\, B^T((1-s)x_{k+1} + s\, x_k) - g\,)$
End For

If $s = 0$, P is replaced by $\frac{1}{\omega}P$ with $\omega \in (0, 2)$ and Q is replaced by $\frac{1}{\tau}Q$ with $\tau > 0$, then the SGPIU method reduces to the PIU method. In particular, if $s = 0$, $P = \frac{1}{\omega}A$ and Q is replaced by $\frac{1}{\tau}Q$, then the SGPIU method reduces to the PU method.

Assume that the rank of B is r, i.e., $r = rank(B) < n$. Let

$$B = W\Sigma V^* \qquad \text{and} \qquad \Sigma = \begin{pmatrix} \Sigma_r & 0 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{m\times n} \tag{8}$$

be the singular value decomposition of B, where W and V are unitary matrices, $\Sigma_r = diag(\sigma_1, \sigma_2, \ldots, \sigma_r)$ and σ_i's are positive singular values of B. Let W and V be partitioned into $W = (W_1, W_2)$ and $V = (V_1, V_2)$ with $W_1 \in \mathbb{C}^{m\times r}$, $W_2 \in \mathbb{C}^{m\times(m-r)}$, $V_1 \in \mathbb{C}^{n\times r}$, $V_2 \in \mathbb{C}^{n\times(n-r)}$, respectively. Let us define an $(m+n)\times(m+n)$ unitary matrix $\mathcal{P}$ as

$$\mathcal{P} = \begin{pmatrix} W & 0 \\ 0 & V \end{pmatrix}. \tag{9}$$

Let $\hat{H}_4(s) = \mathcal{P}^* H_4(s)\mathcal{P}$. If we define $\hat{P} = W^* P W$, $\hat{A} = W^* A W$, and $\hat{Q} = V^* Q V$. Since $Q = B^T M^{-1} B$ and $B = W\Sigma V^*$, one can obtain

$$\hat{Q} = V^* Q V = \begin{pmatrix} \hat{Q}_1 & 0 \\ 0 & 0 \end{pmatrix}, \tag{10}$$

where $\hat{Q}_1 = \Sigma_r W_1^* M^{-1} W_1 \Sigma_r$ is an $r\times r$ SPD matrix. Thus

$$\hat{Q}^\dagger = V^* Q^\dagger V = \begin{pmatrix} \hat{Q}_1^{-1} & 0 \\ 0 & 0 \end{pmatrix} \tag{11}$$

and

$$\hat{H}_4(s) = \begin{pmatrix} I_m - \hat{P}^{-1}\hat{A} & -\hat{P}^{-1}\Sigma \\ \hat{Q}^\dagger \Sigma^T (I_m + (s-1)\hat{P}^{-1}\hat{A} & I_n + (s-1)\hat{Q}^\dagger \Sigma^T \hat{P}^{-1}\Sigma \end{pmatrix}. \tag{12}$$

If we let $B_1 = \begin{pmatrix} \Sigma_r \\ 0 \end{pmatrix} \in \mathbb{R}^{m\times r}$, then using (10) to (12)

$$\hat{H}_4(s) = \begin{pmatrix} I_m - \hat{P}^{-1}\hat{A} & -\hat{P}^{-1}B_1 & 0 \\ \hat{Q}_1^{-1} B_1^T (I_m + (s-1)\hat{P}^{-1}\hat{A} & I_r + (s-1)\hat{Q}_1^{-1} B_1^T \hat{P}^{-1} B_1 & 0 \\ 0 & 0 & I_{n-r} \end{pmatrix} \tag{13}$$

and $\hat{Q}_1 = B_1^T (W^* M^{-1} W) B_1$. If we let

$$\bar{H}(s) = \begin{pmatrix} I_m - \hat{P}^{-1}\hat{A} & -\hat{P}^{-1}B_1 \\ \hat{Q}_1^{-1} B_1^T (I_m + (s-1)\hat{P}^{-1}\hat{A} & I_r + (s-1)\hat{Q}_1^{-1} B_1^T \hat{P}^{-1} B_1 \end{pmatrix}, \tag{14}$$

then $\bar{H}(s)$ is the iteration matrix of the SGPIU method applied to the following nonsingular saddle point problem

$$\begin{pmatrix} \hat{A} & B_1 \\ -B_1^T & 0 \end{pmatrix} \begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix} = \begin{pmatrix} \hat{f} \\ -\hat{g} \end{pmatrix} \tag{15}$$

with the preconditioning matrix $\hat{Q}_1$ and $\hat{P}$ as an approximation of $\hat{A}$.

Before proceeding to semi-convergence analysis of the SGPIU method for solving the singular saddle point problem (1), we first consider convergence of the SGPIU method for solving the nonsingular saddle point problem (15) whose iteration matrix is $\bar{H}(s)$. Note that B_1 has full column rank r. From the convergence analysis described in [7], [27], one can easily obtain the following three lemmas.

Lemma 3.1: Let λ be an eigenvalue of $\bar{H}(s)$ and $\begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix}$ be the corresponding eigenvector. Then $\lambda \neq 1$ and $\hat{x} \neq 0$.

Lemma 3.2: Let λ be an eigenvalue of $\bar{H}(s)$ and $\begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix}$ be the corresponding eigenvector. Then λ satisfies the following quadratic equation

$$\lambda^2 + \frac{\hat{\beta} - 2\hat{\alpha} + (1-s)\hat{\gamma}}{\hat{\alpha}}\lambda + \frac{\hat{\alpha} - \hat{\beta} + s\hat{\gamma}}{\hat{\alpha}} = 0,$$

where $\hat{\alpha} = \frac{\hat{x}^* \hat{P} \hat{x}}{\hat{x}^* \hat{x}}$, $\hat{\beta} = \frac{\hat{x}^* \hat{A} \hat{x}}{\hat{x}^* \hat{x}}$ and $\hat{\gamma} = \frac{\hat{x}^* B_1 \hat{Q}_1^{-1} B_1^T \hat{x}}{\hat{x}^* \hat{x}}$.

Lemma 3.3: Let λ be an eigenvalue of $\bar{H}(s)$ and $\begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix}$ be the corresponding eigenvector. Then $|\lambda| < 1$ if and only if

$$\hat{\gamma} - 4\hat{\alpha} + 2\hat{\beta} < 2s\hat{\gamma} < 2\hat{\beta} \quad \text{and} \quad \hat{\gamma} > 0,$$

where $\hat{\alpha} = \frac{\hat{x}^* \hat{P} \hat{x}}{\hat{x}^* \hat{x}}$, $\hat{\beta} = \frac{\hat{x}^* \hat{A} \hat{x}}{\hat{x}^* \hat{x}}$ and $\hat{\gamma} = \frac{\hat{x}^* B_1 \hat{Q}_1^{-1} B_1^T \hat{x}}{\hat{x}^* \hat{x}}$.

Theorem 3.4: Let $\lambda \neq 1$ be an eigenvalue of $H_4(s)$ and $\begin{pmatrix} x \\ y \end{pmatrix}$ be the corresponding eigenvector. Then $|\lambda| < 1$ if and only if

$$\gamma - 4\alpha + 2\beta < 2s\gamma < 2\beta \quad \text{and} \quad \gamma > 0,$$

where $\alpha = \frac{x^* P x}{x^* x}$, $\beta = \frac{x^* A x}{x^* x}$ and $\gamma = \frac{x^* B Q^\dagger B^T x}{x^* x}$.

Proof: Since $H_4(s)$ and $\hat{H}_4(s)$ are similar, λ is an eigenvalue of $\hat{H}_4(s)$. Since $\lambda \neq 1$, from (13) and (14) λ is also an eigenvalue of $\bar{H}(s)$. Let $\begin{pmatrix} \hat{x} \\ \hat{y} \end{pmatrix}$ be an eigenvector of $\hat{H}_4(s)$ corresponding to the λ. Then $\begin{pmatrix} \hat{x} \\ \hat{z} \end{pmatrix}$ is an eigenvector of $\bar{H}(s)$ corresponding to the λ, where $\hat{z}$ is the subvector consisting of the first r components of $\hat{y}$. From the relation $\hat{H}_4(s) = \mathcal{P}^* H_4(s) \mathcal{P}$, it is easy to show that $\hat{x} = W^* x$ and $\hat{y} = V^* y$. From Lemma 3.3, $|\lambda| < 1$ if and only if $\hat{\gamma} - 4\hat{\alpha} + 2\hat{\beta} < 2s\hat{\gamma} < 2\hat{\beta}$ and $\hat{\gamma} > 0$. Since $\hat{x} = W^* x$, $\hat{\alpha} = \alpha$ and $\hat{\beta} = \beta$ are immediately obtained. On the other hand,

$$\begin{aligned} x^* B Q^\dagger B^T x &= x^* B V \hat{Q}^\dagger V^* B^* x = x^* W \Sigma \hat{Q}^\dagger \Sigma^T W^* x \\ &= x^* W B_1 \hat{Q}_1^{-1} B_1^T W^* x = \hat{x}^* B_1 \hat{Q}_1^{-1} B_1^T \hat{x}. \end{aligned} \tag{16}$$

From (16), $\hat{\gamma} = \gamma$ is also obtained. Therefore, the proof is complete. ∎

Corollary 3.5: Let λ be an eigenvalue of $\bar{H}(s)$ and $\begin{pmatrix} x \\ y \end{pmatrix}$ be an eigenvector of $H_4(s)$ corresponding to the eigenvalue λ. Then $|\lambda| < 1$ if and only if

$$\gamma - 4\alpha + 2\beta < 2s\gamma < 2\beta \quad \text{and} \quad \gamma > 0, \tag{17}$$

where $\alpha = \frac{x^* P x}{x^* x}$, $\beta = \frac{x^* A x}{x^* x}$ and $\gamma = \frac{x^* B Q^\dagger B^T x}{x^* x}$.

Proof: Since $\lambda \neq 1$ from Lemma 3.1, this corollary follows from Theorem 3.4. ∎

The following theorem shows that the condition $\gamma > 0$ in (17) can be omitted.

Theorem 3.6: Let λ be an eigenvalue of $\bar{H}(s)$ and $\begin{pmatrix} x \\ y \end{pmatrix}$ be an eigenvector of $H_4(s)$ corresponding to the eigenvalue λ. Then $|\lambda| < 1$ if and only if

$$\gamma - 4\alpha + 2\beta < 2s\gamma < 2\beta, \tag{18}$$

where $\alpha = \frac{x^* P x}{x^* x}$, $\beta = \frac{x^* A x}{x^* x}$ and $\gamma = \frac{x^* B Q^\dagger B^T x}{x^* x}$.

Proof: From Theorem 3.4, it was shown that $\hat{\alpha} = \alpha > 0$, $\hat{\beta} = \beta > 0$ and $\hat{\gamma} = \gamma \geq 0$. If $\gamma = 0$ (i.e., $x \in N(B^T)$), then Lemma 3.2 implies that λ satisfies the following quadratic equation

$$\lambda^2 - (2 - \frac{\beta}{\alpha})\lambda + 1 - \frac{\beta}{\alpha} = 0.$$

Since $\lambda \neq 1$ from Lemma 3.1, $\lambda = 1 - \frac{\beta}{\alpha}$ is obtained. $|\lambda| = |1 - \frac{\beta}{\alpha}| < 1$ is equivalent to $\beta < 2\alpha$, which is exactly the same condition as the inequality (18) for $\gamma = 0$. Hence this theorem follows from Corollary 3.5. ∎

Theorem 3.7: Assume that $2P - A$ is symmetric positive definite. Then $\rho(\bar{H}(s)) < 1$ if the following inequality holds

$$\frac{1}{2} - \frac{\lambda_{min}(2P - A)}{\rho(BQ^\dagger B^T)} < s < \frac{\lambda_{min}(A)}{\rho(BQ^\dagger B^T)}.$$

Proof: Since $2P - A$ is symmetric positive definite, $2\alpha > \beta$ and thus the inequality (18) is true for all s when $\gamma = 0$. Suppose that $\gamma > 0$. Then the inequality (18) is equivalent to

$$\frac{1}{2} - \frac{2\alpha - \beta}{\gamma} < s < \frac{\beta}{\gamma}. \tag{19}$$

Hence, this corollary follows from Theorem 3.6 and the inequality (19). ∎

Now we provide semi-convergence result for the SGPIU method for solving the singular saddle point problem (1).

Theorem 3.8: Assume that $2P - A$ is symmetric positive definite. Let $Q = B^T M^{-1} B$ be a singular preconditioning matrix, where M is a SPD matrix which approximates P. Then the SGPIU method with the singular Q for solving the singular saddle point problem (1) is semi-convergent if the following inequality holds

$$\frac{1}{2} - \frac{\lambda_{min}(2P - A)}{\rho(BQ^\dagger B^T)} < s < \frac{\lambda_{min}(A)}{\rho(BQ^\dagger B^T)}.$$

Proof: By Theorem 3.7, $\rho(\bar{H}(s)) < 1$. From (13) and (14), it is clear that the matrix $\hat{H}_4(s)$ is semi-convergent. Since $\hat{H}_4(s) = \mathcal{P}^* H_4(s) \mathcal{P}$, $H_4(s)$ is also semi-convergent. Notice that $H_4(s) = I - (\mathcal{D} + (s-1)\mathcal{L})^\dagger \mathcal{A}$. From Theorem 2.2, we need to show that $N(\mathcal{A}) = N((\mathcal{D} + (s-1)\mathcal{L})^\dagger \mathcal{A})$. Hence, it is sufficient to show that $N((\mathcal{D} + (s-1)\mathcal{L})^\dagger \mathcal{A}) \subset N(\mathcal{A})$. Suppose that $\begin{pmatrix} x \\ y \end{pmatrix} \in N((\mathcal{D} + (s-1)\mathcal{L})^\dagger \mathcal{A})$. Then

$$P^{-1}(Ax + By) = 0 \quad \text{and} \quad -Q^\dagger B^T x + (1-s) Q^\dagger B^T P^{-1}(Ax + By) = 0.$$

From these equations, $Ax + By = 0$ and $-Q^\dagger B^T x = 0$. Since $QQ^\dagger B^T = B^T$, one obtains $\begin{pmatrix} x \\ y \end{pmatrix} \in N(\mathcal{A})$. Therefore, the proof is complete. ∎

Corollary 3.9: Let $\hat{P}$ be a SPD matrix which approximates A, $P = \frac{1}{\omega}\hat{P}$, $\hat{Q} = B^T M^{-1} B$ and $Q = \frac{1}{\tau}\hat{Q}$, where $0 < \omega < 2$, $\tau > 0$ and M is a SPD matrix which approximates P. Assume that $2P - A$ is symmetric positive definite. Then the SGPIU method with the singular

Q for solving the singular saddle point problem (1) is semi-convergent if the following inequality holds

$$\frac{1}{2} - \frac{\lambda_{min}(\frac{2}{\omega}\hat{P} - A)}{\tau\,\rho(B\hat{Q}^{\dagger}B^T)} < s < \frac{\lambda_{min}(A)}{\tau\,\rho(B\hat{Q}^{\dagger}B^T)}.$$

Corollary 3.10: Let $P = \frac{1}{\omega}A$, $\hat{Q} = B^T M^{-1} B$ and $Q = \frac{1}{\tau}\hat{Q}$, where $0 < \omega < 2$, $\tau > 0$ and M is a SPD matrix which approximates P. Then the SGPIU method with the singular Q for solving the singular saddle point problem (1) is semi-convergent if the following inequality holds

$$\frac{1}{2} - \left(\frac{2}{\omega} - 1\right)\frac{\lambda_{min}(A)}{\tau\,\rho(B\hat{Q}^{\dagger}B^T)} < s < \frac{\lambda_{min}(A)}{\tau\,\rho(B\hat{Q}^{\dagger}B^T)}.$$

Proof: Since $0 < \omega < 2$, $2\,P - A$ is symmetric positive definite. Hence this corollary follows from Corollary 3.9. ∎

IV. Numerical results

In this section, we first provide a methodology of how to choose nearly quasi-optimal parameters of the special GPIU method, and then we provide numerical experiments in order to examine the effectiveness of the SGPIU method with singular preconditioners for solving the singular saddle point problem (1). Performance of the SGPIU method with singular preconditioners is compared with that of the SGPIU method with nonsingular preconditioners and the PU or PIU methods with singular or nonsingular preconditioners.

In Tables II to V, *Iter* denotes the number of iteration steps, and *CPU* denotes the elapsed CPU time in seconds excluding the computational time of $Q^{\dagger}$ for the singular case of Q or the Cholesky factorization time of Q for the nonsingular case of Q. In all experiments, the right hand side vector $(f^T, -g^T)^T \in \mathbb{R}^{m+n}$ was chosen such that the exact solution of the saddle point problem (1) is $(x_*^T, y_*^T)^T = (1, 1, \ldots, 1)^T \in \mathbb{R}^{m+n}$, and the initial vector was set to the zero vector. All iterations for the singular saddle point problem are terminated if the current iteration satisfies RES $< 10^{-6}$, where *RES* is defined by

$$\text{RES} = \frac{\sqrt{\|f - Ax_k - By_k\|^2 + \|g - B^T x_k\|^2}}{\sqrt{\|f\|^2 + \|g\|^2}},$$

where $\|\cdot\|$ denotes the L_2-norm.

All numerical tests are carried out on a PC equipped with Intel Core i5-4570 3.2GHz CPU and 8GB RAM using MATLAB R2014b. For the elapsed CPU time, every experiment is repeated five times. The best and the worst ones out of 5 CPU times are discarded, and then the average of the remaining 3 CPU times is reported in Tables II to V.

Example 4.1 ([28]): We consider the saddle point problem (1), in which

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

$$B = \begin{pmatrix} \hat{B} & \tilde{B} \end{pmatrix} = \begin{pmatrix} \hat{B} & b_1 & b_2 \end{pmatrix} \in \mathbb{R}^{2p^2 \times (p^2+2)},$$

$$\hat{B} = \begin{pmatrix} I \otimes F \\ F \otimes I \end{pmatrix} \in \mathbb{R}^{2p^2 \times p^2}, \quad b_1 = \hat{B}\begin{pmatrix} e_{p^2/2} \\ 0 \end{pmatrix},$$

$$b_2 = \hat{B}\begin{pmatrix} 0 \\ e_{p^2/2} \end{pmatrix}, \quad e_{p^2/2} = (1, 1, \ldots, 1)^T \in \mathbb{R}^{p^2/2},$$

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

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

with $\otimes$ denoting the Kronecker product and $h = \frac{1}{p+1}$ the discretization mesh size. For this example, $m = 2p^2$ and $n = p^2 + 2$. Thus the total number of variables is $3p^2 + 2$. Numerical results for this example are listed in Tables II and III.

Example 4.2: Consider the Stokes equations of the following form: find $\mathbf{u}$ and v such that

$$\begin{cases} -\triangle \mathbf{u} + \nabla w = \mathbf{f} & \text{in } \Omega \\ -\nabla \cdot \mathbf{u} = 0 & \text{in } \Omega \end{cases}, \tag{20}$$

where $\Omega = (0,1) \times (0,1)$, $\mathbf{u}$ is a vector-valued function representing the velocity, and w is a scalar function representing the pressure. The boundary conditions are $\mathbf{u} = (0,0)^T$ on the three fixed walls ($x = 0,\ y = 0,\ x = 1$) and $\mathbf{u} = (1,0)^T$ on the moving wall ($y = 1$). Dividing Ω into a uniform grid with mesh size $h = \frac{1}{p}$ and discretizing (20) by using MAC (marker and cell) finite difference scheme [9], [12], the singular saddle point problem (1) is obtained, where $A \in \mathbb{R}^{2p(p-1) \times 2p(p-1)}$ is a symmetric positive definite matrix and $B = \begin{pmatrix} \hat{B} & \tilde{B} \end{pmatrix} \in \mathbb{R}^{2p(p-1) \times p^2}$ is a rank-deficient matrix of $rank(B) = p^2 - 1$ with $\hat{B} \in \mathbb{R}^{2p(p-1) \times (p^2-1)}$ and $\tilde{B} \in \mathbb{R}^{2p(p-1)}$. For this example, $m = 2p(p-1)$ and $n = p^2$. Thus the total number of variables is $3p^2 - 2p$. Numerical results for this example are listed in Tables IV and V.

For the SGPIU method, the symmetric positive definite matrices P are chosen as $P = \frac{1}{\omega}\hat{P}$ with a positive parameter $\omega \in (0,2)$ in three different ways. The first choice is $\hat{P} = A$, the second choice is $\hat{P} = (E-F)E^{-1}(E-F)^T$, where $A = E - F - F^T$ is a splitting of the symmetric positive definite matrix A with E a diagonal matrix and F a strictly lower triangular matrix, and the third choice is $\hat{P} = L_0 L_0^T$, where $A = L_0 L_0^T - R_0$ is a splitting of A obtained by an incomplete Cholesky factorization of A with no fill-in. The singular or nonsingular preconditioning matrices Q are chosen as $Q = \frac{1}{\tau}\hat{Q}$ with a positive parameter τ, where the matrices $\hat{Q}$ are chosen as in Table I. In Table I, Diag$(\hat{B}^T\hat{A}^{-1}\hat{B}, \tilde{B}^T\tilde{B})$ denotes a block diagonal matrix consisting of two submatrices $\hat{B}^T\hat{A}^{-1}\hat{B}$ and $\tilde{B}^T\tilde{B}$. The SGPIU algorithm for the nonsingular case of Q is the same as that for the singular case of Q except that Q^{-1} is used instead of $Q^{\dagger}$.

For these choices of P and Q, the SGPIU method with $s = 0$ reduces to the PU method for $P = \frac{1}{\omega}A$ or the PIU method for other two choices of P. For $s = 0$, the parameters ω and τ are chosen as the optimal or quasi-optimal parameters which are computed using the formulas given in [28] or [19], respectively (see data reported in the first line of Tables II - V for each case of $\hat{P}$). For $s \neq 0$, the parameters are chosen in two different ways: One choice is that ω and τ are chosen first as the optimal or quasi-optimal parameters and then s is chosen as the best one by tries (see data reported in the second line of Tables II - V for each case of $\hat{P}$), and the other choice is the experimentally chosen optimal parameters s, ω and τ (see data reported in the third line of Tables II - V for each case of $\hat{P}$). For singular matrix Q, $Q^{\dagger}$ is computed only once using the Matlab function *pinv* with a drop tolerance 10^{-13}, and then it is stored for later use. For nonsingular matrix Q, the Cholesky factorization of Q is computed only

TABLE I
CHOICES OF $\hat{Q}$ FOR SINGULAR OR NONSINGULAR PRECONDITIONING MATRICES $Q = \frac{1}{\tau}\hat{Q}$

Case Number	$\hat{Q}$	Description	Property of Q
I	$B^TM^{-1}B$	$M = \text{diag}(A)$	singular
II	$B^TM^{-1}B$	$M = \text{tridiag}(A)$	singular
III	$\text{Diag}(\hat{B}^T\hat{A}^{-1}\hat{B}, \tilde{B}^T\tilde{B})$	$\hat{A} = \text{diag}(A)$	nonsingular
IV	$\text{tridiag}\Big(\text{Diag}(\hat{B}^T\hat{A}^{-1}\hat{B}, \tilde{B}^T\tilde{B})\Big)$	$\hat{A} = \text{tridiag}(A)$	nonsingular

TABLE II
PERFORMANCE OF SGPIU METHOD WITH $P = \frac{1}{\omega}\hat{P}$ AND SINGULAR $Q = \frac{1}{\tau}\hat{Q}$ FOR EXAMPLE 4.1

			Case I of $\hat{Q}$					Case II of $\hat{Q}$				
m	n	$\hat{P}$	s	ω	τ	$Iter$	CPU	s	ω	τ	$Iter$	CPU
1152	578	A	0	0.2488	0.1423	131	0.138	0	0.3307	0.1985	90	0.096
			0.002	0.2488	0.1423	96	0.102	0.002	0.3307	0.1985	70	0.076
			-0.04	0.26	0.12	90	0.096	-0.02	0.33	0.19	68	0.074
		$(E-F)E^{-1}(E-F)^T$	0	1.7657	0.0626	219	0.040	0	1.8654	0.0588	232	0.041
			-0.35	1.7657	0.0626	108	0.024	-0.01	1.8654	0.0588	226	0.040
			-0.35	1.65	0.12	71	0.016	-0.30	1.65	0.12	108	0.022
		$L_0L_0^T$	0	1.3236	0.0910	176	0.031	0	1.4733	0.0811	174	0.032
			-0.25	1.3236	0.0910	81	0.017	-0.30	1.4733	0.0811	127	0.025
			-0.25	1.20	0.15	58	0.013	-0.30	1.35	0.15	81	0.017
2048	1026	A	0	0.1956	0.1084	174	0.365	0	0.2635	0.1519	120	0.264
			0.004	0.1956	0.1084	137	0.296	0.002	0.2635	0.1519	94	0.206
			-0.04	0.21	0.09	117	0.255	-0.05	0.25	0.14	93	0.204
		$(E-F)E^{-1}(E-F)^T$	0	1.8494	0.0377	332	0.224	0	1.9177	0.0347	360	0.250
			-0.35	1.8494	0.0377	159	0.113	-0.01	1.9177	0.0347	360	0.250
			-0.35	1.65	0.12	77	0.060	-0.35	1.65	0.12	117	0.087
		$L_0L_0^T$	0	1.4259	0.0568	250	0.140	0	1.5389	0.0489	259	0.145
			-0.25	1.4259	0.0568	114	0.068	-0.30	1.5389	0.0489	191	0.112
			-0.25	1.20	0.15	64	0.042	-0.30	1.35	0.15	87	0.055

TABLE III
PERFORMANCE OF SGPIU METHOD WITH $P = \frac{1}{\omega}\hat{P}$ AND NONSINGULAR $Q = \frac{1}{\tau}\hat{Q}$ FOR EXAMPLE 4.1

			Case III of $\hat{Q}$					Case IV of $\hat{Q}$				
m	n	$\hat{P}$	s	ω	τ	$Iter$	CPU	s	ω	τ	$Iter$	CPU
1152	578	A	0	0.2489	0.1423	131	0.138	0	0.5622	2.9447	44	0.047
			0.002	0.2489	0.1423	96	0.105	0.003	0.5622	2.9447	39	0.041
			-0.04	0.25	0.13	91	0.096	-0.01	0.52	3.10	38	0.040
		$(E-F)E^{-1}(E-F)^T$	0	1.7657	0.0626	219	0.050	0	0.9617	1.8293	238	0.039
			-0.35	1.7657	0.0626	108	0.028	0.01	0.9617	1.8293	238	0.039
			-0.35	1.65	0.12	71	0.021	0.35	1.30	0.95	160	0.028
		$L_0L_0^T$	0	1.3236	0.0910	176	0.041	0	0.7849	1.8970	177	0.028
			-0.25	1.3236	0.0910	81	0.022	0.01	0.7849	1.8970	177	0.028
			-0.25	1.20	0.15	58	0.018	0.40	1.0	1.1	119	0.021
2048	1026	A	0	0.1956	0.1084	174	0.324	0	0.5115	3.3270	52	0.099
			0.004	0.1956	0.1084	137	0.259	0.002	0.5115	3.3270	44	0.081
			-0.04	0.21	0.09	117	0.221	-0.02	0.49	3.30	41	0.075
		$(E-F)E^{-1}(E-F)^T$	0	1.8494	0.0377	332	0.128	0	0.9580	1.8482	318	0.075
			-0.35	1.8494	0.0377	159	0.065	0.01	0.9580	1.8482	318	0.075
			-0.35	1.65	0.12	77	0.036	0.30	1.35	0.98	207	0.053
		$L_0L_0^T$	0	1.4259	0.0568	250	0.097	0	0.7844	1.9042	236	0.056
			-0.25	1.4259	0.0568	114	0.048	0.01	0.7844	1.9042	236	0.056
			-0.25	1.20	0.15	64	0.030	0.40	1.0	1.1	163	0.043

once using the Matlab function *chol*, and then it is stored for later use.

For singular Q, $Q^\dagger b$ is computed using matrix-times-vector operation after constructing $Q^\dagger$ explicitly, which is very time-consuming. For nonsingular Q, $Q^{-1}b$ is computed using the forward and backward substitutions after constructing the Cholesky factorization of Q explicitly. As can be seen in Tables II to V, $P = \frac{1}{\omega}L_0L_0^T$ provides better performance and faster convergence rate than other two cases of P. From Tables II to V, it can be also seen that the SGPIU method with an appropriately chosen number s and optimal or quasi-optimal parameters ω and τ performs better than the PU or PIU methods with optimal or quasi-optimal parameters ω and τ (i.e., the SGPIU methods with $s = 0$). More specifically, when $P = \frac{1}{\omega}A$, SGPIU method with an appropriately chosen number s corresponding to optimal parameters ω and τ of the PU method performs significantly better than PU method with the optimal parameters ω and τ for all types of preconditioners Q used in this paper. When $P = \frac{1}{\omega}(E-F)E^{-1}(E-F)^T$ or $\frac{1}{\omega}L_0L_0^T$, SGPIU method with an appropriately chosen number s corresponding to quasi-optimal parameters ω and τ of the PIU method performs much better than PIU method with the quasi-optimal parameters ω and τ for the preconditioners Q of types I and III. Clearly, the SGPIU method with experimentally chosen optimal parameters s, ω and τ performs best. However, we do not have a formula for finding optimal parameters of the SGPIU method, which should be done in the future work.

TABLE IV
PERFORMANCE OF SGPIU METHOD WITH $P = \frac{1}{\omega}\hat{P}$ AND SINGULAR $Q = \frac{1}{\tau}\hat{Q}$ FOR EXAMPLE 4.2

			Case I of $\hat{Q}$					Case II of $\hat{Q}$				
m	n	$\hat{P}$	s	ω	τ	$Iter$	CPU	s	ω	τ	$Iter$	CPU
1104	576	A	0	0.2442	0.1392	132	0.136	0	0.3246	0.1939	89	0.096
			0.002	0.2442	0.1392	96	0.099	0.002	0.3246	0.1939	73	0.082
			-0.04	0.27	0.11	88	0.092	-0.04	0.32	0.18	69	0.075
		$(E-F)E^{-1}(E-F)^T$	0	1.7489	0.0672	239	0.042	0	1.8550	0.0634	255	0.044
			-0.35	1.7489	0.0672	99	0.023	-0.05	1.8550	0.0634	229	0.041
			-0.35	1.65	0.12	67	0.017	-0.35	1.65	0.12	110	0.024
		$L_0L_0^T$	0	1.3034	0.0971	197	0.034	0	1.4599	0.0871	196	0.037
			-0.25	1.3034	0.0971	75	0.018	-0.30	1.4599	0.0871	106	0.022
			-0.25	1.20	0.15	56	0.014	-0.30	1.30	0.15	76	0.018
1984	1024	A	0	0.1895	0.1047	177	0.382	0	0.2555	0.1466	119	0.260
			0.004	0.1895	0.1047	137	0.294	0.003	0.2555	0.1466	97	0.212
			-0.04	0.21	0.08	118	0.259	-0.05	0.24	0.14	93	0.205
		$(E-F)E^{-1}(E-F)^T$	0	1.8410	0.0399	368	0.212	0	1.9127	0.0369	399	0.234
			-0.40	1.8410	0.0399	137	0.083	-0.05	1.9127	0.0369	373	0.217
			-0.35	1.65	0.12	69	0.048	-0.35	1.65	0.12	116	0.072
		$L_0L_0^T$	0	1.4146	0.0599	301	0.180	0	1.5321	0.0517	293	0.172
			-0.30	1.4146	0.0599	102	0.066	-0.35	1.5321	0.0517	150	0.093
			-0.30	1.20	0.15	55	0.039	-0.30	1.35	0.15	78	0.052

TABLE V
PERFORMANCE OF SGPIU METHOD WITH $P = \frac{1}{\omega}\hat{P}$ AND NONSINGULAR $Q = \frac{1}{\tau}\hat{Q}$ FOR EXAMPLE 4.2

			Case III of $\hat{Q}$					Case IV of $\hat{Q}$				
m	n	$\hat{P}$	s	ω	τ	$Iter$	CPU	s	ω	τ	$Iter$	CPU
1104	576	A	0	0.2442	0.1392	132	0.136	0	0.0949	22.49	452	0.419
			0.002	0.2442	0.1392	96	0.099	0.001	0.0949	22.49	347	0.320
			-0.04	0.27	0.11	88	0.092	-0.01	0.09	23.00	276	0.260
		$(E-F)E^{-1}(E-F)^T$	0	1.7489	0.0672	239	0.054	0	0.3201	7.7888	792	0.107
			-0.35	1.7489	0.0672	99	0.026	0.01	0.3201	7.7888	792	0.107
			-0.35	1.65	0.12	67	0.020	0.20	1.89	0.083	263	0.041
		$L_0L_0^T$	0	1.3034	0.0971	197	0.044	0	1.6239	0.0391	462	0.064
			-0.25	1.3034	0.0971	75	0.021	0.05	1.6239	0.0391	441	0.063
			-0.25	1.20	0.15	56	0.017	0.06	1.57	0.10	202	0.033
1984	1024	A	0	0.1895	0.1047	177	0.333	0	0.0707	29.94	630	1.036
			0.004	0.1895	0.1047	137	0.257	0.001	0.0707	29.94	487	0.811
			-0.04	0.21	0.08	118	0.223	-0.01	0.07	29.00	411	0.682
		$(E-F)E^{-1}(E-F)^T$	0	1.8410	0.0399	368	0.137	0	0.3091	8.1193	1262	0.243
			-0.35	1.8410	0.0399	144	0.058	0.01	0.3091	8.1193	1262	0.243
			-0.35	1.65	0.12	69	0.031	0.10	1.89	0.083	425	0.089
		$L_0L_0^T$	0	1.4146	0.0599	301	0.115	0	1.6380	0.0221	982	0.197
			-0.30	1.4146	0.0599	102	0.044	0.09	1.6380	0.0221	771	0.159
			-0.30	1.20	0.15	55	0.027	0.02	1.57	0.10	321	0.070

V. CONCLUSION

In this paper, we provided semi-convergence analysis of the SGPIU method with singular preconditioners for solving singular saddle point problems. Numerical experiments show that the SGPIU method with an appropriately chosen number s and optimal or quasi-optimal parameters ω and τ performs better than the PU or PIU methods with optimal or quasi-optimal parameters ω and τ. More specifically, when $P = \frac{1}{\omega}A$, SGPIU method with an appropriately chosen number s corresponding to optimal parameters ω and τ of the PU method performs significantly better than the PU method for all types of preconditioners Q used in this paper. When $P = \frac{1}{\omega}(E-F)E^{-1}(E-F)^T$ or $\frac{1}{\omega}L_0L_0^T$, SGPIU method with an appropriately chosen number s corresponding to quasi-optimal parameters ω and τ of the PIU method performs about twice faster than the PIU method for the preconditioners Q of types I and III. It means that the methodology of choosing an appropriate value of s corresponding to the optimal or quasi-optimal parameters ω and τ of the PU or PIU methods works quite well for the SGPIU method.

It is clear that the SGPIU method with experimentally chosen optimal parameters s, ω and τ performs best. So, further research for finding optimal parameters of the SGPIU method will be done in the future work. The SGPIU method with singular preconditioners performs rather well as compared with that with nonsingular preconditioners. However, the SGPIU method with singular preconditioners Q requires the computation of $Q^{\dagger}b$ for a given vector b, which is very time-consuming. Future work will also include how to compute $Q^{\dagger}b$ efficiently for a given vector b.

ACKNOWLEDGMENT

The author is grateful to the anonymous reviewers for their valuable comments and suggestions which improved the quality and the clarity of the paper.

REFERENCES

[1] M. Arioli, I. S. Duff, P. P. M. de Rijk, "On the augmented system approach to sparse least squares problems," *Numer. Math.*, vol. 55, pp. 667-684, 1989.

[2] Z.-Z. Bai, B. N. Parlett, Z.-Q. Wang, "On generalized successive overrelaxation methods for augmented linear systems," *Numer. Math.*, vol. 102, pp. 1-38, 2005.

[3] Z.-Z. Bai, Z.-Q. Wang, "On parameterized inexact Uzawa methods for generalized saddle point problems," *Linear Algebra Appl.*, vol. 428, pp. 2900-2932, 2008.
[4] A. Berman and R.J. Plemmons, *Nonnegative matrices in the Mathematical sciences*, New York, Academic Press, 1979.
[5] Z.-H. Cao, "On the convergence of general stationary linear iterative methods for singular linear systems," *SIAM J. Matyrix Anal. Appl.*, vol. 29, pp. 1382-1388, 2008.
[6] Z. Chao, G. Chen, "Semi-convergence analysis of the Uzawa-SOR methods for singular saddle point problems," *Appled Mathematics Letters*, vol. 35, pp. 52-57, 2014.
[7] F. Chen, Y.-L. Jiang, "A generalization of the inexact parameterized Uzawa methods for saddle point problems," *Appl. Math. Comput.*, vol. 206, pp. 765-771, 2008.
[8] M. T. Darvishi, P. Hessari, "Symmetric SOR method for augmented systems," *Appl. Math. Comput.*, vol. 183, pp. 409-415, 2006.
[9] H. C. Elman, "Preconditioning for the steady-state Navier-Stokes equations with low viscosity," *SIAM J. Sci. Comput.*, vol. 20, pp. 1299-1316, 1999.
[10] H. C. Elman, D. J. Silvester, "Fast nonsymmetric iteration and preconditioning for Navier-Stokes equations," *SIAM J. Sci. Comput.*, vol. 17, pp. 33-46, 1996.
[11] G. H. Golub, X. Wu, J.-Y. Yuan, "SOR-like methods for augmented systems," *BIT Numerical Mathematics*, vol. 41, pp. 71-85, 2001.
[12] F. H. Harlow and J. E. Welch, "Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface," *Phys. Fluids*, vol. 8, pp. 2182-2189. 1965.
[13] M. Keyanpour and A. Mahmoudi, "A Hybrid Method for Solving Optimal Control Problems," IAENG International Journal of Applied Mathematics, vol. 42, no. 2, pp80-86, 2012.
[14] J.-I. Li and T.-Z. Huang, "The semi-convergence of generalized SSOR method for singular augmented systems," *Lecture Notes in Computer Science: High Performance Computing and Applications*, vol. 5938, pp. 230–235, 2010.
[15] Z.-Z. Liang, G.-F. Zhang, "On semi-convergence of a class of Uzawa methods for singular saddle point problems," *Appl. Math. Comput.*, vol. 247, pp. 397-409, 2014.
[16] X. Shao, Z. Li, C. Li, "Modified SOR-like method for the augmented system," *Intern. J. Comput. Math.*, vol. 84, pp. 1653–1662, 2007.
[17] S.-L. Wu, T.-Z. Huang, X.-L. Zhao, "A modified SSOR iterative method for augmented systems," *J. Comput. Appl. Math.*, vol 228, pp. 424-433, 2009.
[18] S. Wright, "Stability of augmented system factorization in interior point methods," *SIAM J. Matrix Anal. Appl.*, vol. 18, pp. 191-222, 1997.
[19] A.-L. Yang, Y. Dou, Y.-J. Wu, X. Li, "On generalized parameterized inexact Uzawa methods for singular saddle point problems," *Numer. Algor.*, vol. 69, pp. 579-593, 2015.
[20] A.-L. Yang, X. Li, Y.-J. Wu, "On semi-convergence of the Uzawa-HSS method for singular saddle point problems," *Appl. Math. Comput.*, vol. 252, pp. 88-98, 2015.
[21] J. H. Yun, "Variants of the Uzawa method for saddle point problem," *Comput. Math. Appl.*, vol. 65, pp. 1037-1046, 2013.
[22] J. H. Yun, "Convergence of relaxation iterative methods for saddle point problem," *Appl. Math. Comput.*, vol. 251, pp. 65-80, 2015.
[23] J. H. Yun, "Acceleration of one-parameter relaxation methods for singular saddle point problems," *J. Korean Math. Soc.*, vol. 53, pp. 691-707, 2016.
[24] J.-Y. Yuan, A. N. Iusem, "Preconditioned conjugate gradient methods for generalized least squares problem," *J. Comput. Appl. Math.*, vol. 71, pp. 287-297, 1996.
[25] G.-F. Zhang, Q.-H. Lu, "On generalized symmetric SOR method for augmented systems," *J. Comput. Appl. Math.*, vol. 219, pp. 51-58, 2008.
[26] N.-M. Zhang, T.-T. Lu, Y. Wei, "Semi-convergence analysis of Uzawa methods for singular saddle point problems," *J. Comput. Appl. Math.*, vol. 255, pp. 334-345, 2014.
[27] G.-F. Zhang, S.-S. Wang, "A generalization of parameterized inexact Uzawa method for singular saddle point problems," *Appl. Math. Comput.*, vol. 219, pp. 4225-4231, 2013.
[28] B. Zheng, Z.-Z. Bai, X. Yang, "On semi-convergence of parameterized Uzawa methods for singular saddle point problems," *Linear Algebra Appl.*, vol. 431, pp. 808-817, 2009.
[29] L. Zhou, N.-M. Zhang, "Semi-convergence analysis of GMSSOR methods for singular saddle point problems," *Comput. Math. Appl.*, vol. 68, pp. 596-605, 2014.
[30] M.-Z. Zhu, Y.-E. Qi, "On the Eigenvalues Distribution of Preconditioned Block Two-by-two Matrix," IAENG International Journal of Applied Mathematics, vol. 46, no. 4, pp500-504, 2016.