Journal of Computational and Applied Mathematics 311 (2017) 293–305

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

Convergence behavior of generalized parameterized Uzawa method for singular saddle-point problems[☆]

Zhao-Zheng Liang, Guo-Feng Zhang[*]

School of Mathematics and Statistics, Lanzhou University, Lanzhou 730000, PR China

ARTICLE INFO

Article history:
Received 16 January 2015
Received in revised form 9 March 2016

MSC:
65F10
65F50

Keywords:
Singular saddle-point problem
Uzawa method
Convergence
Proper splitting
Moore–Penrose inverse

ABSTRACT

In this paper, we will seek the least squares solution for singular saddle-point problems. The parameterized Uzawa (PU) method is further studied and a generalized PU (GPU) proper splitting is proposed. The convergence behavior of the corresponding GPU iteration is studied. It is proved that the GPU iteration method can converge to the best least squares solutions of the singular saddle-point problems. In addition, we prove that the GPU preconditioned GMRES for singular saddle-point problems will also determine the least squares solution at breakdown. The eigenvalue distributions of the GPU preconditioned matrix are derived. Numerical experiments are presented, which show that the convergence behavior of the singular preconditioning is significantly better than that of the corresponding nonsingular case and demonstrate that the GPU iteration has better convergence behavior than the PU iteration, both as a solver and a preconditioner of GMRES.

1. Introduction

Consider the singular saddle-point problem:

$$\mathscr{A}X \equiv \begin{pmatrix} A & B \\ -B^T & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ -g \end{pmatrix} \equiv c, \tag{1.1}$$

where $A \in \mathbb{R}^{n\times n}$ is a positive real matrix, i.e., the symmetric part of A is positive definite, $B \in \mathbb{R}^{n\times m}$ is a rank deficient matrix with $m < n, f \in \mathbb{R}^n$ and $g \in \mathbb{R}^m$ are two given vectors. Many different scientific and engineering applications can derive linear systems of the structure (1.1), for example the mixed finite element approximation of elliptic PDEs, the computational fluid dynamics, the electronic networks, the image reconstruction and registration, and so forth. See [1–5] and the references therein.

For nonsingular saddle-point problems of the form (1.1), a number of iterative methods have been proposed, such as the Uzawa-type methods [6–11], the HSS-like methods [12–17], the null space methods [18,19], the Krylov subspace methods and the preconditioned variants [20–26], and so on.

If B of (1.1) is rank deficient, then $\mathscr{A}$ of (1.1) is singular. For example, in the finite difference discretization of the Navier–Stokes equation with periodic boundary conditions, B in (1.1) becomes singular. Some authors have studied many iterative methods [27–38] for solving singular saddle-point problems. If an iteration method for solving (1.1) is convergent, it is critical to seek which solution of (1.1) it will converge to, as it may have countless solutions. However, as far as we know

[☆] This work was supported by the National Natural Science Foundation of China (Nos. 11271174, 11511130015).

[*] Corresponding author.
E-mail addresses: liangzz11@lzu.edu.cn (Z.-Z. Liang), gf_zhang@lzu.edu.cn (G.-F. Zhang).

http://dx.doi.org/10.1016/j.cam.2016.07.022
0377-0427/© 2016 Elsevier B.V. All rights reserved.

that the iterative methods for solving singular linear systems in the literature mainly showed concern for the convergence or semi-convergence, instead of which solution the iterative sequences will converge to.

In this paper, we will seek the least squares solution (generalized inverse solution) $\mathscr{A}^\dagger c$ for the singular saddle-point problem (1.1), where $\mathscr{A}^\dagger$ is the Moore–Penrose inverse of $\mathscr{A}$, which satisfies the following Moore–Penrose equations [39,40]:

$$\mathscr{A}\mathscr{A}^\dagger\mathscr{A} = \mathscr{A}, \qquad (\mathscr{A}\mathscr{A}^\dagger)^T = \mathscr{A}\mathscr{A}^\dagger, \qquad (\mathscr{A}^\dagger\mathscr{A})^T = \mathscr{A}^\dagger\mathscr{A}, \qquad \mathscr{A}^\dagger\mathscr{A}\mathscr{A}^\dagger = \mathscr{A}^\dagger. \tag{1.2}$$

In [10], Bai et al. presented the following PU method for solving nonsingular saddle-point problems.

The PU iteration method: Let $A \in \mathbb{R}^{n\times n}$ and $Q \in \mathbb{R}^{m\times m}$ be symmetric positive definite matrices. Given initial vectors $x^{(0)} \in \mathbb{R}^n$, $y^{(0)} \in \mathbb{R}^m$, and two relaxation factors ω, τ with $\omega, \tau \neq 0$, for $k = 0, 1, 2, \ldots$ until the iteration sequence $\{(x^{(k)})^T, (y^{(k)})^T\}$ converges to the exact solution of the nonsingular saddle-point problem of structure (1.1), compute

$$\begin{cases} x^{(k+1)} = x^{(k)} + \omega A^{-1}(f - Ax^{(k)} - By^{(k)}), \\ y^{(k+1)} = y^{(k)} + \tau Q^{-1}(B^T x^{(k+1)} - g). \end{cases} \tag{1.3}$$

Here, Q is an approximate matrix of the Schur complement matrix $B^TA^{-1}B$.

Basic convergence theorems of the PU method for nonsingular saddle-point problems have been established in [10]. The choice of the optimal parameters and the corresponding quasi-optimal convergence factor for the PU method were discussed. In [30], Zheng et al. further proved the semi-convergence of the PU method for singular saddle-point problems. The optimal iteration parameter and the corresponding optimal semi-convergence factor were also derived. However, the results in [30] did not involve the convergence behavior of the PU iteration (1.3), i.e., which solution the iteration converges to is unknown.

In this paper, we will further analyze the PU iteration method and generalize it to a generalized PU (GPU) iteration method for solving singular saddle-point problems. The focus of our work is the convergence behavior of the GPU iteration, i.e., we consider which solution of (1.1) the GPU iteration method converges to. We prove that the GPU splitting can result in a proper splitting and the GPU iteration method converges to the best least squares solution $\mathscr{A}^\dagger c$ of (1.1). Moreover, when (1.1) is consistent, GMRES with the GPU preconditioner can determine $\mathscr{A}^\dagger c$ at breakdown. Eigenvalue distributions of the GPU preconditioned saddle-point matrix are given, which are instructive for the implementations of Krylov subspace methods.

The outline of the paper is organized as follows. In Section 2, the GPU iteration method for singular saddle-point problems is described. In Section 3, we give the convergence analysis of the GPU method. In Section 4, Krylov subspace acceleration with the GPU preconditioner as well as the eigenvalue properties of the GPU preconditioned matrix are presented. In Section 5, numerical examples are given to show that the GPU iteration is effective for solving singular saddle-point problems, both as a solver and a preconditioner of the GMRES method. Some conclusions are drawn in the last section.

Throughout this paper, we denote x^* as the conjugate transpose of a vector x. I is the identity matrix of proper size. The minimum and the maximum eigenvalues of a symmetric matrix H are denoted by $\lambda_{\min}(H)$ and $\lambda_{\max}(H)$, respectively. For a matrix A, rank(A) denotes its rank, $\rho(A)$ denotes its spectral radius. $\mathscr{R}(A)$ and $\mathscr{N}(A)$ denote the range and the null spaces of the matrix A, respectively.

2. Generalized parameterized Uzawa method for singular saddle-point problems

For the coefficient matrix $\mathscr{A}$ of the singular saddle-point problem (1.1), we make the splitting $\mathscr{A} = \mathscr{D} - \mathscr{L} - \mathscr{U}$, where

$$\mathscr{D} := \begin{pmatrix} A & 0 \\ 0 & Q \end{pmatrix}, \qquad \mathscr{L} := \begin{pmatrix} 0 & 0 \\ B^T & 0 \end{pmatrix}, \qquad \mathscr{U} := \begin{pmatrix} 0 & -B \\ 0 & Q \end{pmatrix}$$

and $Q \in \mathbb{R}^{m\times m}$ is an approximation of the matrix $B^TA^{-1}B$.

Define

$$\Omega := \begin{pmatrix} \omega I_m & 0 \\ 0 & \tau I_n \end{pmatrix}$$

with ω and τ being nonzero real parameters. Then we can give the GPU splitting as $\mathscr{A} = \mathscr{M}_{\omega,\tau} - \mathscr{N}_{\omega,\tau}$, with

$$\mathscr{M}_{\omega,\tau} = \Omega^{-1}(\mathscr{D} - \Omega\mathscr{L}) = \begin{pmatrix} \frac{1}{\omega}A & 0 \\ -B^T & \frac{1}{\tau}Q \end{pmatrix} \quad \text{and} \quad \mathscr{N}_{\omega,\tau} = \Omega^{-1}[(I_{m+n} - \Omega)\mathscr{D} + \Omega\mathscr{U}] = \begin{pmatrix} \left(\frac{1}{\omega} - 1\right)A & -B \\ 0 & \frac{1}{\tau}Q \end{pmatrix}. \tag{2.1}$$

Since Q is an approximation of $B^TA^{-1}B$, we choose $Q = B^TP^{-1}B$ with $P \in \mathbb{R}^{n\times n}$ being a positive real matrix. Thus Q is singular and symmetric positive semi-definite. Under these situations, $\mathscr{M}_{\omega,\tau}$ is also singular. Then the GPU iteration scheme can be described as follows:

$$X^{(k+1)} = \mathscr{M}_{\omega,\tau}^\dagger \mathscr{N}_{\omega,\tau} X^{(k)} + \mathscr{M}_{\omega,\tau}^\dagger c. \tag{2.2}$$