

## THE SEMI-CONVERGENCE OF GENERALIZED SSOR METHOD FOR SINGULAR AUGMENTED SYSTEMS

JIAN-LEI LI, TING-ZHU HUANG, AND DANG LUO

**Abstract.** Recently, Zhang and Lu proposed the generalized symmetric SOR (GSSOR) method for solving the nonsingular augmented systems and studied the convergence of the GSSOR method. In this paper, we prove the semi-convergence of the GSSOR method when it is applied to solve the singular augmented systems, which is the generalization of the GSSOR iteration method.

**Key words.** semi-convergence, singular augmented systems, generalized symmetric SOR method, iterative method.

### 1. Introduction

We consider the following augmented systems of the form:

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

where  $A \in \mathbb{R}^{m \times m}$  is a symmetric positive definite matrix,  $B \in \mathbb{R}^{m \times n}$  is a matrix of rank  $r$  with  $n \leq m$ , i.e.,  $\text{rank } B = r$ ,  $0 < r \leq n$ .  $f \in \mathbb{R}^m$  and  $g \in \mathbb{R}^n$  are two given vectors. Denote  $B^T$  as the transpose of the matrix  $B$ . When  $r = n$ , note that the coefficient matrix is nonsingular and the linear systems (1) have a unique solution. When  $r < n$ , the coefficient matrix is singular, in such case, we assume that the linear systems (1) are consistent. Such systems are also referred to as saddle point problems or Karush-Kuhn-Tucker (KKT) systems. The augmented systems (1) are important and arise in a large number of scientific and engineering applications, such as the field of computational fluid dynamics [2], constrained and weighted least squares problems [3], interior point methods in constrained optimization [4], mixed finite element approximations of elliptic partial differential equations (PDEs) [5]. Especially, see [1] for a comprehensive survey.

For the nonsingular augmented systems (1), many efficient iterative methods based on matrix splitting as well as their numerical properties have been studied in the literature. The preconditioned iterative methods [8, 9], the inexact Uzawa methods [6, 10], the SOR-like method [7], the general SOR (GSOR) method [13], the symmetric SOR (SSOR) method [15, 16] and the modified SSOR (MSSOR) method [17] for solving the nonsingular augmented systems (1) were proposed and analyzed, respectively. Furthermore, the general symmetric SOR (GSSOR) method [18] was presented. In most cases, the matrix  $B$  is full column rank in scientific computing and engineering applications, but not always. If  $r < n$ , the augmented systems become the singular linear systems. When the linear systems are consistent, Zheng, Bai and Yang [12] show that the GSOR method proposed in [13] can be used to solve the singular augmented systems (1), and it is semi-convergent.

In this paper, the GSSOR method for solving singular linear augmented systems (1) is further investigated and the semi-convergence conditions are proposed, which

---

Received by the editors December 1, 2010 and, in revised form, March 18, 2011.  
2000 *Mathematics Subject Classification.* 65F10; 65F50.  
This research was supported by NSFC (60973015).

generalize the result of Zhang and Lu [18] for the nonsingular augmented systems to the singular augmented systems.

**2. The semi-convergence of the GSSOR method**

Before the semi-convergence of the GSSOR method is given, we first give some basic concepts and lemmas for latter use.

For a matrix  $A \in \mathbb{R}^{m \times m}$ , the splitting  $A = M - N$  is a nonsingular splitting if  $M$  is nonsingular. Denote  $\sigma(A)$  and  $\rho(A)$  as the spectrum and spectral radius of a square matrix  $A$ , respectively.  $I$  is the identity matrix with appropriate dimension. Let  $T = M^{-1}N$ , then solving linear systems  $Ax = c$  is equivalent to considering the following iterative scheme

$$(2) \quad x_{i+1} = Tx_i + M^{-1}c, \quad k = 0, 1, 2, \dots$$

It is well known that for nonsingular systems the iterative scheme (2) is convergent if and only if  $\rho(T) < 1$ . But for the singular systems, we have  $1 \in \sigma(T)$  and  $\rho(T) \geq 1$ , so that one can require only the semi-convergence of the iterative method (2). By [14], the iterative scheme (2) is semi-convergent if and only if the following three conditions are satisfied:

- (1)  $\rho(T) = 1$ ;
- (2) Elementary divisors associated with  $\lambda = 1 \in \sigma(T)$  are linear, i.e.,  $rank(I - T)^2 = rank(I - T)$ ;
- (3) If  $\lambda \in \sigma(T)$  with  $|\lambda| = 1$ , then  $\lambda = 1$ , i.e.,

$$\vartheta(T) = \max\{|\lambda|, \lambda \in \sigma(T), \lambda \neq 1\} < 1.$$

In this situation, the associated convergence factor is  $\vartheta(T)$ . We call a matrix  $T$  is semi-convergent provided it satisfies the above three conditions, and iterative method (2) is semi-convergent if  $T$  is a semi-convergent matrix. On the semi-convergence of the iterative method for solving general singular linear systems  $Ax = b$ , for more details, one can see [11, 14, 19, 20, 21, 22]. When  $A$  is singular, the following two lemmas give the semi-convergence property about the iteration method (2).

**Lemma 2.1** [14, 22] *Let  $A = M - N$  with  $M$  nonsingular,  $T = M^{-1}N$ . Then for any initial vector  $x_0$ , the iterative scheme (2) is semi-convergent to a solution  $x$  of linear equations  $Ax = c$  if and only if the matrix  $T$  is semi-convergent.*

**Lemma 2.2** [11, 12] *Let  $H \in \mathbb{R}^{l \times l}$  with positive integers  $l$ . Then the partitioned matrix*

$$T = \begin{pmatrix} H & 0 \\ L & I \end{pmatrix}$$

*is semi-convergent if and only if either of the following conditions holds true:*

- (1)  $L = 0$  and  $H$  is semi-convergent;
- (2)  $\rho(H) < 1$ .

Before the semi-convergence of the GSSOR method is discussed, now, we review the GSSOR method presented in [18]. In fact, the augmented systems (1) can be written as the following equivalent form

$$(3) \quad \begin{pmatrix} A & B \\ -B^T & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ -g \end{pmatrix}.$$