JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

ELSEVIER Journal of Computational and Applied Mathematics 71 (1996) 287–297

Preconditioned conjugate gradient method for generalized least squares problems

J.Y. Yuan [a,*,1], A.N. Iusem [b,2]

[a] *Departamento de Matemática, Universidade Federal do Paraná, Centro Politécnico, CP: 19.081, Curitiba, CEP: 81531-990, Brazil*

[b] *Instituto de Matemática Pura e Aplicada, Estrada Dona Castorina 110, Jardim Botânico, Rio de Janeiro, CEP: 22460-320, Brazil*

Received 10 March 1995; revised 27 September 1995

Abstract

A variant of the preconditioned conjugate gradient method to solve generalized least squares problems is presented. If the problem is

$$\min (Ax - b)^{\mathrm{T}} W^{-1} (Ax - b)$$

with $A \in \mathbb{R}^{m \times n}$ and $W \in \mathbb{R}^{m \times m}$ symmetric and positive definite, the method needs only a preconditioner $A_1 \in \mathbb{R}^{n \times n}$, but not the inverse of matrix W or of any of its submatrices. Freund's comparison result for regular least squares problems is extended to generalized least squares problems. An error bound is also given.

Keywords: Preconditioned conjugate gradient method; Least squares; Generalized least squares problems

AMS classifications: 65F20

1. Introduction

Paige [7, 8] transformed the generalized least squares problem

$$\min_{x \in \mathbb{R}^n} (Ax - b)^{\mathrm{T}} W^{+} (Ax - b), \tag{1.1}$$

* Corresponding author. E-mail: yuan@inf.ufpr.br., jin@mat.ufpr.br.

[1] The work of this author was supported by CNPq grant 301035/93-8.

[2] The work of this author was supported by CNPq grant 301280/86.

SSDI 0377-0427(95)00239-1

where $A \in \mathbb{R}^{m\times n}$, $W \in \mathbb{R}^{m\times m}$ is symmetric and positive semi-definite, and $m \geqslant n$, into an equality-constrained least squares problem

$$\min_{x\in\mathbb{R}^n,\, v\in\mathbb{R}^k} \quad \|v\|_2, \tag{1.2}$$

$$\text{s.t.} \quad Ax + Bv = b, \tag{1.3}$$

where $W = BB^{\mathrm{T}}$, $B \in \mathbb{R}^{m\times k}$ and $k = \operatorname{rank}(W)$. Then he presented direct method to solve problem (1.2) and (1.3). He also provided an error analysis for his method. However, Paige's method is a direct method which is not appropriate to handle large sparse problems.

In terms of (1.2) and (1.3), problem (1.1) can be solved by implicit null space iterative methods given by James in [5, 6], or by the BNP algorithm [2]. But, as in Paige's method, the BNP algorithm needs to decompose the weighted matrix W, and find the inverse of B or of a submatrix of (A, B). Also the BNP algorithm does not work when W is semi-definite because (A, B) is not full rank. Even when W is positive definite, the BNP algorithm needs to invert an $m \times m$ matrix to be used as a preconditioner. In order to overcome these limitations, Yuan [9] recently proposed the 2-cyclic SOR method for problem (1.1), which we describe next.

Algorithm 1.1 (2-cyclic SOR Algorithm).

(1) Factorize A_1 and W_{22}, set initial vectors $x^{(0)} = 0$, $r^{(0)} = 0$;

(2) Select a relaxation parameter ω;

(3) Iterate for $k = 0, 1, \dots,$ until “convergence”

$$x^{(k+1)} = (1-\omega)x^{(k)} + \omega A_1^{-\mathrm{T}}[b_1 - (W_{12} - W_{11}P^{\mathrm{T}})r_2^{(k)}],$$

$$r_2^{(k+1)} = (1-\omega)r_2^{(k)} + \omega W_{22}^{-1}(b_2 - W_{12}^{\mathrm{T}}r_1^{(k)} - A_2x^{(k+1)}),$$

$$r_1^{(k+1)} = (1-\omega)r_1^{(k)} - \omega P^{\mathrm{T}}r_2^{(k+1)},$$

where

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \qquad W = \begin{pmatrix} W_{11} & W_{12} \\ W_{12}^{\mathrm{T}} & W_{22} \end{pmatrix},$$

and

$$P = A_2A_1^{-1}.$$

This method is always convergent (see [9]), but it requires estimation of the relaxation parameter ω.

It is well known that the conjugate gradient method does not need any iterative parameter and enjoys a finite termination property. In this paper, the conjugate gradient method is extended to solve generalized least squares problems by preconditioner techniques. The preconditioned conjugate gradient method for problem (1.1) also has the finite termination property in the absence of roundoff errors. For the sake of simplicity, we consider only the case of a symmetric and positive definite weight matrix W and a full rank matrix A. In fact, our method can also work in the case of semi-definite W with $\operatorname{rank}(W) \geqslant m - n$. The numerical experiments show that the method can also work for some cases in which $\operatorname{rank}(W) \leqslant m - n$. Freund showed comparison result for the conjugate gradient method and the SOR method for regular least squares problems in [4]. We establish here a similar comparison result between the preconditioned conjugate gradient method and the 2-cyclic SOR method for problem (1.1), so extending Freund's result [4] to generalized least squares problems. Our analysis follows the analysis in [2].

In Section 2, a symmetric and positive definite subsystem is derived from the normal equation of problem (1.1) by preconditioner techniques. Then the conjugate gradient method is applied to solve the subsystem. The error bound of the method for problem (1.1) is given in Section 3. Some theoretical comparison results among SOR-type methods and preconditioned conjugate gradient method for problem (1.1) are obtained in Section 4. It follows from the comparison results that the conjugate gradient method is better than SOR-type methods for problem (1.1) in the same Krylov subspace. Finally, some remarks on theoretical results and numerical experiments are given in Section 5.

We define the elliptic norm $\|\cdot\|_D$ as $\|x\|_D = \sqrt{x^T Dx}$, where D is symmetric and positive definite.

2. Preconditioned conjugate gradient algorithm

Suppose that A is split in the following way:

$$A = \begin{pmatrix} A_1 \\ A_2 \end{pmatrix}, \tag{2.1}$$

where $A_1 \in \mathbb{R}^{n\times n}$ is nonsingular and $A_2 \in \mathbb{R}^{(m-n)\times n}$. Then W and b have corresponding splittings:

$$W = \begin{pmatrix} W_{11} & W_{12} \\ W_{12}^T & W_{22} \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}, \tag{2.2}$$

where $W_{11} \in \mathbb{R}^{n\times n}$ and $W_{22} \in \mathbb{R}^{(m-n)\times(m-n)}$ are symmetric and positive definite, because W is symmetric and positive definite, and $W_{12} \in \mathbb{R}^{n\times(m-n)}$. The normal equation of (1.1) is

$$\begin{pmatrix} A_1 & W_{12} & W_{11} \\ A_2 & W_{22} & W_{12}^T \\ 0 & A_2^T & A_1^T \end{pmatrix} \begin{pmatrix} x \\ r_2 \\ r_1 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ 0 \end{pmatrix}, \tag{2.3}$$

where $r = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix} = W^{-1}(b - Ax)$ is the weighted residual vector corresponding to the splitting form (2.1) of A. We will use preconditioner matrices D_1 and D_2 defined respectively as

$$D_1 = \begin{pmatrix} A_1 & 0 & 0 \\ A_2 & I & 0 \\ 0 & 0 & A_1^T \end{pmatrix} \quad \text{and} \quad D_2 = \begin{pmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & -P^T & I \end{pmatrix},$$

where $P = A_2 A_1^{-1}$. It follows from

$$D_1^{-1} \begin{pmatrix} A_1 & W_{12} & W_{11} \\ A_2 & W_{22} & W_{12}^T \\ 0 & A_2^T & A_1^T \end{pmatrix} D_2 D_2^{-1} \begin{pmatrix} x \\ r_2 \\ r_1 \end{pmatrix} = D_1^{-1} \begin{pmatrix} b_1 \\ b_2 \\ 0 \end{pmatrix}, \tag{2.4}$$

that

$$\begin{pmatrix} I & A_1^{-1}(W_{12} - W_{11}P^{\mathrm{T}}) & 0 \\ 0 & W_{22} - PW_{12} - (W_{12}^{\mathrm{T}} - PW_{11})P^{\mathrm{T}} & W_{12}^{\mathrm{T}} - PW_{11} \\ 0 & 0 & I \end{pmatrix} \begin{pmatrix} x \\ r_2 \\ r_1 + P^{\mathrm{T}} r_2 \end{pmatrix} = \begin{pmatrix} A_1^{-1} b_1 \\ b_2 - Pb_1 \\ 0 \end{pmatrix}. \tag{2.5}$$

Therefore it holds that

$$A_1 x = b_1 - (W_{12} - W_{11}P^{\mathrm{T}}) r_2, \tag{2.6}$$

$$(P, \quad -I)\, W \begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix} r_2 = b_2 - Pb_1, \tag{2.7}$$

and

$$r_1 + P^{\mathrm{T}} r_2 = 0.$$

It is evident that the matrix in the left-hand side of (2.7) is symmetric and positive definite, since W is symmetric and positive definite. Hence the conjugate gradient method can be applied to the reduced system (2.7), and the method converges to the solution of (2.7) in at most $m - n$ steps, in the absence of rounding errors. Next, we present the conjugate gradient algorithm for generalized least squares problems.

Algorithm 2.1.
(1) Factorize A_1, and set initial values $r_2^{(0)} = 0, v^{(0)} = b_2 - Pb_1, p^{(0)} = v^{(0)}$;
(2) Iterate for $k = 0, 1, \ldots,$ until $v^{(k+1)} = 0$ (or $\|v^{(k+1)}\| \leqslant$ tolerance),

$$q = (P, \quad -I)\, W \begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix} p^{(k)},$$

$$\lambda_k = \frac{\|v^{(k)}\|_2^2}{\langle p^{(k)}, q\rangle},$$

$$v^{(k+1)} = v^{(k)} - \lambda_k q,$$

$${r_2}^{k+1} = {r_2}^{k} + \lambda_k p^{(k)},$$

$$\alpha_{k+1} = \frac{\|v^{(k+1)}\|_2^2}{\|v^{(k)}\|_2^2},$$

$$p^{(k+1)} = v^{(k+1)} + \alpha_{k+1} p^{(k)};$$

(3) Solve the extra subsystem

$$A_1 x = b_1 + (W_{11}P^{\mathrm{T}} - W_{12}) r_2^{(l)},$$

where $r_2^{(l)}$ is the solution obtained in step 2.

3. Error bound

In this section, we estimate the error bound for Algorithm 2.1. We need three preparatory results.

Lemma 3.1. *Let S be an arbitrary $m \times n$ matrix, and Q an $m \times m$ symmetric matrix. Then*

$$\kappa(S^{\mathrm{T}}QS) \leqslant \kappa(Q)\kappa(S^{\mathrm{T}}S), \tag{3.1}$$

where $\kappa(B)$ is the spectral condition number of matrix B.

Proof. Since $S^{\mathrm{T}}QS$ and Q are symmetric, there exist orthogonal matrices $U, V \in \mathbb{R}^{m\times m}$ such that

$$U^{\mathrm{T}}S^{\mathrm{T}}QSU = \Sigma = \mathrm{diag}\,(\sigma_1, \sigma_2, \ldots, \sigma_k, 0, \ldots, 0), \tag{3.2}$$

and

$$V^{\mathrm{T}}QV = \Lambda = \mathrm{diag}\,(\lambda_1, \ldots, \lambda_l, 0, \ldots, 0), \tag{3.3}$$

where $\sigma_1 \geqslant \sigma_2 \geqslant \cdots \geqslant \sigma_k > 0$ and $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_l > 0$. Hence

$$\sigma_1 = \max_{\|x\|=1} x^{\mathrm{T}}\Sigma x = \max_{\|x\|=1} (VSUx)^{\mathrm{T}}\Lambda(VSUx) \leqslant \lambda_1 \max_{\|x\|=1} (Ux)^{\mathrm{T}}S^{\mathrm{T}}S(Ux), \tag{3.4}$$

and

$$\sigma_k = \min_{\|x\|=1} x^{\mathrm{T}}\Sigma x = \min_{\|x\|=1} (VSUx)^{\mathrm{T}}\Lambda(VSUx) \geqslant \lambda_l \min_{\|x\|=1} (Ux)^{\mathrm{T}}S^{\mathrm{T}}S(Ux). \tag{3.5}$$

It follows from (3.4) and (3.5) that

$$\kappa(S^{\mathrm{T}}QS) = \frac{\sigma_1}{\sigma_k} \leqslant \kappa(Q)\kappa(S^{\mathrm{T}}S). \qquad \square$$

Lemma 3.2.

$$\kappa\left(\begin{pmatrix} P & -I \end{pmatrix} W \begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix}\right) \leqslant (1 + \|P\|_2^2)\kappa(W), \tag{3.6}$$

where $W \in \mathbb{R}^{m\times m}$ is symmetric, and $P \in \mathbb{R}^{(m-n)\times n}$.

Proof. Apply Lemma 3.1 with $S = \begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix}$, and note that $\kappa(S^{T}S) \leqslant 1 + \|P\|_2^2$. $\square$

Lemma 3.3. *Let the function $f(x)$ be defined by*

$$f(x) = \frac{\left[\dfrac{x-1}{(1+\sqrt{x})^2}\right]^k}{1 + \left[\dfrac{x-1}{(1+\sqrt{x})^2}\right]^{2k}} \tag{3.7}$$

for $x \geqslant 1$. Then $f(x)$ is an increasing function of x for all $x \geqslant 1$ and any $k \geqslant 0$.

Proof. Taking the derivative in (3.7) we obtain

$$f'(x) = \frac{1-y^2}{(1+y^2)^2} \frac{k}{\sqrt{x}} \frac{(\sqrt{x}-1)^{k-1}}{(1+\sqrt{x})^{k+1}} \geqslant 0,$$

where

$$y(x) = \left[\frac{x-1}{(1+\sqrt{x})^2}\right]^k = \left[\frac{\sqrt{x}-1}{1+\sqrt{x}}\right]^k,$$

for all $x \geqslant 1$ and all $k \geqslant 0$, i.e., $f(x)$ is an increasing function of x for all $x \geqslant 1$ and any $k \geqslant 0$. □

Now we can establish the desired error bound.

Theorem 3.4. *The standard error bound based on the Chebyshev polynomials for the CG method applied to the generalized linear least squares problem is given by*

$$\frac{\|A(x^* - x^{(k)})\|_{W^{-1}}}{\|A(x^* - x^{(0)})\|_{W^{-1}}} \leqslant 2 \frac{\left[\dfrac{(1+\alpha^2)\beta - 1}{(1+\sqrt{(1+\alpha^2)\beta})^2}\right]^k}{1 + \left[\dfrac{(1+\alpha^2)\beta - 1}{(1+\sqrt{(1+\alpha^2)\beta})^2}\right]^{2k}}, \tag{3.8}$$

where $\alpha = \|P\|_2 = \|A_2 A_1^{-1}\|_2$, $\beta = \kappa(W) = \mu_{\max}(W)/\mu_{\min}(W)$ is the spectral condition number of the symmetric and positive definite matrix W, $x^{(0)}$ is a vector corresponding to an arbitrary initial vector $r_2^{(0)}$ and x^* is the solution of the problem (1.1).

Proof. It is well known [1] that the conjugate gradient method applied to system (2.7) computes from any starting vector $r_2^{(0)} \in \mathbb{R}^{m-n}$, with residual $r_0 = b_2 - Pb_1 - Er_2^{(0)}$, where

$$E = (P \quad -I)\, W \begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix},$$

and further iterates $r_2^{(k)} \in \mathbb{R}^{m-n}$ with the following minimization property in terms of the norm $\|r_2\|_E$:

$$\|r_2^{(k)} - r_2^*\|_E = \min_{r_2 \in r_2^{(0)} + K_k(r_0;E)} \|r_2 - r_2^*\|_E, \tag{3.9}$$

$$r_2 - r_2^{(0)} \in K_k(r_0;E), \tag{3.10}$$

where r_2^* is the exact solution of the reduced system (2.7) and

$$K_k(r_0;E) = \mathrm{Span}\{r_0, Er_0, \ldots, E^{k-1}r_0\} \tag{3.11}$$

is the kth Krylov subspace of $\mathbb{R}^{m-n}$ generated by r_0 and E.

It is well known that the error bound of the regular conjugate gradient method applied to a symmetric and positive definite system $Sx = d$, is given

$$\frac{\|x^{(k)} - x^*\|_2}{\|x^{(0)} - x^*\|_2} \leqslant 2\frac{\left[\dfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right]^k}{1+\left[\dfrac{\sqrt{\kappa}-1}{\sqrt{\kappa}+1}\right]^{2k}},$$

where κ is the spectral condition number of S [1]. It follows from Lemma 3.2, and Lemma 3.3 and the bound above it

$$\frac{\|r_2^{(k)} - r_2^*\|_E}{\|r_2^{(0)} - r_2^*\|_E} \leqslant 2\frac{\left[\dfrac{(1+\alpha^2)\beta-1}{(1+\sqrt{(1+\alpha^2)\beta})^2}\right]^k}{1+\left[\dfrac{(1+\alpha^2)\beta-1}{(1+\sqrt{(1+\alpha^2)\beta})^2}\right]^{2k}}. \tag{3.12}$$

Since

$$\|r_2 - r_2^*\|_E^2 = (r_2 - r_2^*)^{\mathrm{T}}(P, \quad -I)\,W\begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix}(r_2 - r_2^*) \tag{3.13}$$

and

$$\begin{pmatrix} P^{\mathrm{T}} \\ -I \end{pmatrix}(r_2 - r_2^*) = \begin{pmatrix} P^{\mathrm{T}}(r_2 - r_2^*) \\ -(r_2 - r_2^*) \end{pmatrix}, \tag{3.14}$$

it follows from $r_1 = -P^{\mathrm{T}}r_2$ and $r = \begin{pmatrix} r_1 \\ r_2 \end{pmatrix}$ that

$$\|r_2 - r_2^*\|_E^2 = (r - r^*)^{\mathrm{T}}W(r - r^*) = (x - x^*)^{\mathrm{T}}A^{\mathrm{T}}W^{-1}A(x - x^*) \tag{3.15}$$

using the fact that $r - r^* = W^{-1}A(x - x^*)$. The result follows replacing (3.15) in (3.12).

In Algorithm 2.1, we do not compute effectively the matrix $P = A_2A_1^{-1}$. Instead of computing it, we solve the subsystem $A_1y = c$ and perform the matrix-vector product $z = A_2y$ or $A_1^{\mathrm{T}}y = A_2^{\mathrm{T}}c$ by some direct method. Observe that for many problems n is much smaller than m, so that the size of A_1 is also much smaller than the size W. In these cases, it is much easier to factorize A_1 than W (or B), as is needed when using BNP algorithm [2], Freund's method [4], or James' method [5]. How to get a preconditioner A_1 from a general sparse matrix A will be discussed in [3].

4. Comparison results

In the previous section, we presented the conjugate gradient method for problem (1.1). In this section, we will compare this algorithm with the 2-cyclic SOR algorithm for problem (1.1) presented in Section 1. We start with some known results.

Lemma 4.1. *Consider the linear system*

$$(I - B)x = c.$$

Let B be a weakly 2-cyclic matrix. Suppose that the initial vector $x^{(0)}$ is partitioned in the same way as B so that

$$(I - B)x^{(0)} = c + \begin{pmatrix} 0 \\ u \end{pmatrix}.$$

Then the SOR method for $(I - B)x = c$ generates iterates $x^{(k)}$ with

(a) $x_1^{(1)} = x_1^{(0)}, \quad x_2^{(1)} = x_2^{(0)} - \omega u,$

and for $k = 2, 3, \ldots$

(b) $x_1^{(k)} \in x_1^{(0)} + K_{k-1}(B_1 u; B_{1,2});$

$x_2^{(k)} \in x_2^{(0)} + K_k(u; B_{2,1}),$

where

$$B = \begin{pmatrix} 0 & B_1 \\ B_2 & 0 \end{pmatrix}, \tag{4.1}$$

$$B_{1,2} = B_1 B_2 \quad \text{and} \quad B_{2,1} = B_2 B_1, \tag{4.2}$$

and $K_k(c; B) = \text{Span}\{c, Bc, B^2c, \ldots, B^{k-1}c\}$ is the kth Krylov subspace generated by the vector c and the matrix B.

Proof. See [4]. □

For the system corresponding to Algorithm 1.1, the matrix B of the form is

$$B = \begin{pmatrix} 0 & B_1 \\ B_2 & 0 \end{pmatrix}, \tag{4.3}$$

where

$$B_1 = \begin{pmatrix} -A_1^{-1}(W_{12} - W_{11}P^{\mathrm{T}}) \\ -P^{\mathrm{T}} \end{pmatrix} \tag{4.4}$$

and

$$B_2 = -(\, W_{22}^{-1}A_2 \quad W_{22}^{-1}W_{12}^{\mathrm{T}} \,) \tag{4.5}$$

with the notation of Lemma 4.1. Hence we can extend Freund's result to our problem.

Theorem 4.2. *Suppose that 2-cyclic SOR algorithm is started with initial vectors $x^{(0)}$, $r_1^{(0)}$ and $r_2^{(0)}$. Then the method generates iterates $x^{(k)}$, $r_1^{(k)}$ and $r_2^{(k)}$ with*

$$x^{(1)} = x^{(0)}, \quad r_1^{(1)} = r_1^{(0)}, \quad r_2^{(1)} = r_2^{(0)} - \omega v^{(0)},$$

where

$$v^{(0)} = W_{22}^{-1}b_2 - r_2^{(0)} - W_{22}^{-1}(A_2 x^{(0)} + W_{12}^{\mathrm{T}} r_1^{(0)}), \tag{4.6}$$

and for $k = 2, 3, \ldots$

$$x^{(k)} \in x^{(0)} + A_1^{-1}(W_{12} - W_{11}P^{\mathrm{T}})K_{k-1}(v^{(0)}; E), \tag{4.7}$$

$$r_1^{(k)} \in r_1^{(0)} + P^{\mathrm{T}}K_{k-1}(v^{(0)}; E), \tag{4.8}$$

$$r_2^{(k)} \in r_2^{(0)} + K_k(v^{(0)}; E), \tag{4.9}$$

where $P = A_2A_1^{-1}$, *and*

$$E = W_{22}^{-1}[P(W_{12} - W_{11}P^{\mathrm{T}}) + W_{12}^{\mathrm{T}}P^{\mathrm{T}}]. \tag{4.10}$$

Proof. By Lemma 4.1 with B_1 and B_2 as in (4.4) and (4.5) respectively, we have

$$r_2^{(k)} \in r_2^{(0)} + K_k(v^{(0)}; B_{21}), \tag{4.11}$$

and

$$\begin{pmatrix} x^{(k)} \\ r_1^{(k)} \end{pmatrix} \in \begin{pmatrix} x^{(0)} \\ r_1^{(0)} \end{pmatrix} + K_{k-1}(B_1v^{(0)}; B_{12}). \tag{4.12}$$

Now (4.9) follows from

$$B_{21} = W_{22}^{-1}[P(W_{12} - W_{11}P^{\mathrm{T}}) + W_{12}^{\mathrm{T}}P^{\mathrm{T}}]. \tag{4.13}$$

Since

$$B_{12} = \begin{pmatrix} A_1^{-1}(W_{12} - W_{11}P^{\mathrm{T}})W_{22}^{-1}A_2 & A_1^{-1}(W_{12} - W_{11}P^{\mathrm{T}})W_{22}^{-1}W_{12}^{\mathrm{T}} \\ P^{\mathrm{T}}W_{22}^{-1}A_2 & P^{\mathrm{T}}W_{22}^{-1}W_{12}^{\mathrm{T}} \end{pmatrix} \tag{4.14}$$

and for any index $v \geqslant 0$

$$(B_{12})^v B_1 = B_1(B_{21})^v,$$

we get

$$\begin{aligned} K_{k-1}(B_1v^{(0)}; B_{12}) &= \mathrm{Span}\{B_1v^{(0)}, B_{12}B_1v^{(0)}, \ldots, (B_{12})^{k-2}B_1v^{(0)}\} \\ &= B_1\mathrm{Span}\{v^{(0)}, B_{21}v^{(0)}, \ldots, (B_{21})^{k-2}v^{(0)}\} \\ &= B_1K_{k-1}(v^{(0)}; B_{21}). \end{aligned}$$

The proof is complete. □

Corollary 4.3. *If the* 2-*cyclic SOR algorithm* 1.1 *and the conjugate gradient algorithm* 2.1 *are both started with the same vector* $x^{(0)} \in \mathbb{R}^n$, *then*

$$\|b - Ax_{\mathrm{CG}}^{(k)}\|_{W^{-1}} \leqslant \|b - Ax_{\mathrm{SOR}}^{(k+1)}\|_{W^{-1}}, \quad k = 0, 1, \ldots \tag{4.15}$$

where $x_{\mathrm{CG}}^{(k)}$ *and* $x_{\mathrm{SOR}}^{(k)}$ *are generated by the conjugate gradient algorithm* 2.1 *and the* 2-*cyclic SOR algorithm* 1.1 *respectively.*

Proof. In the conjugate gradient algorithm 2.1, $r_2^{(k)}$ has the following property

$$\begin{aligned} r_2^{(k)} &= \arg\min_{r_2 \in r_2^{(0)}+K_k(v^{(0)};\ -F)} \|r_2 - r_2^*\|_E \\ &= \arg\min_{r_2 \in r_2^{(0)}+K_k(v^{(0)};\ F)} \|r_2 - r_2^*\|_E, \end{aligned} \tag{4.16}$$

where

$$F = W_{22}^{-1}[P(W_{12} - W_{11}P^{\mathrm{T}}) + W_{12}^{\mathrm{T}}P^{\mathrm{T}}],$$

and $E = I - F$. Since

$$\|r_2^{(k)} - r_2^*\|_E = \|b - Ax^{(k)}\|_{W^{-1}},$$

the result follows immediately from Theorem 4.2. □

5. Remarks and conclusions

From the analysis of the convergence behavior of the conjugate gradient method, we know that the error bound given in (3.8) still works in the presence of roundoff errors in which case we may have $r^{(n)} \neq 0$.

If $W = I$, i.e., for regular linear least squares problems, and $\kappa(W) = 1$, it follows from (3.6) that $\kappa(E) \leqslant 1 + \|P\|_2^2$. Therefore (3.8) reduces to

$$\frac{\|A(x^* - x^{(k)})\|_2}{\|A(x^* - x^{(0)})\|_2} \leqslant 2\frac{\left[\dfrac{\alpha}{1+\sqrt{1+\alpha^2}}\right]^{2k}}{1+\left[\dfrac{\alpha}{1+\sqrt{1+\alpha^2}}\right]^{4k}}, \tag{5.1}$$

which coincides with the result given by Freund in [4].

Algorithm 2.1 obtains an $(m-n)$-dimensional vector r_2 by the conjugate gradient method, and then solves the extra subsystem (2.7). The algorithm requires only A_1^{-1} and W but not W^{-1}. So this algorithm is more efficient in actual applications than Freund's algorithm, which requires W^{-1}. In the case of $W_{12} = 0$, we can apply the conjugate gradient method to the reduced systems given in [9] and get another preconditioned conjugate gradient algorithm which will reduce to other known preconditioned conjugate gradient algorithms. Theorem 4.2 and Corollary 4.3 prove that Freund's result [4] is true also for the generalized least squares problem. The result in Corollary 4.3 reduces to Freund's result and James' result [5] respectively for the regular least squares problem and the equality constrained least squares problem.

Our results show that the preconditioned conjugate gradient algorithm is preferable to SOR method for solving generalized least squares problems.

Numerical results presented in [9] show that though the preconditioned conjugate gradient method is numerically less stable than Paige's method, it is much faster than this procedure, especially for sparse problems. Also the method works for semi-definite W, even when $\operatorname{rank}(W) < m - n$, in which case the BNP algorithm does not work.

Acknowledgements

We thank Dr. Å. Björck, who conjectured the result of Corollary 4.3 and encouraged us to study the issue.

References

[1] O. Axelsson and G. Lindskog, On the rate of convergence of the preconditioned conjugate gradient method, *Numer. Math.* **48** (1986) 499–523.
[2] J.L. Barlow, N.K. Nichols and R.J. Plemmons, Iterative methods for equality constrained least squares problems, *SIAM J. Sci. Statist. Comput.* **9** (1988) 892–906.
[3] Å. Björck and J.Y. Yuan, Preconditioner for least squares problems by LU factorization, to be published.
[4] R. Freund, A note on two block SOR methods for sparse least-squares problems, Linear Algebra and its Appl. 88/89 (1987) 211–221.
[5] D.J. James, Conjugate gradient methods for constrained least squares problems, Ph.D Dissertation, Dept. of Math., North Carolina State University, Raleigh, 1990.
[6] D.J. James, Implicit nullspace iterative methods for constrained least squares problems, *SIAM J. Matrix Anal. Appl.* **13** (1992) 962–978.
[7] C.C. Paige, Fast numerically stable computations for generalized least-squares problems, *SIAM J. Numer. Anal.* **16** (1979) 165–171.
[8] C.C. Paige, Computer solution and perturbation analysis of generalized linear least-squares problems, *Math. Comp.* **33** (1979) 171–184.
[9] J.Y. Yuan, Iterative methods for generalized least squares problems, Ph.D. Thesis, IMPA, Brazil, 1993.