Journal of Computational and Applied Mathematics 317 (2017) 203–217

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

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

Block Conjugate Gradient algorithms for least squares problems

Hao Ji [a], Yaohang Li [b,*]

[a] Department of Computer Science, California State Polytechnic University, Pomona, United States

[b] Department of Computer Science, Old Dominion University, Norfolk, VA 23529, United States

ARTICLE INFO

Article history:
Received 4 May 2015
Received in revised form 21 July 2016

Keywords:
Block Krylov subspace
Block Conjugate Gradient Least Squares
Rank deficiency
Breakdown-Free
Deflation

ABSTRACT

In this paper, extensions for the Conjugate Gradient Least Squares (CGLS) algorithm in block forms, so-called Block Conjugate Gradient Least Squares (BCGLS), are described. Block parameter matrices are designed to explore the block Krylov subspace so that multiple right-hand sides can be treated simultaneously, while maintaining orthogonality and minimization properties along iterations. Search subspace is reduced adaptively in case of (near) rank deficiency to prevent breakdown. A deflated form of BCGLS is developed to accelerate convergence. Numerical examples demonstrate effectiveness in handling rank deficiency and efficiency in convergence accelerations in these BCGLS forms.

1. Introduction

We consider the problem of stably finding the least squares solutions to a linear system of equations with multiple right-hand sides,

$$AX = B$$

where A is an $m \times n(m \geq n)$ sparse, rectangular or square matrix with rank n, X is an $n \times s$ unknown matrix, B is an $m \times s$ right-hand side matrix, and $s(s \geq 1)$ is the number of right-hand sides. When A is large and sparse, block iterative methods are natural candidates for solving the least squares problem with multiple right-hand sides.

Using block methods to solve the least squares problems has three major advantages. First of all, solutions corresponding to multiple right-hand sides can be estimated simultaneously. This is particularly useful for applications such as multi-objective optimization [1] interested in finding solutions with respect to different right-hand side vectors. Secondly, compared to solvers with a single right-hand side, block methods can potentially accelerate convergence by exploring multiple directions in Krylov subspace at each iteration. Thirdly, a block formulation can lead to computational efficiency [2–4] for linear systems involving very large coefficient matrices. In particular, when the coefficient matrix A is too large to fit in core memory or the elements of A need to be reproduced every time in use, a matrix–vector multiplication, which requires a pass over all elements in A, is very computational costly. In this situation, passing over A becomes the main computational bottleneck. Computing the action of A on multiple vectors at once adds little to the overall cost of computing a single matrix–vector product but significantly reduces the total number of passes over A. Moreover, if $s \ll n$, the block methods involve a lot of multiplication operations on "tall-and-skinny" matrices, which can be easily parallelized with Level 3 BLAS subroutines [5–7].

[*] Corresponding author.
E-mail address: yaohang@cs.odu.edu (Y. Li).

http://dx.doi.org/10.1016/j.cam.2016.11.031
0377-0427/© 2016 Elsevier B.V. All rights reserved.

Given a general linear system

$$Ax = b,$$

where $A \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$, and $b \in \mathbb{R}^m$, the Conjugate Gradient (CG) method can be applied for minimizing the L_2 norm of the residual error, $\|r\|_2^2 = \|Ax - b\|_2^2$, in the corresponding normal equation. If the linear system is underdetermined ($m < n$), the Conjugate Gradient Normal Equation (CGNE) method [8] solves the normal equation $AA^Ty = b$ for y and then obtains the solution x by computing $x = A^Ty$. If the system is overdetermined ($m > n$), this becomes the least squares problem. The Conjugate Gradient for Least Squares (CGLS) method, also known as the Conjugate Gradient Normal Residual (CGNR) method [9,10], solves $A^TAx = A^Tb$. The LSQR method [11,12] is derived from the Lanczos process with Golub–Kahan bi-diagonalization to handle the least squares problems, which has been regarded as a mathematical equivalence of CGLS. More recently, the LSMR method [13], a variant of LSQR, is developed to approximate least squares solutions, which is equivalent to MINRES [14] for solving $A^TAx = A^Tb$.

In this paper, we extend the Conjugate Gradient for Least Squares (CGLS) algorithm [9,15] to a Block Conjugate Gradient for Least Squares (BCGLS) algorithm to handle least squares problems with multiple right-hand sides. Block matrix operations in BCGLS are developed to approximate the least squares solutions by ensuring orthogonality properties while minimizing the residual error function

$$Trace\left((B - AX)^T\,(B - AX)\right),$$

over the underlying Krylov subspace, where $trace(\cdot)$ denotes the trace of a matrix. New forms of parameter matrices are derived to reduce the basis of search space while maintaining orthogonality to avoid potential breakdown in case of rank deficiency. Deflation techniques handling the extreme eigenvalues are applied in BCGLS to achieve convergence acceleration.

The rest of the paper is organized as follows. We firstly review the CGLS algorithm in Section 2. Then, in Section 3, we present the BCGLS algorithm to handle the least squares problem with multiple right-hand sides. An improved form of the BCGLS algorithm to handle rank deficiency is designed in Section 4. In Section 5, we describe the deflation techniques that can be applied to the BCGLS algorithm to accelerate convergence. Numerical examples are reported in Section 6. Finally, Section 7 summarizes our conclusions and future research directions.

2. CGLS algorithm

CGLS Algorithm

Input: $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, initial guess $x_0 \in \mathbb{R}^n$, tolerance $tol \in \mathbb{R}$, and maximum number of iterations $maxit \in \mathbb{R}$.
Output: an approximate solution $x_{sol} \in \mathbb{R}^n$.

```
$r_0 = b - Ax_0$
$s_0 = p_0 = A^Tr_0$
$\gamma_0 = \|s_0\|^2$
for $i = 0, \cdots, maxit$
    $q_i = Ap_i$
    $\alpha_i = \gamma_i/\|q_i\|^2$
    $x_{i+1} = x_i + \alpha_i p_i$
    $r_{i+1} = r_i - \alpha_i q_i$
    if converged within $tol$, then stop.
    $s_{i+1} = A^Tr_{i+1}$
    $\gamma_{i+1} = \|s_{i+1}\|^2$
    $\beta_i = \gamma_{i+1}/\gamma_i$
    $p_{i+1} = s_{i+1} + \beta_i p_i$
end
$x_{sol} = x_{i+1}$
```

CGLS was originally proposed by Hestenes and Stiefel [9] for solving the least squares problem. Starting from an initial solution guess x_0, a sequence of estimates $\{x_1, x_2, \ldots\}$ is generated to approximate the solution along the iterations in CGLS. Vectors r_i and p_i denote the residual and the search direction at the ith iteration, respectively. Parameters α_i and β_i are computed in a way to ensure that the following two important orthogonality properties hold along CGLS iterations,

(i) $p_j^TA^Tr_{i+1} = 0$; $(j < i + 1)$
(ii) $p_j^TA^TAp_{i+1} = 0$ $(j < i + 1)$.

As a result, the new solution x_{i+1} obtained at the ith iteration is not only the optimal approximation along the search direction p_i, but also a global minimizer of the residual error function $\|Ax-b\|_2^2$ over the exploited Krylov subspace,

$$span\left\{A^T r_0, \left(A^T A\right) A^T r_0, \ldots, \left(A^T A\right)^i A^T r_0\right\}.$$

CGLS is a memory-efficient algorithm in approximating the least squares solutions, where all previous search directions are not necessary to be stored. Moreover, CGLS is mathematically equivalent to the Conjugate Gradient (CG) method on solving the normal equation $A^TAx = A^Tb$, and thus CGLS has the convergence rate of $\left(\frac{1-\sqrt{\kappa^{-1}}}{1+\sqrt{\kappa^{-1}}}\right)^2$, where $\kappa = \lambda_n/\lambda_1$, λ_n and λ_1 are the nth and 1st eigenvalues of A^TA ordered by magnitude, respectively. Furthermore, CGLS avoids the explicit computation of A^TA during iterations and appears to be more accurate compared to the other variants of CG for solving the least squares problem [16,17].

3. Block Conjugate Gradient Least Squares (BCGLS) algorithm

BCGLS considers a linear system in block form with s right-hand sides in the block matrix B,

$$AX = B.$$

Compared to obtaining solution for each right-hand side individually in CGLS, BCGLS is a more effective scheme by treating all right-hand sides simultaneously. BCGLS constructs Krylov subspace in block form [18,19], i.e.,

$$\text{block-span}\left(A^T R_0, \left(A^T A\right) A^T R_0, \ldots, \left(A^T A\right)^i A^T R_0, \ldots\right).$$

Here, 'block-span' is defined as

$$\left\{\sum_{i=0} (A^TA)^i A^T R_0 \Psi_i\right\},$$

where $\Psi_i \in \mathbb{R}^{s\times s}$ are related to the parameter matrices α_j and β_j ($j \le i$) in BCGLS. Then, a block quadratic function is minimized over the block Krylov subspace, i.e.,

$$\min_X Trace\left((B-AX)^T (B-AX)\right).$$

Block CGLS (BCGLS) Algorithm

Input: $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times s}$, initial guess $X_0 \in \mathbb{R}^{n\times s}$, tolerance $tol \in \mathbb{R}$, and maximum number of iterations $maxit \in \mathbb{R}$.
Output: solution $X_{sol} \in \mathbb{R}^{n\times s}$.

```
$R_0 = B - AX_0$
$S_0 = A^T R_0$
$P_0 = S_0$
for $i = 0, \cdots, maxit$
    $Q_i = AP_i$
    $\alpha_i = (Q_i^T Q_i)^{-1} S_i^T S_i$
    $X_{i+1} = X_i + P_i\alpha_i$
    $R_{i+1} = R_i - Q_i\alpha_i$
    if converged within $tol$, then stop.
    $S_{i+1} = A^T R_{i+1}$
    $\beta_i = (S_i^T S_i)^{-1} S_{i+1}^T S_{i+1}$
    $P_{i+1} = S_{i+1} + P_i\beta_i$
end
$X_{sol} = X_{i+1}$
```

BCGLS starts with an $n \times s$ initial solution matrix X_0. At the ith iteration, the new approximate block solution X_{i+1} is updated by $X_i + P_i\alpha_i$ and its residual matrix R_{i+1} is evaluated in the search space spanned by P_i. A new search matrix P_{i+1} specified by $S_{i+1} + P_i\beta_i$ is evaluated. Notice that both α_i and β_i are now in matrix form, different from the scalar parameters of α and β in CGLS. Similar to CGLS, orthogonality among vectors in the block residual matrices as well as the search matrices is required to ensure the minimization property in the underlying Krylov subspace. Assuming the existences of $(Q_i^T Q_i)^{-1}$ and $(S_i^T S_i)^{-1}$, Lemma 1 shows that such orthogonality properties hold. We first prove Lemma 1 by induction and then we show the minimization property of BCGLS in Theorem 2.

Lemma 1. *The following orthogonality relations hold for the residual matrices and the search matrices in the BCGLS Algorithm,*

(i) $P_j^TA^TR_{i+1}=0; \quad (j<i+1)$
(ii) $P_j^TA^TAP_{i+1}=0 \ (j<i+1)$.

Proof. Lemma 1 is proved by induction.

Base case: when $i=0$, it is easy to find that

$$P_0^TA^TR_1=P_0^TA^TR_0-P_0^TA^TAP_0\alpha_0=0.$$

Since $R_0^TAA^TR_1=P_0^TA^TR_1=0$, we have

$$\begin{aligned}P_0^TA^TAP_1&=P_0^TA^TAS_1+P_0^TA^TAP_0\beta_0\\&=P_0^TA^TAS_1+P_0^TA^TAP_0(S_0^TS_0)^{-1}S_1^TS_1\\&=-(\alpha_0^T)^{-1}R_1^TAA^TR_1+(\alpha_0^T)^{-1}R_0^TAA^TR_1+P_0^TA^TAP_0(S_0^TS_0)^{-1}S_1^TS_1\\&=0.\end{aligned}$$

Induction step: we assume that $P_j^TA^TR_k=0$, $R_j^TAA^TR_k=0$, and $P_j^TA^TAP_k=0$ hold for all $j<k$. Then, we examine $P_j^TA^TR_{k+1}$ and $P_j^TA^TAP_{k+1}(j<k+1)$,

(i) $P_j^TA^TR_{k+1}=P_j^TA^TR_k-P_j^TA^TAP_k\alpha_k$;
If $j<k$, according to the induction hypothesis, $P_j^TA^TR_{k+1}=0$;
If $j=k$, we have

$$\begin{aligned}P_k^TA^TR_{k+1}&=P_k^TA^TR_k-P_k^TA^TAP_k\alpha_k\\&=(S_k+P_{k-1}\beta_{k-1})^T A^TR_k-S_k^TS_k\\&=S_k^TS_k+\beta_{k-1}^TP_{k-1}^TA^TR_k-S_k^TS_k\\&=0.\end{aligned}$$

(ii) In order to prove $P_j^TA^TAP_{k+1}=0$, we first show that $R_j^TAA^TR_{k+1}=0$.
If $j=0$, it follows that $R_0^TAA^TR_{k+1}=R_0^TAA^TR_k-R_0^TAA^TQ_k\alpha_k=0$;
If $k\geq j\geq 1$, we have

$$\begin{aligned}R_j^TAA^TR_{k+1}&=R_j^TAA^TR_k-R_j^TAA^TQ_k\alpha_k\\&=R_j^TAA^TR_k-(P_j^TA^TAP_k-\beta_{j-1}^TP_{j-1}^TA^TAP_k)\alpha_i\\&=0.\end{aligned}$$

Based on $R_j^TAA^TR_{k+1}=0$, for $j<k$, $P_j^TA^TAP_{k+1}$ can be written as

$$\begin{aligned}P_j^TA^TAP_{k+1}&=P_j^TA^TAS_{k+1}+P_j^TA^TAP_k\beta_k\\&=-(\alpha_j^T)^{-1}(R_{j+1}-R_j)^TAS_{k+1}+P_j^TA^TAP_k\beta_k\\&=-(\alpha_j^T)^{-1}R_{j+1}^TAA^TR_{k+1}+(\alpha_j^T)^{-1}R_j^TAA^TR_{k+1}+P_j^TA^TAP_k\beta_k\\&=0.\end{aligned}$$

When $j=k$, we have

$$\begin{aligned}P_k^TA^TAP_{k+1}&=-(\alpha_k^T)^{-1}R_{k+1}^TAA^TR_{k+1}+P_k^TA^TAP_k(S_k^TS_k)^{-1}S_{k+1}^TS_{k+1}\\&=-(\alpha_k^T)^{-1}S_{k+1}^TS_{k+1}+(\alpha_k^T)^{-1}S_{k+1}^TS_{k+1}\\&=0.\end{aligned}$$

Conclusion: Based on the principle of induction, statements (i) and (ii) are true for all $j<i+1$. □

Theorem 2. *At the ith iteration of BCGLS, X_{i+1} is a global minimizer of the block residual error function*

$$Trace\left((B-AX)^T(B-AX)\right),$$

over the block Krylov subspace $X_0+\text{block-span}\left\{A^TR_0,\left(A^TA\right)A^TR_0,\ldots,\left(A^TA\right)^iA^TR_0\right\}$.

Proof. Let $x_{i+1}^{(k)}$ denote the kth column of X_{i+1} and let $b^{(k)}$ be the kth column of B. The quadratic function $f^{(k)}(x)$ for each right-hand side in B is defined as

$$\begin{aligned} f^{(k)}(x) &= (b^{(k)} - Ax)^T (b^{(k)} - Ax) \\ &= x^T A^T Ax - 2{b^{(k)}}^T Ax - {b^{(k)}}^T b^{(k)}. \end{aligned}$$

Since $x_{i+1}^{(k)}$ is a point in $x_0^{(k)}$ + block-span $\{P_0, P_1, \ldots, P_i\}$, it can be expressed as

$$\begin{aligned} x_{i+1}^{(k)} &= x_0^{(k)} + P_0\alpha_0^{(k)} + P_1\alpha_1^{(k)} + \cdots + P_i\alpha_i^{(k)} \\ &= x_0^{(k)} + [P_0, P_1, \ldots, P_i]\, \theta_i^{(k)}, \end{aligned}$$

where $\theta_i^{(k)} = \begin{bmatrix} \alpha_0^{(k)} \\ \alpha_1^{(k)} \\ \vdots \\ \alpha_i^{(k)} \end{bmatrix}$ and $\alpha_j^{(k)}$ is the kth column of parameter matrix α_j $(j \le i)$.

Differentiating $f^{(k)}(\cdot)$ with respect to $\theta_i^{(k)}$, we have

$$\frac{df^{(k)}\left(x_{i+1}^{(k)}\right)}{d\theta_i^{(k)}} = -2\left([P_0, P_1, \ldots, P_i]\right)^T A^T r_{i+1}^{(k)},$$

where $r_{i+1}^{(k)} = b^{(k)} - Ax_{i+1}^{(k)}$ is the kth column of R_{i+1}. According to Lemma 1, $P_j^T A^T R_{i+1} = 0$ for all $j < i+1$, we have

$$\frac{df^{(k)}\left(x_{i+1}^{(k)}\right)}{d\theta_i^{(k)}} = 0,$$

which implies that $x_{i+1}^{(k)}$ is the minimizer of $f^{(k)}(x)$ over $x_0^{(k)}$ + block-span $\{P_0, P_1, \ldots, P_i\}$.

As the search spaces P_j's are constructed from the underlying block Krylov subspace, block-span $\{P_0, P_1, \ldots, P_i\}$ is an equivalent space to

$$\text{block-span}\left\{A^T R_0, \left(A^T A\right) A^T R_0, \ldots, \left(A^T A\right)^i A^T R_0\right\}.$$

Because of

$$Trace\left((B - AX_{i+1})^T (B - AX_{i+1})\right) = \sum_{k=0}^{s-1} f^{(k)}\left(x_{i+1}^{(k)}\right),$$

where the s quadratic functions $f^{(k)}(\cdot)$, $k = 0, \ldots, s-1$, for the s linear systems corresponding to the s right-hand sides are independent, X_{i+1} obtained from BCGLS is the global minimizer of the block quadratic function in the exploited block Krylov subspace X_0 + block-span $\left\{A^T R_0, \left(A^T A\right) A^T R_0, \ldots, \left(A^T A\right)^i A^T R_0\right\}$. □

The minimization property of BCGLS indicates that an optimal point is chosen from X_0 + block-span $\left\{A^T R_0, \left(A^T A\right) A^T R_0, \ldots, \left(A^T A\right)^i A^T R_0\right\}$ to be the least squares approximation at the ith iteration step, which can be expressed as a polynomial $\Phi_i(A^T A)$ of degree i, i.e.,

$$\Phi_i(A^T A) = X_0 + \sum_{j=0}^{i} (A^T A)^j A^T R_0 \Psi_j,$$

where the coefficient matrices Ψ_j are related to the parameter matrices α_k and β_k $(k \le j)$ in BCGLS. Therefore, similar to the Block Conjugate Gradient (BCG) methods [20,19], BCGLS yields a faster convergence rate of $\left(\frac{1-\sqrt{\kappa'^{-1}}}{1+\sqrt{\kappa'^{-1}}}\right)^2$ compared to CGLS, where $\kappa' = \lambda_n/\lambda_s$, and λ_n and λ_s are the nth and sth eigenvalues of the product matrix $A^T A$ ordered by magnitude, respectively.

4. Addressing the (near) breakdown problem

Despite the attractive advantages of block methods, similar to other block solvers [20–22], one practical issue of BCGLS is the breakdown problem that potentially occurs along iterations. More specifically, during BCGLS iterations, some vectors in the block matrices may become linearly dependent or zero, which leads to rank deficiency in these block matrices.

Consequently, generating at least one of the parameter matrices has to evaluate the inverse of a singular matrix, which results in BCGLS breakdown. Many situations can give rise to rank deficiency of block matrices, for instances, inappropriate guess of initial vectors, unbalanced convergence speeds of multiple right hand sides, and accumulation of roundoff errors, which have been analyzed in our previous work [19].

To ensure the numerical stability of block Krylov subspace methods, in general, there are several strategies in handling rank deficiency situations, including restarting [20,23,24], keeping linearly dependent vectors and reintroducing in later iterations [25,21,26], and reducing search space [27,19,22,18,28]. We here extend our previous work of Breakdown-Free Conjugate Gradient (BFBCG) [19] to the more general block least squares problems.

Breakdown-Free BCGLS (BFBCGLS) Algorithm

```
**Input:** $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times s}$, initial guess $X_0 \in \mathbb{R}^{n\times s}$, tolerance $tol \in \mathbb{R}$, and maximum number of iterations $maxit \in \mathbb{R}$.
**Output:** an approximate solution $X_{sol} \in \mathbb{R}^{n\times s}$.

$R_0 = B - AX_0$
$S_0 = A^T R_0$
$\widetilde{P}_0 = orth(S_0)$
for $i = 0, \cdots, maxit$
    $\widetilde{Q}_i = A\widetilde{P}_i$
    $\widetilde{\alpha}_i = (\widetilde{Q}_i^T \widetilde{Q}_i)^{-1} \widetilde{Q}_i^T R_i$
    $X_{i+1} = X_i + \widetilde{P}_i \widetilde{\alpha}_i$
    $R_{i+1} = R_i - \widetilde{Q}_i \widetilde{\alpha}_i$
    if converged within $tol$, then stop.
    $S_{i+1} = A^T R_{i+1}$
    if no rank deficiency occurs, then
        $\widetilde{\beta}_i = (S_i^T S_i)^{-1} S_{i+1}^T S_{i+1}$,
    else
        $\widetilde{\beta}_i = -(\widetilde{Q}_i^T \widetilde{Q}_i)^{-1} \widetilde{Q}_i^T A S_{i+1}$
    endif
    $\widetilde{P}_{i+1} = orth(S_{i+1} + \widetilde{P}_i \widetilde{\beta}_i)$
end
$X_{sol} = X_{i+1}$
```

Algorithm Breakdown-Free BCGLS (BFBCGLS) presents a simple solution to address the potential breakdown problem caused by rank deficiency in BCGLS. There are two key differences on the parameter matrices between BFBCGLS and BCGLS, and here we use the matrix symbols with a "∼" notation to distinguish them. First, a rank revealing operation $orth(\cdot)$ is applied to the search direction blocks $\widetilde{P}_i$ to remove linearly dependent or zero vectors. When rank deficiency occurs at the ith iteration, the dimension of space $\mathcal{P}_i$ spanned by $\widetilde{P}_i$ reduces from s to $r_i (r_i < s)$ and correspondingly the search block $\widetilde{P}_i$ shrinks to be an $n \times r_i$ matrix. Second, with respect to the change in search direction block $\widetilde{P}_i$, the parameter matrices $\widetilde{\alpha}_i$ and $\widetilde{\beta}_i$ become $r_i \times s$ rectangular matrices and $\widetilde{Q}_i$ appears to be $m \times r_i$. Either QR factorization with column pivoting or a Singular Value Decomposition (SVD) on $S_{i+1} + \widetilde{P}_i \widetilde{\beta}_i$ can be used to detect the reduction of the dimension of search space $\mathcal{P}_{i+1}$ and to construct an orthogonal basis $\widetilde{P}_{i+1}$ of the reduced search space $\mathcal{P}_{i+1}$. Due to the fact that the minimization property of BCGLS relies on the orthogonality among block residual matrices as well as search matrices, Theorems 3 and 6 show that there exist parameter matrices $\widetilde{\alpha}_i$ and $\widetilde{\beta}_i$ that guarantee such orthogonality properties in case of rank deficiency, respectively.

Theorem 3. *Suppose that rank deficiency occurs at the ith iteration. Let $\widetilde{P}_i \in \mathbb{R}^{n\times r_i}$ be an orthonormal basis of search space $\mathcal{P}_i$ with dimension $r_i (r_i < s)$ at the ith iteration and $\widetilde{Q}_i = A\widetilde{P}_i$. Then, there exists a parameter matrix $\widetilde{\alpha}_i \in \mathbb{R}^{r_i \times s}$*

$$\widetilde{\alpha}_i = (\widetilde{Q}_i^T \widetilde{Q}_i)^{-1} \widetilde{Q}_i^T R_i,$$

so that

(i) $\widetilde{P}_i^T A^T R_{i+1} = 0$;

(ii) $\widetilde{P}_j^T A^T R_{i+1} = 0$ *(for all $j < i + 1$), if all previous search spaces $\mathcal{P}_j (j \le i)$ are $A^T A$-orthogonal.*

Proof. Since A is an $m \times n (m \ge n)$ matrix of rank n and $\widetilde{P}_i \in \mathbb{R}^{n\times r_i}$ is an orthonormal basis of search space $\mathcal{P}_i$, $\widetilde{Q}_i^T \widetilde{Q}_i = \widetilde{P}_i^T A^T A \widetilde{P}_i \in \mathbb{R}^{r_i \times r_i}$ is nonsingular. Therefore, there exists a parameter matrix $\widetilde{\alpha}_i \in \mathbb{R}^{r_i \times s}$ such that $\widetilde{\alpha}_i = (\widetilde{Q}_i^T \widetilde{Q}_i)^{-1} \widetilde{Q}_i^T R_i$.

(i) Based on residual recurrence formula

$$R_{i+1} = R_i - \widetilde{Q}_i\widetilde{\alpha}_i = R_i - A\widetilde{P}_i\widetilde{\alpha}_i, \tag{1}$$

left multiplying both sides of (1) by $\widetilde{P}_i^T A^T$, we have

$$\begin{aligned}\widetilde{P}_i^T A^T R_{i+1} &= \widetilde{P}_i^T A^T R_i - \widetilde{P}_i^T A^T A\widetilde{P}_i\widetilde{\alpha}_i\\ &= \widetilde{P}_i^T A^T R_i - \widetilde{P}_i^T A^T A\widetilde{P}_i(\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^T R_i\\ &= 0.\end{aligned}$$

(ii) Left multiplying both sides of (1) by $\widetilde{P}_{i-1}^T A^T$ and then

$$\widetilde{P}_{i-1}^T A^T R_{i+1} = \widetilde{P}_{i-1}^T A^T R_i - \widetilde{P}_{i-1}^T A^T A\widetilde{P}_i\widetilde{\alpha}_i.$$

Based on (i), with derivation of $\widetilde{\alpha}_{i-1}$ given in BFBCGLS, $\widetilde{P}_{i-1}^T A^T R_i = 0$ holds. Since all previous search spaces $\mathcal{P}_j (j \leq i)$ are $A^T A$-orthogonal, we have $\widetilde{P}_{i-1}^T A^T A\widetilde{P}_i = 0$ and thus $\widetilde{P}_{i-1}^T A^T R_{i+1} = 0$. By induction, we can conclude that $\widetilde{P}_j^T A^T R_{i+1}$ for all $j < i+1$. □

According to Theorem 3, other orthogonality relations stated in Corollaries 4 and 5 can also be obtained. These orthogonality properties are used in the proof of Theorem 6. The proofs for Corollaries 4 and 5 are included in the Appendix.

Corollary 4. *Assuming that all previous search spaces $\mathcal{P}_j (j \leq i)$ are $A^T A$-orthogonal, then $R_{i+1}^T A A^T R_j = 0$, for $j < i+1$.*

Corollary 5. *$R_{i+1}^T A A^T A\widetilde{P}_j = 0$, for $j < i$.*

Maintaining $A^T A$-orthogonality among search spaces is critical to ensure the minimization property of BCGLS, as analyzed in Section 3. Theorem 6 shows that the derivation of parameter matrix $\widetilde{\beta}_i$ can lead to an orthogonal basis $\widetilde{P}_{i+1}$ $A^T A$-orthogonal to all previous search spaces.

Theorem 6. *Suppose that rank deficiency occurs at the ith iteration in BCGLS. Let $\widetilde{P}_i \in \mathbb{R}^{n\times r_i}$ be an orthonormal basis of search space $\mathcal{P}_i$ with dimension $r_i (r_i < s)$, $\widetilde{Q}_i = A\widetilde{P}_i$, and $S_{i+1} = A^T R_{i+1}$. Then, there exists a parameter matrix $\widetilde{\beta}_i \in \mathbb{R}^{r_i\times s}$*

$$\widetilde{\beta}_i = -(\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^T AS_{i+1},$$

so that the new search space $\mathcal{P}_{i+1}$ obtained from $A^T R_{i+1}$ is $A^T A$-orthogonal to all previous search spaces $\mathcal{P}_j$ where $j < i+1$.

Proof. Since A is an $m \times n (m \geq n)$ sparse matrix of rank n and $\widetilde{P}_i \in \mathbb{R}^{n\times r_i}$ is an orthonormal basis of search space $\mathcal{P}_i$, $\widetilde{Q}_i^T\widetilde{Q}_i = \widetilde{P}_i^T A^T A\widetilde{P}_i \in \mathbb{R}^{r_i\times r_i}$ is nonsingular. Therefore, there exists a series of matrices $\widetilde{\beta}_i \in \mathbb{R}^{r_i\times s}$ such that $\widetilde{\beta}_i = -(\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^T AS_{i+1}$.

Since P_{i+1} can be derived from

$$P_{i+1} = S_{i+1} + \widetilde{P}_i\widetilde{\beta}_i, \tag{2}$$

where $S_{i+1} = A^T R_{i+1}$ and $\widetilde{\beta}_i$ is the associated weight matrix of $\widetilde{P}_i$, we first show that $\widetilde{P}_j^T A^T A\widetilde{P}_{i+1} = 0$, for all $j < i$.

Left multiplying (2) both sides by $\widetilde{P}_j^T A^T A$ where $j < i$, we have

$$\begin{aligned}\widetilde{P}_j^T A^T AP_{i+1} &= \widetilde{P}_j^T A^T AS_{i+1} + \widetilde{P}_j^T A^T A\widetilde{P}_i\widetilde{\beta}_i\\ &= \widetilde{P}_j^T A^T AA^T R_{i+1} + \widetilde{P}_j^T A^T A\widetilde{P}_i\widetilde{\beta}_i\\ &= \widetilde{P}_j^T A^T AA^T R_{i+1}.\end{aligned}$$

According to Corollary 5, we can get $R_{i+1}^T AA^T A\widetilde{P}_j = 0$ for all $j < i$. Assuming that $\widetilde{P}_j^T A^T A\widetilde{P}_i = 0 (j < i)$, $\widetilde{P}_j^T A^T AP_{i+1} = 0$ is obtained for all $j < i$.

Then, we can show that $\widetilde{P}_i^T A^T AP_{i+1} = 0$ by left multiplying (2) both sides by $\widetilde{P}_i^T A^T A$,

$$\begin{aligned}\widetilde{P}_i^T A^T AP_{i+1} &= \widetilde{P}_i^T A^T AA^T R_{i+1} + \widetilde{P}_i^T A^T A\widetilde{P}_i\widetilde{\beta}_i\\ &= \widetilde{P}_i^T A^T AA^T R_{i+1} - \widetilde{P}_i^T A^T A\widetilde{P}_i(\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^T AS_{i+1}\\ &= \widetilde{P}_i^T A^T AA^T R_{i+1} - \widetilde{P}_i^T A^T AA^T R_{i+1}\\ &= 0.\end{aligned}$$

Let the range space of P_{i+1} be the new search space $\mathcal{P}_{i+1}$ and then the new search space $\mathcal{P}_{i+1}$ is $A^T A$-orthogonal to all previous search spaces $\mathcal{P}_j (j < i+1)$. □

In case of rank deficiency, the rectangular parameter matrices $\widetilde{\alpha}_i$ and $\widetilde{\beta}_i$ are calculated without estimating the inverse of a non-full rank matrix in BFBCGLS. As a result, the orthogonality properties are maintained in search space and the breakdown problem due to rank deficiency can be avoided.

In practice, it is very rare that the residual block has exact linear dependency in BCGLS; however, much more often, vectors in the residual block will become nearly linearly dependent. Studies [29,30] have shown that the nearly linear dependency in the residual block may cause near-breakdown and have a serious impact to convergence of block Krylov subspace methods. In BFBCGLS, linear dependency in the residual block S_i is monitored. If the smallest eigenvalue of S_i is lower than a designated threshold parameter τ, the search space will be reduced accordingly in BFBCGLS. The linear dependency threshold parameter τ has an impact on solution precision as well as convergence speed and needs to be carefully selected, which will be analyzed in Section 6.

5. BCGLS algorithm with deflation

Deflation is one of the popular techniques used in Krylov subspace methods to accelerate convergence via pre-adding the Krylov subspace with a space spanned by a deflation matrix [31–33], which contains approximations to the extreme eigenvectors. Deflation has been used widely to handle positive definite systems [34,35] and unsymmetric systems [36–39]. Recently, when multiple right-hand sides in a linear system are considered, deflation has been applied to BCG [23] and BGMRES [26]. More comprehensive analysis on Krylov subspace methods with deflation can be found in [32,31,26,40].

Deflation can also be applied to BCGLS to improve convergence in finding solutions for least squares problems. Given a deflation matrix W, an augmented block Krylov subspace

$$\text{block-span}\left\{W, A^TR_0, \left(A^TA\right)A^TR_0, \dots, \left(A^TA\right)^iA^TR_0, \dots\right\},$$

is constructed. In BCGLS with Deflation (BCGLSD), an initial guess X_0 is formed as

$$X_0 = X_{-1} + W\left(L^TL\right)^{-1}L^TR_{-1},$$

where X_{-1} is an arbitrary matrix and $L = AW$. Meanwhile, matrix orthogonalizations related to W are carried out to generate the subsequent search matrices.

```
BCGLS Algorithm with Deflation (BCGLSD)

Input: $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{m\times s}$, initial guess $X_{-1} \in \mathbb{R}^{n\times s}$, tolerance $tol \in \mathbb{R}$, maximum number
of iterations $maxit \in \mathbb{R}$, and deflation matrix $W \in \mathbb{R}^{n\times t}$.
Output: an approximate solution $X_{sol} \in \mathbb{R}^{n\times s}$.

$L = AW$
$R_{-1} = B - AX_{-1}$
$X_0 = X_{-1} + W\left(L^TL\right)^{-1}L^TR_{-1}$
$R_0 = B - AX_0$
$S_0 = A^TR_0$
$\widetilde{P}_0 = orth(S_0 - W\left(L^TL\right)^{-1}L^TAS_0)$
for $i = 0, \cdots, maxit$
    $\widetilde{Q}_i = A\widetilde{P}_i$
    $\widetilde{\alpha}_i = (\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^TR_i$
    $X_{i+1} = X_i + \widetilde{P}_i\widetilde{\alpha}_i$
    $R_{i+1} = R_i - \widetilde{Q}_i\widetilde{\alpha}_i$
    if converged within $tol$, then stop.
    $S_{i+1} = A^TR_{i+1}$
    $\widetilde{\beta}_i = -(\widetilde{Q}_i^T\widetilde{Q}_i)^{-1}\widetilde{Q}_i^TAS_{i+1}$
    $\widetilde{P}_{i+1} = orth(S_{i+1} + \widetilde{P}_i\widetilde{\beta}_i - W\left(L^TL\right)^{-1}L^TAS_{i+1})$
end
$X_{sol} = X_{i+1}$
```

Theorem 7 shows that the residual matrices $R_i (i \geq 0)$ and the search matrices $\widetilde{P}_i (i \geq 0)$ are constructed A-orthogonal and A^TA-orthogonal to deflation matrix W in BCGLSD, respectively.

Theorem 7. *Given a deflation matrix W, the following two orthogonality relations hold in BCGLSD,*

(i) $W^TA^TA\widetilde{P}_i = 0, \ (i \geq 0)$;
(ii) $W^TA^TR_i = 0, \ (i \geq 0)$.

Proof. (i) For any $k \geq 0$, since $\widetilde{P}_{k+1}$ is an orthogonal basis of the space spanned by the columns of $S_{k+1} + \widetilde{P}_k\widetilde{\beta}_k - W(L^TL)^{-1}L^TAS_{k+1}$, there exists an $s \times r_k$ matrix δ such that

$$\widetilde{P}_{k+1} = (S_{k+1} + \widetilde{P}_k\widetilde{\beta}_k - W(L^TL)^{-1}L^TAS_{k+1})\delta.$$

Then, we have

$$\begin{aligned} W^TA^TA\widetilde{P}_{k+1} &= W^TA^TA(S_{k+1} + \widetilde{P}_k\widetilde{\beta}_k - W(L^TL)^{-1}L^TAS_{k+1})\delta \\ &= W^TA^TA\widetilde{P}_k\widetilde{\beta}_k\delta. \end{aligned}$$

Clearly, because $W^TA^TA\widetilde{P}_0 = 0$, subsequently, $W^TA^TA\widetilde{P}_i = 0$ for all $i \geq 0$.

(ii) Since $R_{k+1} = R_k - \widetilde{Q}_k\widetilde{\alpha}_k$ and (i), we have

$$\begin{aligned} W^TA^TR_{k+1} &= W^TA^TR_k - W^TA^T\widetilde{Q}_k\widetilde{\alpha}_k \\ &= W^TA^TR_k. \end{aligned}$$

As $X_0 = X_{-1} + W(L^TL)^{-1}L^TR_{-1}$ and $R_0 = (I - AW(L^TL)^{-1}L^T)R_{-1}$, it follows that

$$W^TA^TR_0 = W^TA^T(I - AW(L^TL)^{-1}L^T)R_{-1} = 0.$$

As a deduction, we can get $W^TA^TR_i = 0$ for all $i \geq 0$. □

According to Theorem 7, in the subsequent BCGLSD iterations, the block Krylov subspace

$$\text{block-span}\left\{A^TR_0, \left(A^TA\right)A^TR_0, \ldots, \left(A^TA\right)^iA^TR_0, \ldots\right\},$$

is constructed to be orthogonal to the subspace spanned by W. Let $H = I - W(L^TL)^{-1}L^TA$ be the orthogonal projection onto the orthogonal complement of W, BCGLSD is in fact equivalent to BCGLS starting with A^TR_0 on a system with a transformed coefficient matrix H^TA^TAH.

The ideal deflation matrix W is composed of the exact extreme eigenvectors of A^TA. Assuming that the columns in W contain t eigenvectors of matrix A^TA corresponding to the t smallest eigenvalues, the impacts from these eigenvectors of matrix A^TA can be removed from matrix H^TA^TAH at the beginning, and thus BCGLSD has potentially faster convergence in a deflated system with a smaller condition number $\kappa'' = \lambda_n/\lambda_{s+t}$, where λ_n and λ_{s+t} are the nth and $(s+t)$th eigenvalues of A^TA, respectively. In practice, fast approximations to the eigenvectors corresponding to the extreme eigenvalues are often obtained from a separate Lanczos [33,23] or Arnoldi process [37,26].

6. Numerical results

We present three numerical examples to illustrate the capability of various forms of BCGLS in handling (near) rank deficiency and accelerating convergence using deflation. The sparse matrices used in these examples are obtained from the UFL sparse matrix collection [41].

6.1. Handling rank deficiency

We compute the least squares solutions of a linear system with coefficient matrix "illc1850" to demonstrate the robustness of BFBCGLS in case of rank deficiency. "illc1850" is a 1850×712 rectangular matrix with 8636 nonzero elements arisen from a least squares problem in surveying. A right-hand side block matrix B containing 100 column vectors with full column rank are generated randomly. A system is considered converged if the relative residual error of each solution with respect to its corresponding right-hand side is within the tolerance of 10^{-7}.

We start BFBCGLS with a zero initial solution block. Fig. 1 shows the number of columns in search matrix P_i after the rank-revealing operations (upper), the condition number of $Q_i^TQ_i$ (middle), and the maximum and minimum relative residual errors among all solution columns in X_i (lower) along BFBCGLS iterations. One can find that rank deficiency (from 100 down to 88) starts to occur at the 6th iteration. After all, BFBCGLS is able to continue to explore the Krylov subspaces with reduced search space without suffering breakdown, which leads to further residual error reduction in all systems as shown in Fig. 1 (lower).

Fig. 2 compares the solution precision measured by the maximum residual norm among columns in X_i with respect to different linear dependency threshold parameter τ values. It is interesting to note that, when a large τ value is used, only low precision solutions are obtained in BCGLS. This is due to the fact that, when a large τ value is reached, some solutions or linear combinations of solutions are considered converged and the search space is reduced without further improving these solutions. More importantly, a large τ value slows down convergence because of early reduction of search space. On the other hand, a τ value close to floating-point precision (10^{-16}) does not necessarily lead to more precise solutions due to low-quality search spaces where the Galerkin conditions are not fully satisfied any more. Our results indicate that the appropriate τ value should be in the range of 10^{-12}–10^{-14} for BCGLS using double precision floating point operations.

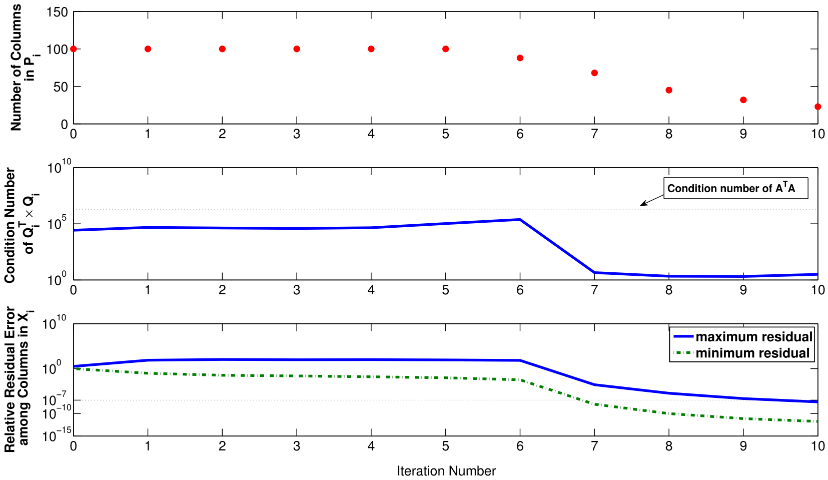

Fig. 1. Number of Columns in P_i (upper), condition number of $Q_i^T Q_i$(middle), and maximum and minimum relative residual norms of columns in X_i (lower) for a block linear system with 100 right hand sides using "illc1850" as the coefficient matrix along BFBCGLS iterations.

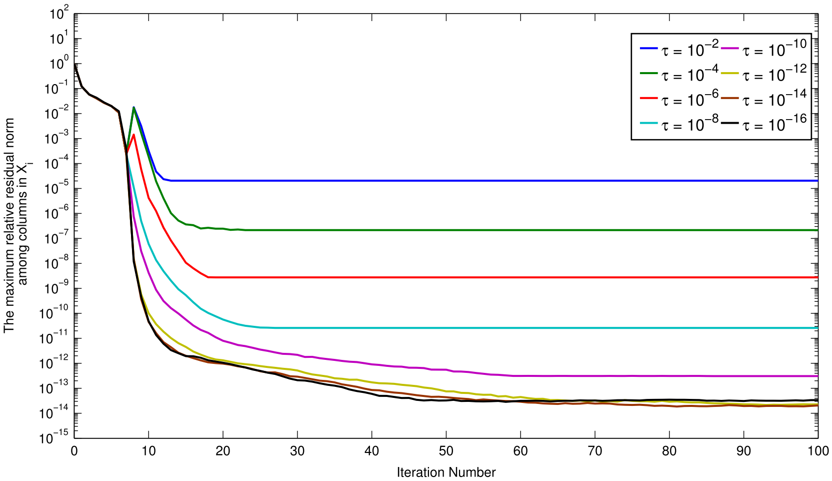

Fig. 2. Solution precisions obtained using different linear dependency threshold parameter τ values.

6.2. Convergence accelerations using deflation

We compare the convergence of CGLS, BCGLS, and BCGLSD on a least squares problem with coefficient matrix "wang4" from semiconductor device problem. "wang4" is a 26,068 × 26,068 unsymmetric matrix with 177,196 nonzero elements. Assuming that we are only interested in the solution to a single right-hand side. To accommodate with the block form in BCGLS and BCGLSD, we expand the single right-hand side to a block form with 100 right-hand sides by supplying 99 Gaussian random vectors to the right hand side. The deflation matrix W contains 50 approximate eigenvectors of matrix "wang4" estimated by an inverse randomized Singular Value Decomposition (SVD) algorithm [42–44].

Fig. 3 displays the numerical results of CGLS, BCGLS, and BCGLSD. One can clearly observe that by expanding the linear system from a single right-hand side to a block form with multiple right-hand sides, BCGLS (1834 steps) requires less iteration steps to converge to 10^{-7} relative residual error than CGLS (59,765 steps). Even though overall BCGLS involves more computational operations measured by the total number of matrix–vector multiplications than those of CGLS, it is important to note that BCGLS is a communication-efficient algorithm that can significantly reduce the number of passes over matrix A, the main computational bottleneck if passing over all elements in A is extremely costly. More importantly, when an approximate deflation matrix is applied, convergence can be significantly accelerated, where the number of iterations to reach convergence is further reduced down to 935 steps.

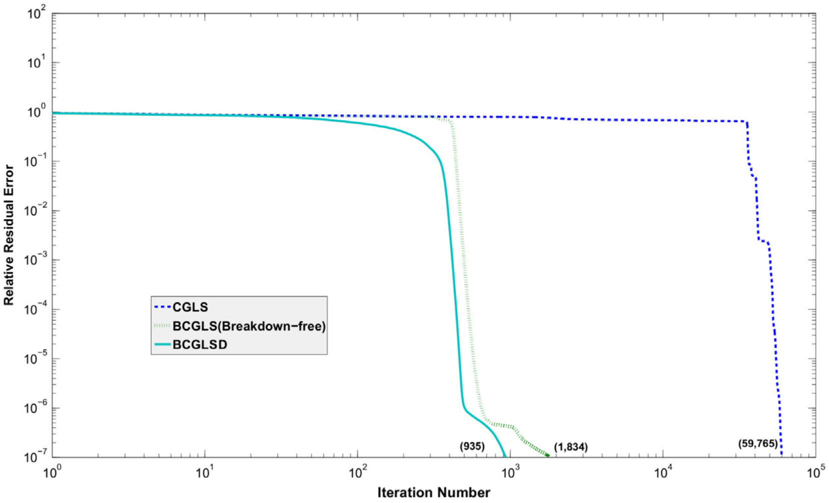

Fig. 3. Comparison of convergence in CGLS, BCGLS, and BCGLSD on a least squares problem using “wang4” as the coefficient matrix.

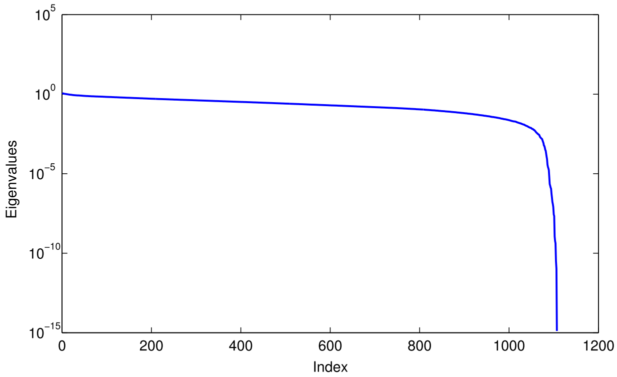

Fig. 4. Distribution of the eigenvalues in A^TA (“gre_1107”).

6.3. *Handling ill-conditioned coefficient matrices using deflation*

We use a linear system with “gre_1107”, a 1107 × 1107 unsymmetric matrix with 5664 nonzero elements, as the coefficient matrix to study the behavior of BCGLS in ill-conditioned least squares problems. Fig. 4 shows the distribution of the eigenvalues in A^TA. One can find that the 40 extremely small eigenvalues lead to a large condition number in A^TA. The condition number of $Q_i^TQ_i$ is bounded by that of A^TA. As shown in Fig. 5, when the condition number of $Q_i^TQ_i$ is small during BCGLS iterations before step 11, orthogonality is well preserved. However, at iteration step 11, the large condition number of $Q_{11}^TQ_{11}$ causes subsequent loss of orthogonality, as shown in the colormap of A^TA-orthogonality among the first 31 search matrices, where the colors in Q_i and Q_j intersection correspond to the base-10 logarithms of the absolute values of the elements in $Q_i^TQ_j = P_i^TA^TAP_j$. Consequently, BCGLS converges slowly and does not reach desired precision of 10^{-7} even after 10,000 iterations. An appropriate deflation matrix can address this issue and accelerate the convergence of BCGLS. Here we use a deflation matrix consisting of 40 approximate eigenvectors corresponding to the 40 extreme eigenvalues obtained from a separate Lanczos process. When the deflation matrix is applied, the condition number of $Q_i^TQ_i$ remains relatively small and orthogonality is mostly preserved during BCGLSD iterations, as shown in Fig. 6. As a result, BCGLSD converges at iteration step 11.

7. Conclusions and future research directions

In this paper, we extend the CGLS method to a block form to evaluate solutions for least squares problems with multiple right-hand sides. To address the potential breakdown problem due to rank deficiency, we derive new parameter matrices to generate search matrices with variable block size. Finally, a deflated form of BCGLS is designed to handle the

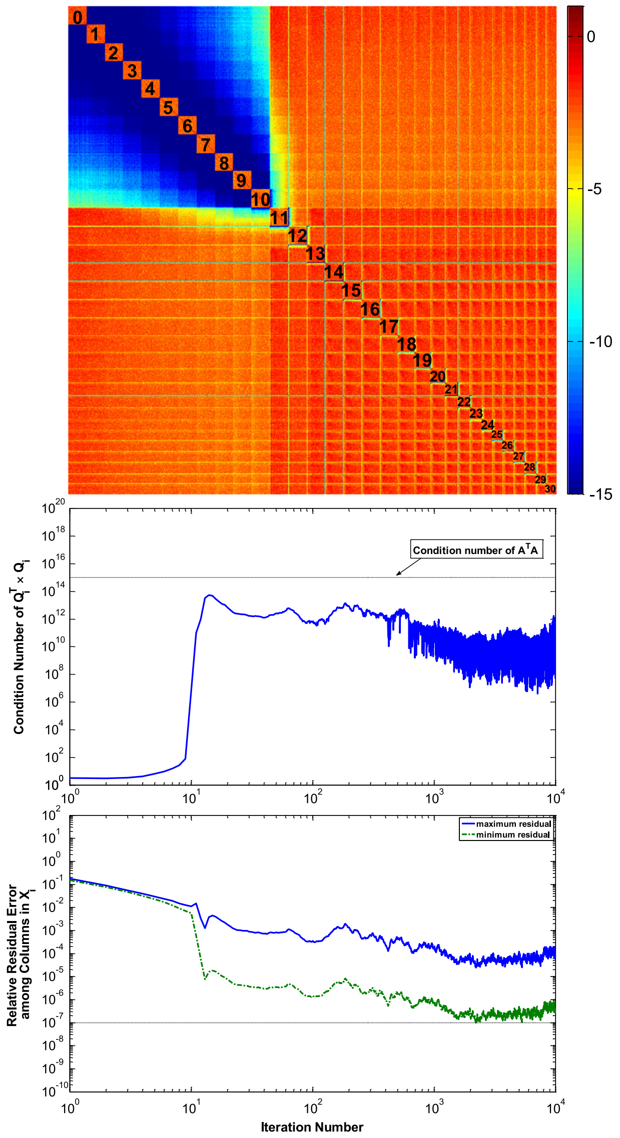

Fig. 5. Colormap of A^TA-orthogonality between Search Matrices in the first 31 iterations (upper), condition number of $Q_i^TQ_i$ (middle), and maximum and minimum relative residual norms of columns in X_i (lower) for a block linear system with 100 right hand sides using "gre_1107" as the coefficient matrix along BCGLS iterations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

extreme eigenvalues to accelerate convergence. The corresponding numerical stability and computational efficiency are demonstrated in numerical examples.

In addition to BCGLS which is based on CGLS, block LSQR [45], which extends LSQR by block bi-diagonalization [11,12], can also be used to solve least squares problems with multiple right-hand sides. Björck [16,46] showed that LSQR is likely to yield less iterations than CGLS to reach convergence but at the cost of more storage and computation per iteration. However, the block LSQR described in [45] suffers from breakdown when rank deficiency occurs. The variable block size method and deflation scheme provided in this paper may also be applied to block LSQR, as well as the other block Krylov subspace methods that employ short recurrences to update search matrices [47], to address the breakdown problem and accelerate convergence. After all, these will become our future research directions.

Acknowledgments

We would like to thank Martin H. Gutknecht and other reviewers for their valuable suggestions that helped to improve the contents of the manuscript. Yaohang Li acknowledges support from National Science Foundation Grant Number 1066471. Hao Ji acknowledges support from ODU Modeling and Simulation Fellowship.

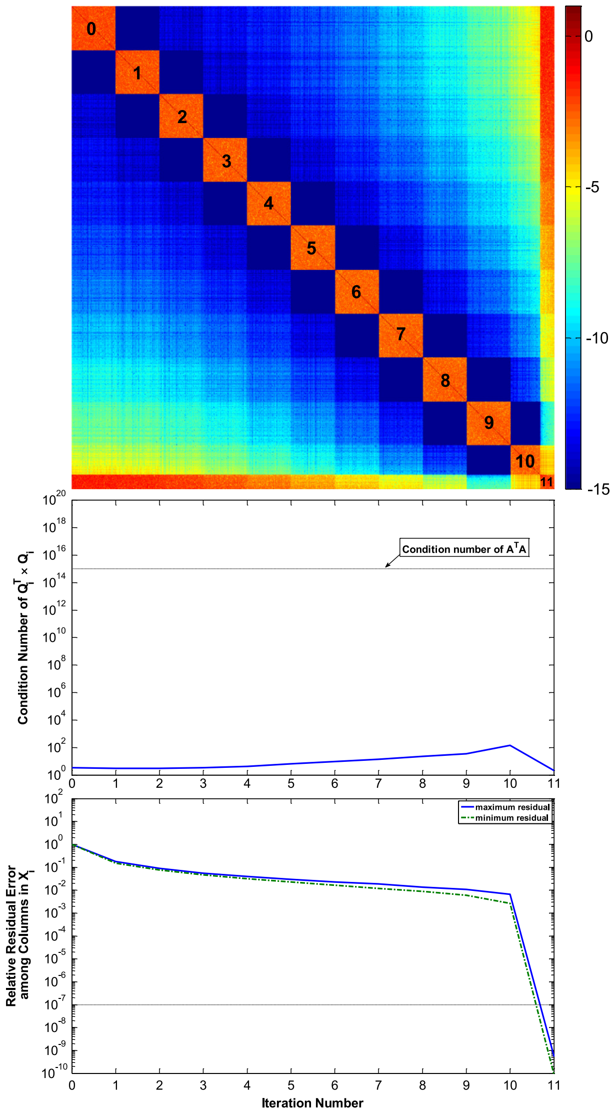

Fig. 6. Colormap of A^TA-orthogonality between Search Matrices in the first 12 iterations (upper), condition number of $Q_i^TQ_i$ (middle), and maximum and minimum relative residual norms of columns in X_i (lower) for a block linear system with 100 right hand sides using "gre_1107" as the coefficient matrix along BCGLSD iterations, where the deflation matrix consists of 40 approximated extreme eigenvectors. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Appendix

Corollary A.1. *Assuming that all previous search spaces $\mathcal{P}_j (j \leq i)$ are A^TA-orthogonal, then $R_{i+1}^TAA^TR_j = 0$, for $j < i + 1$.*

Proof. As $\widetilde{P}_i \in \mathbb{R}^{n\times r_i}$ is an orthonormal basis of search space $\mathcal{P}_i$, the range space of $A^TR_i + \widetilde{P}_{i-1}\widetilde{\beta}_{i-1}$, we have

$$A^TR_i + \widetilde{P}_{i-1}\widetilde{\beta}_{i-1} = \widetilde{P}_i\delta,$$

where δ is an $r_i \times s$ matrix of rank r_i. Then, left multiplying both sides by R_{i+1}^TA,

$$R_{i+1}^TAA^TR_i + R_{i+1}^TA\widetilde{P}_{i-1}\widetilde{\beta}_{i-1} = R_{i+1}^TA\widetilde{P}_i\delta.$$

Based on Theorem 3, we have $\widetilde{P}_j^TA^TR_{i+1} = 0$ for all $j < i + 1$. Under the assumption that all previous search spaces $\mathcal{P}_j (j \leq i)$ are A^TA-orthogonal, we can get $R_{i+1}^TAA^TR_i = 0$. By induction, we can conclude that $R_{i+1}^TAA^TR_j = 0$ for all $j < i + 1$. □

Corollary A.2. $R_{i+1}^TAA^TA\widetilde{P}_j = 0$, *for $j < i$.*

Proof. Since $R_{j+1} = R_j - A\widetilde{P}_j\widetilde{\alpha}_j$, left multiplying the equation by $R_{i+1}^T AA^T$ on both sides and we have

$$R_{i+1}^T AA^T R_{j+1} = R_{i+1}^T AA^T R_j - R_{i+1}^T AA^T A\widetilde{P}_j\widetilde{\alpha}_j.$$

When $j < i$, according to Corollary A.1, we can get $R_{i+1}^T AA^T R_j = 0$ and $R_{i+1}^T AA^T R_{j+1} = 0$. Therefore, $R_{i+1}^T AA^T A\widetilde{P}_j\widetilde{\alpha}_j = 0$ is derived for all $j < i$.

As $\widetilde{\alpha}_j = (\widetilde{Q}_j^T\widetilde{Q}_j)^{-1}\widetilde{Q}_j^T R_j$ where $\widetilde{Q}_j = A\widetilde{P}_j$, we have

$$R_{i+1}^T AA^T A\widetilde{P}_j(\widetilde{P}_j^T A^T A\widetilde{P}_j)^{-1}\widetilde{P}_j^T A^T R_j = 0.$$

Since $\widetilde{P}_j^T A^T A\widetilde{P}_j$ is an $r_j \times r_j$ matrix with full rank, $\widetilde{P}_j^T A^T R_j \in \mathbb{R}^{r_j \times s}$ is a matrix of rank r_j, and $r_j \leq s$, we finally obtain $R_{i+1}^T AA^T A\widetilde{P}_j = 0$, for all $j < i$. □

References

[1] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, John Wiley & Sons, Inc., New York, NY, USA, 2001.

[2] D.P. O'Leary, Parallel implementation of the block conjugate gradient algorithm, Parallel Comput. 5 (1) (1987) 127–139.

[3] H. Ji, M. Sosonkina, Y. Li, An implementation of block conjugate gradient algorithm on CPU–GPU processors, in: Hardware-Software Co-Design for High Performance Computing, Co-HPC, 2014, pp. 72–77.

[4] G.W. Stewart, Block Gram–Schmidt orthogonalizatio, SIAM J. Sci. Comput. 31 (1) (2008) 761–775.

[5] J.J. Dongarra, J.D. Cruz, S. Hammerling, I.S. Duff, Algorithm 679: A set of level 3 basic linear algebra subprograms: Model implementation and test programs, ACM Trans. Math. Software 16 (1) (1990) 18–28.

[6] K. Gallivan, W. Jalby, U. Meier, The use of BLAS3 in linear algebra on a parallel processor with a hierarchical memory, SIAM J. Sci. Stat. Comput. 8 (6) (1987) 1079–1084.

[7] J.W. Demmel, N.J. Higham, Stability of block algorithms with fast level–3 BLAS, ACM Trans. Math. Software 18 (3) (1992) 274–291.

[8] E.J. Craig, The n-step iteration procedures, J. Math. Phys. 34 (1) (1955) 64–73.

[9] M.R. Hestenes, E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Natl. Bur. Stand. 49 (1952) 409–436.

[10] G.H. Golub, C.F. Van Loan, Matrix Computations, fourth ed., Johns Hopkins University Press, Baltimore, MD, 2013.

[11] C.C. Paige, M.A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, ACM Trans. Math. Software 8 (1) (1982) 43–71.

[12] C.C. Paige, M.A. Saunders, Algorithm 583, LSQR: Sparse linear equations and least squares problems, ACM Trans. Math. Software 8 (2) (1982) 195–209.

[13] D.C.-L. Fong, M. Saunders, LSMR: An iterative algorithm for sparse least–squares problems, SIAM J. Sci. Comput. 33 (5) (2011) 2950–2971.

[14] C.C. Paige, M.A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12 (4) (1975) 617–629.

[15] R. Fletcher, Conjugate gradient methods for indefinite systems, in: Numerical Analysis (Proc 6th Biennial Dundee Conf., Univ. Dundee, Dundee, 1975), in: Lecture Notes in Mathematics, vol. 506, Springer, 1976, pp. 73–89.

[16] Å. Björck, T. Elfving, Z. Strakos, Stability of conjugate gradient and Lanczos methods for linear least squares problems, SIAM J. Matrix Anal. Appl. 19 (3) (1998) 720–736.

[17] T. Elfving, On the Conjugate Gradient method for solving linear least squares problems, Report LiTH-MAT-R-78-3, Department of Math., Linköping University, Linköping, Sweden, 1978.

[18] M.H. Gutknecht, Block Krylov space methods for linear systems with multiple right-hand sides: An introduction, in: Modern Mathematical Models, Methods and Algorithms for Real World Systems, Anamaya Publishers, New Delhi, India, 2006, pp. 420–447.

[19] H. Ji, Y. Li, BIT Numer. Math. (2016) http://dx.doi.org/10.1007/s10543-016-0631-z.

[20] D.P. O'Leary, The block conjugate gradient algorithm and related methods, Linear Algebra Appl. 29 (1980) 293–322.

[21] M. Robbé, M. Sadkane, Exact and inexact breakdowns in the block GMRES method, Linear Algebra Appl. 419 (1) (2006) 265–285.

[22] R.W. Freund, M. Malhotra, A block QMR algorithm for non-Hermitian linear systems with multiple right-hand sides, Linear Algebra Appl. 254 (1) (1997) 119–157.

[23] J. Chen, A deflated version of the block conjugate gradient algorithm with an application to Gaussian process maximum likelihood estimation, Preprint ANL/MCS-P1927-0811, Argonne National Laboratory, Argonne, IL, 2011.

[24] V. Simoncini, A stabilized QMR version of block BICG, SIAM J. Matrix Anal. Appl. 18 (2) (1997) 419–434.

[25] J. Langou, Iterative methods for solving linear systems with multiple right-hand sides (Ph.D. thesis), Department of Mathematics, CERFACS, Toulouse, France, 2003.

[26] E. Agullo, L. Giraud, Y.F. Jing, Block GMRES method with inexact breakdowns and deflated restarting, SIAM J. Matrix Anal. Appl. 35 (4) (2014) 1625–1651.

[27] A.A. Nikishin, A.Y. Yeremin, Variable block CG algorithms for solving large sparse symmetric positive definite linear systems on parallel computers, i: General iterative scheme, SIAM J. Matrix Anal. Appl. 16 (4) (1995) 1135–1153.

[28] M.H. Gutknecht, T. Schmelzer, The block grade of a block Krylov space, Linear Algebra Appl. 430 (1) (2009) 174–185.

[29] T. Schmelzer, Block Krylov methods for Hermitian linear systems (Diploma thesis), Department of Mathematics, University of Kaiserslautern, Germany, 2004.

[30] M.H. Gutknecht, Block Krylov space solvers: A survey, 2005. http://www.sam.math.ethz.ch/~mhg/talks/bkss.pdf (last accessed 20.12.15).

[31] A. Gaul, M.H. Gutknecht, J. Liesen, R. Nabben, A framework for deflated and augmented Krylov subspace methods, SIAM J. Matrix Anal. Appl. 34 (2) (2013) 495–518.

[32] J. Erhel, F. Guyomarc'h, An augmented conjugate gradient method for solving consecutive symmetric positive definite linear systems, SIAM J. Matrix Anal. Appl. 21 (4) (2000) 1279–1299.

[33] Y. Saad, M. Yeung, J. Erhel, F. Guyomarc'h, A deflated version of the conjugate gradient algorithm, SIAM J. Sci. Comput. 21 (5) (2000) 1909–1926.

[34] R.A. Nicolaides, Deflation of conjugate gradients with applications to boundary value problems, SIAM J. Numer. Anal. 24 (2) (1987) 355–365.

[35] Z. Dostál, Conjugate gradient method with preconditioning by projector, Int. J. Comput. Math. 23 (3–4) (1988) 315–323.

[36] J. Erhel, K. Burrage, B. Pohl, Restarted GMRES preconditioned by deflation, J. Comput. Appl. Math. 69 (2) (1996) 303–318.

[37] R.B. Morgan, A restarted GMRES method augmented with eigenvectors, SIAM J. Matrix Anal. Appl. 16 (4) (1995) 1154–1171.

[38] E. De Sturler, Nested Krylov methods based on GCR, J. Comput. Appl. Math. 67 (1) (1996) 15–41.

[39] S.A. Kharchenko, A.Y. Yeremin, Eigenvalue translation based preconditioners for the GMRES(k) method, Numer. Linear Algebra Appl. 2 (1) (1995) 51–77.

[40] M.H. Gutknecht, Deflated and augmented Krylov subspace methods: A framework for deflated BiCG and related solvers, SIAM J. Matrix Anal. Appl. 35 (4) (2014) 1444–1466.

[41] T.A. Davis, Y. Hu, The university of Florida sparse matrix collection, ACM Trans. Math. Software 38 (2011) 1–25.

[42] N. Halko, P.G. Martinnson, J.A. Tropp, Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions, SIAM Rev. 53 (2) (2011) 217–288.

[43] H. Ji, Y. Li, Monte Carlo methods and their applications in big data analysis, in: Mathematical Problems in Data Science – Theoretical and Practical Methods, Springer International Publishing, 2015, pp. 125–139.

[44] H. Ji, W. Yu, Y. Li, A rank revealing randomized singular value decomposition (r3svd) algorithm for low-rank matrix approximations, Comput. Res. Repository (2016) 1–10. arXiv:1605.08134.

[45] S. Karimi, F. Toutounian, The block least squares method for solving nonsymmetric linear systems with multiple right-hand sides, Appl. Math. Comput. 177 (2) (2006) 852–862.

[46] Å. Björck, Use of conjugate gradients for solving linear least squares problems, in: I.S. Duff (Ed.), Conjugate-Gradient Methods and Similar Techniques, Rep. AERE R-9636, Computer Science and Systems Division, AERE Harwell, England, 1979, pp. 48–71.

[47] M.H. Gutknecht, A general framework for recursions for Krylov space solvers, 2005. https://www.sam.math.ethz.ch/sam_reports/reports_final/reports2005/2005-09.pdf (last accessed 20.12.15).