A quasi-minimal residual variant of the BiCORSTAB method for nonsymmetric linear systems

Sun, Dong-Lin; Jing, Yan-Fei; Huang, Ting-Zhu; Carpentieri, Bruno

Published in:
Computers & Mathematics with Applications

DOI:
10.1016/j.camwa.2014.04.014

Computers and Mathematics with Applications 67 (2014) 1743–1755

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

A quasi-minimal residual variant of the BiCORSTAB method for nonsymmetric linear systems

Dong-Lin Sun [a], Yan-Fei Jing [a,*], Ting-Zhu Huang [a], Bruno Carpentieri [b]

[a] School of Mathematical Sciences/Institute of Computational Science, University of Electronic Science and Technology of China, Chengdu, Sichuan, 611731, PR China

[b] Institute of Mathematics and Computing Science, University of Groningen, Nijenborgh 9, PO Box 407, 9700 AK Groningen, Netherlands

ARTICLE INFO

Article history:
Received 17 December 2013
Received in revised form 6 April 2014
Accepted 21 April 2014
Available online 5 May 2014

Keywords:
Quasi-minimal residual variant
Nonsymmetric linear systems
BiCORSTAB
QMRCGSTAB
QMRCORSTAB

ABSTRACT

The Biconjugate *A*-orthogonal residual stabilized method named as BiCORSTAB was proposed by Jing et al. (2009), where the numerical experiments therein demonstrate that the BiCORSTAB method converges more smoothly than the Bi-Conjugate Gradient stabilized (BiCGSTAB) method in some circumstances. In order to further stabilize the convergence behavior and hopefully to accelerate the convergence speed of the BiCORSTAB algorithm when it has erratic convergence curves, a quasi-minimal residual variant of the BiCORSTAB algorithm, named as QMRCORSTAB, will be developed and investigated for solving nonsymmetric systems of linear equations borrowing the same further-smooth-effect idea for the QMRCGSTAB method. Numerical experiments on typical sets of both sparse and dense matrices will show that the proposed QMRCORSTAB method shares attractive smoother effect over its basic parent and also outperforms its counterpart.

1. Introduction

Krylov subspace methods with preconditioning techniques are widely used for iterative solution of large sparse linear system

$$Ax = b, \tag{1}$$

where A is a nonsymmetric $n \times n$ matrix, and b is an n-vector. One of the most popular iterative methods for the above system is the generalized minimum residual (GMRES) method proposed by Saad and Schultz [1,2]. The method tries to find appropriate approximate solution which minimizes the residual over the m-dimensional Krylov subspace $x_0 + \mathcal{K}_m(A, r_0)$ generated by A and an initial residual $r_0 = b - Ax_0$ with an initial guess x_0. However, it turns to impractical when m becomes large because of the growth of memory and computational requirements as m increases. To limit the cost of GMRES, it is often restarted after each cycle of m iterations, which produces the restarted GMRES method denoted by GMRES(m) [1]. Restarting GMRES deteriorates convergence significantly. Several techniques have been proposed in the past few years that attempt to tackle this kind of problem; see, to name a few, [3–6] for recent work on deflation and augmentation accelerating techniques.

Another well-known popular Krylov subspace method for solving this system is the Biconjugate Gradient (BiCG) method proposed by Fletcher [7]. However, the BiCG method shows irregular convergence behavior. Many ingenious methods have

* Corresponding author. Tel.: +86 15828606538.
E-mail addresses: donglin1220@126.com (D.-L. Sun), yanfeijing@uestc.edu.cn, 00jyfvictory@163.com (Y.-F. Jing), tingzhuhuang@126.com (T.-Z. Huang), bcarpentieri@gmail.com (B. Carpentieri).

http://dx.doi.org/10.1016/j.camwa.2014.04.014
0898-1221/

been devoted to improving the performance of the BiCG method, such as the Conjugate Gradient Squared (CGS) method developed by Sonneveld [8], the van der Vorst's Biconjugate Gradient stabilized (BiCGSTAB) method [9], the BiCGSTAB2 method by Gutknecht [10], the BiCGSTAB(l) method by Sleijpen and Fokkema [11], and the Generalized Product-type method based on BiCG (GPBiCG) method introduced by Zhang [12].

Recently, a new family, named Biconjugate A-orthogonal Residual (BiCOR) family, of efficient short-recurrence methods for solving the system of linear equations (1), was proposed by Jing, Huang, Zhang, et al. [13] and Carpentieri, Jing and Huang [14]. The BiCOR family of solvers shows their competitiveness with other popular Krylov solvers in use today in different scientific and engineering applications [15,16], especially when memory is a concern. The first variant of the BiCOR family of iterative methods—the BiCOR method, gives much smoother convergence behavior and often converges faster than the BiCG method. Based on certain product-type variants of the BiCG method, Jing, Huang, Zhang, et al. [13] presented the Conjugate A-orthogonal Residual Squared (CORS) method and the Biconjugate A-orthogonal Residual stabilized (BiCORSTAB) method. The same residual polynomial of the BiCOR method is still used in the CORS method, i.e., the BiCOR residual polynomial is squared in the CORS method. Therefore, the convergence of the CORS method may be more erratic than that of the BiCOR method when the BiCOR method has irregular convergence behavior. In order to overcome this kind of convergence problem, the BiCORSTAB method has been established sharing the same strategies with the BiCGSTAB method. While the BiCORSTAB method works well in many cases, it still has quite erratic oscillation in some difficult problems; see e.g., the experiments reported in [13] on the HB/young1c matrix problem from aero research applications in the Harwell/Boeing collection [17]. Based on the idea of BiCORSTAB, various generalized methods have been proposed such as GPBiCOR [18] and BiCORSTAB2 [19].

For the Lanczos-type product methods are often faced with apparently irregular convergence behaviors, along with the associated problems of round-off errors, in practice, a method with smoother convergence behavior is more desirable [20]. In such cases, Freund proposed the QMR method [21], and its transpose-free variant (TFQMR) [22] by quasi-minimizing the residual norms generated by the CGS method. Similarly, an approach called QMRCGSTAB to smooth the highly erratic convergence behavior of the BiCGSTAB method was proposed by Chan, Gallopoulos, Simoncini et al. [23]; for related references, we refer to [24,25,20,26–30]. In this paper, combining the best of the BiCORSTAB method and the QMR-type strategies [21,22], a quasi-minimal residual variant of the BiCORSTAB method, named QMRCORSTAB will be developed to overcome the irregular convergence behavior of the BiCORSTAB method having in mind the idea for the development of the QMRCGSTAB method [23]. Moreover, explicit formulations of the approximate residual vectors at each iteration will be provided for both the QMRCORSTAB and QMRCGSTAB methods, whereas the true residual vectors are used for the implementation of the QMRCGSTAB method in [23]. It will be numerically demonstrated that the proposed new variant—QMRCORSTAB outperforms its counterpart—QMRCGSTAB as well as its basic parent—BiCORSTAB in certain circumstances. It is remarked that almost simultaneously Abe and Sleijpen [31] independently developed the hybrid BiCR variants for solving nonsymmetric linear systems by replacing the BiCG part in the residual polynomial of the hybrid BiCG methods with BiCR [32], but the comparison of the BiCOR family of solvers [13] and the hybrid BiCR variants [31] is not the emphasis of this paper.

The remainder of the paper is organized as follows. A brief description of the BiCORSTAB method is recalled in Section 2 and its quasi-minimal residual variant with explicit formulations of the approximate residual vectors provided each iteration is presented in Section 3. Numerical experiments on typical sets of both sparse and dense matrices are reported to show the smoother effect of the QMRCORSTAB method than the BiCORSTAB method as well as the associated efficiency obtained over the QMRCGSTAB method in Section 4. Finally, some perspectives are made in Section 5.

Throughout this paper, we denote by A^T and A^H the transpose and conjugate transpose of A, respectively. $\langle x, y\rangle = x^H y$ denotes the Euclidean inner product with $\|x\|$ denoting the Euclidean norm $\|x\| = \sqrt{x^H x}$. For $p \in R^+$, $[p]$ is the integer part of p. The nested Krylov subspace of dimension k generated by A from v is of the form

$$\mathcal{K}_k(A, v) = \text{span}\left\{v, Av, A^2 v, \ldots, A^{k-1} v\right\}.$$

2. The BiCORSTAB algorithm

Given an initial guess x_0 to the complex nonsymmetric linear system (1), we consider an oblique projection method onto $\mathcal{K}_m(A, v_1)$ and orthogonal to $\mathcal{L}_m(A, w_1)$, taking $v_1 = \frac{r_0}{\|r_0\|_2}$ and w_1 is arbitrary, provided $\langle w_1, Av_1\rangle \neq 0$, which is often chosen to be equal to $\frac{Av_1}{\|Av_1\|_2^2}$. The biconjugate A-orthogonal residual (BiCOR) algorithm seeks an approximate solution x_m from the affine subspace $x_0 + \mathcal{K}_m(A, v_1)$ of dimension m by imposing the Petrov–Galerkin condition

$$b - Ax_m \perp \mathcal{L}_m,$$

where $\mathcal{L}_m = A^H \mathcal{K}_m(A^H, w_1)$. Exploiting the glorious idea of the BiCGSTAB [9] method, Jing, Huang, Zhang, et al. [13] developed the BiCORSTAB algorithm. The BiCORSTAB produces iterates whose residual vectors satisfy

$$r_i = \psi_i(A)\phi_i(A)r_0,$$

and direction vectors are defined as

$$p_i = \psi_i(A)\pi_i(A)r_0,$$

in which, ψ_i, ϕ_i and π_i are Lanczos-type polynomials of degree less than or equal to i satisfying $\phi_i(0) = 1$. Specifically, $\psi_i(t)$ is defined by the simple recurrence $\psi_{i+1}(t) = (1 - \omega_i t)\psi_i(t)$ in which the scalar ω_i is to be determined. The pseudocode for the left preconditioned BiCORSTAB algorithm [13] is shown as in Algorithm 1.

Algorithm 1 *Left preconditioned BiCORSTAB method.*

Compute $r_0 = b - Ax_0$ for some initial guess x_0.
Choose $r_0^* = P(A)r_0$ such that $\langle r_0^*, Ar_0\rangle \neq 0$, where $P(t)$ is a polynomial in t. (For example, $r_0^* = Ar_0$).
for $i = 1, 2, \dots$ **do**
    solve $Mz_{i-1} = r_{i-1}$
    $\hat{z} = Az_{i-1}$
    $\rho_{i-1} = \langle r_0^*, \hat{z}\rangle$
    if $\rho_{i-1} = 0$, **method fails**
    if $i = 1$ **then**
        $p_0 = r_0$
        solve $Mzp_0 = p_0$
        $q_0 = \hat{z}$
    else
        $\beta_{i-2} = (\rho_{i-1}/\rho_{i-2}) \times (\alpha_{i-2}/\omega_{i-2})$
        $zp_{i-1} = z_{i-1} + \beta_{i-2}(zp_{i-2} - \omega_{i-2}zq_{i-2})$
        $q_{i-1} = \hat{z} + \beta_{i-2}(q_{i-2} - \omega_{i-2}\widehat{zq}_{i-2})$
    end if
    solve $Mzq_{i-1} = q_{i-1}$
    $\hat{zq}_{i-1} = Azq_{i-1}$
    $\alpha_{i-1} = \rho_{i-1}/\langle r_0^*, \hat{zq}_{i-1}\rangle$
    $s = r_{i-1} - \alpha_{i-1}q_{i-1}$
    check norm of s; if small enough: set $x_i = x_{i-1} + \alpha_{i-1}zp_{i-1}$ and stop
    $zs = z_{i-1} - \alpha_{i-1}zq_{i-1}$
    $t = \hat{z} - \alpha_{i-1}\hat{zq}_{i-1}$
    $\omega_{i-1} = \langle t, s\rangle / \langle t, t\rangle$
    $x_i = x_{i-1} + \alpha_{i-1}zp_{i-1} + \omega_{i-1}zs$
    $r_i = s - \omega_{i-1}t$
    check convergence; continue if necessary
    for continuation it is necessary that $\omega_{i-1} \neq 0$
end for

3. The QMRCORSTAB algorithm

The algorithm to be proposed in this section is derived from the BiCORSTAB algorithm, inspired by the QMRCGSTAB algorithm [23] which combines virtues of BiCGSTAB and quasi-minimal principle. The vectors s_i and r_{i+1} generated by Algorithm 1 are as follows:

$$s_i = r_i - \alpha_i A p_i, \qquad r_{i+1} = s_i - \omega_i A s_i. \tag{2}$$

We set

$$y_m = \begin{cases} p_i, & m = 2i - 1, \quad \text{for } i = 1, \dots, [m + 1/2] \\ s_i, & m = 2i, \quad \text{for } i = 1, \dots, [m/2]. \end{cases} \tag{3}$$

Similarly, w_m and δ_m are defined as

$$w_m = \begin{cases} r_i, & m = 2i - 1, \quad \text{for } i = 1, \dots, [m + 1/2] \\ s_i, & m = 2i, \quad \text{for } i = 1, \dots, [m + 1/2], \end{cases} \tag{4}$$

$$\delta_m = \begin{cases} \alpha_i, & m = 2i - 1, \quad \text{for } i = 1, \dots, [m + 1/2] \\ \omega_i, & m = 2i, \quad \text{for } i = 1, \dots, [m/2]. \end{cases} \tag{5}$$

With these settings, Eq. (2) is translated into a single equation

$$Ay_k = (w_k - w_{k+1})/\delta_k, \quad k = 1, \dots, m. \tag{6}$$

Reformulate Eq. (6) into matrix form as

$$AY_k = W_{k+1}\overline{B}_k, \tag{7}$$

where $Y_k = [y_1, y_2, \dots, y_k]$, $W_{k+1} = (w_1, w_2, \dots, w_k, w_{k+1})$ and $\overline{B}_k$ is the $(k + 1) \times k$ bidiagonal matrix, i.e.,

$$\overline{B}_k = \begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \\ & & & -1 \end{bmatrix} \times \operatorname{diag}\left\{\frac{1}{\delta_1}, \frac{1}{\delta_2}, \dots, \frac{1}{\delta_k}\right\}.$$

One main relation that we should mention here is that the columns of Y_k and W_k will span the same subspace of $\mathcal{K}_k(A, r_0)$, where the basis of span $\{Y_k\}$ is generated by the BiCORSTAB method. Any vector x_k in $x_0 + \mathcal{K}_k(A, r_0)$ can be written as

$$x_k = x_0 + Y_k z_k, \quad \text{for } z_k \in C^k.$$

Hence, using Eq. (7) and $w_1 = r_0$, the residual can be written as

$$r_k = r_0 - AY_k z_k = W_{k+1}(e_1^{k+1} - \overline{B}_k z_k), \tag{8}$$

where $e_1^{k+1} = (1, 0, \ldots, 0)^T \in C^{k+1}$. In fact, the columns of W_{k+1} are not orthogonal to each other. We can multiply W_{k+1} from the right-hand side by a scaling matrix to make its each column have a 2-norm equal to unity. Let this scaling matrix be the inverse of the matrix $\Sigma_{k+1} = \mathrm{diag}(\sigma_1, \sigma_2, \ldots, \sigma_{k+1})$ and set $\sigma_k = \|w_k\|$, then we can rewrite Eq. (8) as

$$r_k = W_{k+1}\Sigma_{k+1}^{-1}(\sigma_1 e_1^{k+1} - \overline{H}_{k+1} z_k), \tag{9}$$

where $\overline{H}_{k+1} = \Sigma_{k+1}\overline{B}_k$.

The ideal solution is to determine z_k by minimizing the 2-norm of the right-hand side of Eq. (9). We apply the strategy of the QMR method to Eq. (9), i.e., we solve the least-squares problem $\min_z \left\|\sigma_1 e_1^{k+1} - \overline{H}_{k+1} z\right\|$.

It is easy to verify that the BiCORSTAB iterates $\widetilde{x}_k$ satisfy the following form:

$$\widetilde{x}_k = x_0 + Y_k \widetilde{z}_k, \quad \text{with } \widetilde{z}_k = H_k^{-1}(\sigma_1 e_1^k) = (\delta_1, \delta_2, \ldots, \delta_k)^T, \tag{10}$$

defining H_k to be the $k \times k$ matrix obtained from $\overline{H}_{k+1}$ by deleting its last row. Then Eq. (10) can be rewritten as:

$$\widetilde{x}_k = \widetilde{x}_{k-1} + \delta_k y_k. \tag{11}$$

Since $\overline{H}_{k+1}$ is an upper Hessenberg matrix, QR decomposition with Givens rotations is the best way for us to choose. Moreover, exploiting Lemma 4.1 in [22], the QMRCORSTAB iterates satisfy the relations

$$x_k - x_{k-1} = c_k^2(\widetilde{x}_k - x_{k-1}), \tag{12}$$

$$\theta_k = \frac{w_{k+1}}{\tau_{k-1}}, \qquad c_k = \frac{1}{\sqrt{1+\theta_k^2}}, \qquad \tau_k = \tau_{k-1}\theta_k c_k. \tag{13}$$

Setting

$$d_k \equiv \frac{1}{\delta_k}(\widetilde{x}_k - x_{k-1}) = \frac{1}{c_k^2\delta_k}(x_k - x_{k-1}), \tag{14}$$

$$\eta_k \equiv c_k^2\delta_k, \tag{15}$$

the above expression for x_k becomes

$$x_k = x_{k-1} + \eta_k d_k. \tag{16}$$

From Eqs. (11)–(14), a recurrence relation from d_k can be extracted as

$$d_k = y_k + \frac{\theta_{k-1}^2\eta_{k-1}}{\delta_k} d_{k-1}. \tag{17}$$

In practice, stopping criterion for convergence check is usually based on the 2-norm $\|r_k\|_2$ of the residual vector r_k associated with x_k. However, the residual vectors r_k's are not explicitly shown in the QMRCORSTAB method as well in the QMRCGSTAB method [23]. The following inequality provides an upper bound on the residual norm

$$\|r_k\| \le \left\|W_{k+1}\Sigma_{k+1}^{-1}\right\| \ \left\|\sigma_1 e_1^{k+1} - \overline{H}_{k+1} z_k\right\| \le \sqrt{k+1}\,|\tau|\,.$$

However, in practical algorithmic implementations, we would prefer to choose the updated residual norms for the stopping criterion. Moreover, in order to reduce the number of matrix–vector multiplications in practice instead of computing the true residual vector $r_k = b - Ax_k$ at each iteration in [23], we can update the approximate residual vectors r_k explicitly as

$$r_k = r_{k-1} - \eta_k e_k, \quad \text{where } e_k = Ad_k \tag{18}$$

by left multiplying Eq. (17) by A on both sides, namely,

$$Ad_k = Ay_k + \frac{\theta_{k-1}^2\eta_{k-1}}{\delta_k} Ad_{k-1} \tag{19}$$

and together with Eqs. (6) and (16).

Algorithm 2 is a version of the left preconditioned QMRCGSTAB method with explicit approximate residual vectors updated per iterations, which is not available in the original QMRCGSTAB method in [23]. Combining Eqs. (3)–(5) and

Algorithm 2 *Left preconditioned QMRCGSTAB method.*

Compute $r_0 = b - Ax_0$ for some initial guess x_0.
Choose r_0^* such that $\langle r_0^*, r_0 \rangle \neq 0$.
Solve $Mz_0^{BG} = r_0^{BG}$.
Set $\theta_0 = \eta_0 = 0$; $\tau = \left\| z_0^{BG} \right\|$; $zd_0 = e_0 = \mathbf{0}$; $r_0^{BG} = r_0$.
for $i = 1, 2, \ldots$ **do**
  $\rho_{i-1} = \left\langle r_0^*, z_{i-1}^{BG} \right\rangle$
  if $\rho_{i-1} = 0$, **method fails**
  if $i = 1$ **then**
    $p_0 = r_0^{BG}$
    solve $Mzp_0 = p_0$
  else
    $\beta_{i-2} = \left(\rho_{i-1}/\rho_{i-2}\right) \times \left(\alpha_{i-2}/\omega_{i-2}\right)$
    $zp_{i-1} = z_{i-1}^{BG} + \beta_{i-2}\left(zp_{i-2} - \omega_{i-2} z\hat{v}_{i-2}\right)$
  end if
  $\hat{v}_{i-1} = Azp_{i-1}$
  solve $Mz\hat{v}_{i-1} = \hat{v}_{i-1}$
  $\alpha_{i-1} = \rho_{i-1} / \left\langle r_0^*, z\hat{v}_{i-1} \right\rangle$
  $s_{i-1} = r_{i-1}^{BG} - \alpha_{i-1}\hat{v}_{i-1}$
  $zs_{i-1} = z_{i-1}^{BG} - \alpha_{i-1} z\hat{v}_{i-1}$
  $\widetilde{\theta}_i = \left\| zs_{i-1} \right\| / \tau$; $c = 1/\sqrt{1 + \widetilde{\theta}_i^2}$; $\widetilde{\tau} = \tau \widetilde{\theta}_i c$
  $\widetilde{\eta}_i = c^2 \alpha_{i-1}$
  $\widetilde{zd}_i = zp_{i-1} + \frac{\theta_{i-1}^2 \eta_{i-1}}{\alpha_{i-1}} zd_{i-1}$
  $\widetilde{x}_i = x_{i-1} + \widetilde{\eta}_i \widetilde{zd}_i$
  $\widetilde{e}_i = \hat{v}_{i-1} + \frac{\theta_{i-1}^2 \eta_{i-1}}{\alpha_{i-1}} e_{i-1}$
  $\widetilde{r}_i = r_{i-1} - \widetilde{\eta}_i \widetilde{e}_i$
  check norm of $\widetilde{r}_i$; if small enough then stop
  $\hat{t} = Azs_{i-1}$
  $\omega_{i-1} = \left\langle \hat{t}, s_{i-1} \right\rangle / \left\langle \hat{t}, \hat{t} \right\rangle$
  solve $Mz\hat{t} = \hat{t}$
  $r_i^{BG} = s_{i-1} - \omega_{i-1}\hat{t}$
  $z_i^{BG} = zs_{i-1} - \omega_{i-1} z\hat{t}$
  $\theta_i = \left\| z_i^{BG} \right\| / \widetilde{\tau}$; $c = 1/\sqrt{1 + \theta_i^2}$; $\tau = \widetilde{\tau}\theta_i c$
  $\eta_i = c^2 \omega_{i-1}$
  $zd_i = zs_{i-1} + \frac{\widetilde{\theta}_i^2 \widetilde{\eta}_i}{\omega_{i-1}} \widetilde{zd}_i$
  $x_i = \widetilde{x}_i + \eta_i zd_i$
  $e_i = \hat{t} + \frac{\widetilde{\theta}_i^2 \widetilde{\eta}_i}{\omega_{i-1}} \widetilde{e}_i$
  $r_i = \widetilde{r}_i - \eta_i e_i$
  check convergence; continue if necessary
end for

Table 1
Computational cost per iteration for the un-preconditioned (preconditioned) BiCORSTAB, QMRCGSTAB and QMRCORSTAB methods.

Method	*MVPs*	*DOTs*	*AXPYs*	Preconditioner solve
BiCORSTAB	2	4	10 (11)	2
QMRCGSTAB	2	6	12 (14)	2
QMRCORSTAB	2	6	15 (17)	2

Eqs. (13), (16)–(19), it is straightforward to derive the QMRCORSTAB algorithm, whose pseudocode with left preconditioner is illustrated as in Algorithm 3. In Algorithm 2 and Algorithm 3, the r^{BG} and r^{BO} represent the residual vectors respectively generated by the BiCGSTAB and BiCORSTAB methods. Similar to the notation in Algorithm 1, a prefix z is added to preconditioned variables, and a hat symbol ˆ is used for matrix–vector products.

A comparison of the computational cost per iteration for the BiCORSTAB, QMRCGSTAB and QMRCORSTAB methods is given in Table 1, where *MVPs*, *DOTs* and *AXPYs* denote the number of matrix–vector products, the number of inner products and the number of operations of the form "(scalar) × (vector) + (vector)", respectively. It is noted from Table 1 that the QMRCORSTAB method needs more *DOTs* and *AXPYs* than both the QMRCGSTAB and BiCORSTAB methods, but the numerical experiments in the coming section will verify the advantages of the QMRCORSTAB method over both the QMRCGSTAB and BiCORSTAB methods from the points of view of smooth convergence behavior and fast convergence rate in most circumstances.

Algorithm 3 *Left preconditioned QMRCORSTAB method.*

Compute $r_0 = b - Ax_0$ for some initial guess x_0.
Choose $r_0^* = P(A)r_0$ such that $\langle r_0^*, Ar_0\rangle \neq 0$, where $P(t)$ is a polynomial in t. (For example, $r_0^* = Ar_0$).
Solve $Mz_0^{BO} = r_0^{BO}$.
Set $\theta_0 = \eta_0 = 0$; $\tau = \left\| z_0^{BO} \right\|$; $zd_0 = e_0 = \mathbf{0}$; $r_0^{BO} = r_0$.
for $i = 1, 2, \ldots$ **do**
  $\hat{z} = Az_{i-1}^{BO}$
  $\rho_{i-1} = \langle r_0^*, \hat{z}\rangle$
  if $\rho_{i-1} = 0$, **method fails**
  if $i = 1$ **then**
    $p_0 = r_0^{BO}$
    solve $Mzp_0 = p_0$
    $q_0 = \hat{z}$
  else
    $\beta_{i-2} = (\rho_{i-1}/\rho_{i-2}) \times (\alpha_{i-2}/\omega_{i-2})$
    $zp_{i-1} = z_{i-1}^{BO} + \beta_{i-2}(zp_{i-2} - \omega_{i-2}zq_{i-2})$
    $q_{i-1} = \hat{z} + \beta_{i-2}(q_{i-2} - \omega_{i-2}\hat{zq}_{i-2})$
  end if
  solve $Mzq_{i-1} = q_{i-1}$
  $\hat{zq}_{i-1} = Azq_{i-1}$
  $\alpha_{i-1} = \rho_{i-1}/\langle r_0^*, \hat{zq}_{i-1}\rangle$
  $s_{i-1} = r_{i-1}^{BO} - \alpha_{i-1}q_{i-1}$
  $zs_{i-1} = z_{i-1}^{BO} - \alpha_{i-1}zq_{i-1}$
  $\widetilde{\theta}_i = \|zs_{i-1}\|/\tau$; $c = 1/\sqrt{1+\widetilde{\theta}_i^2}$; $\widetilde{\tau} = \tau\widetilde{\theta}_i c$
  $\widetilde{\eta}_i = c^2\alpha_{i-1}$
  $\widetilde{zd}_i = zp_{i-1} + \frac{\theta_{i-1}^2\eta_{i-1}}{\alpha_{i-1}} zd_{i-1}$
  $\widetilde{x}_i = x_{i-1} + \widetilde{\eta}_i\widetilde{zd}_i$
  $\widetilde{e}_i = q_{i-1} + \frac{\theta_{i-1}^2\eta_{i-1}}{\alpha_{i-1}} e_{i-1}$
  $\widetilde{r}_i = r_{i-1} - \widetilde{\eta}_i\widetilde{e}_i$
  check norm of $\widetilde{r}_i$; if small enough then stop
  $t = \hat{z} - \alpha_{i-1}\hat{zq}_{i-1}$
  $\omega_{i-1} = \langle t, s_{i-1}\rangle / \langle t, t\rangle$
  solve $Mzt = t$
  $r_i^{BO} = s_{i-1} - \omega_{i-1}t$
  $z_i^{BO} = zs_{i-1} - \omega_{i-1}zt$
  $\theta_i = \left\| z_i^{BO} \right\| / \widetilde{\tau}$; $c = 1/\sqrt{1+\theta_i^2}$; $\tau = \widetilde{\tau}\theta_i c$
  $\eta_i = c^2\omega_{i-1}$
  $zd_i = zs_{i-1} + \frac{\widetilde{\theta}_i^2\widetilde{\eta}_i}{\omega_{i-1}}\widetilde{zd}_i$
  $x_i = \widetilde{x}_i + \eta_i zd_i$
  $e_i = t + \frac{\widetilde{\theta}_i^2\widetilde{\eta}_i}{\omega_{i-1}}\widetilde{e}_i$
  $r_i = \widetilde{r}_i - \eta_i e_i$
  check convergence; continue if necessary
end for

Table 2
Parameter settings for Example 4.1.

Set	γ	β	*Gridsize*
1	[50:10:80]	−100	15
2	50	−[100:100:400]	15
3	50	−100	[15:2:21]

4. Numerical experiments

In this section, we report three typical sets of numerical experiments with the BiCORSTAB, QMRCGSTAB and QMRCORSTAB methods. For the detailed comparison and analysis of the BiCGSTAB and QMRCGSTAB method, we refer to [23]. The sets of test matrices are respectively obtained from a partial differential operator for a three-dimensional convection–diffusion problem discretized by finite difference scheme, numerical solution of boundary integral equations in radar-cross-section calculation of three-dimensional objects in electromagnetic scattering by the Method of Moments, and the University of Florida Sparse Matrix Collection [17]. The experiments have been carried out in double precision floating point arithmetic with MATLAB (Version 7.0.4.365 (R14) Service Pack 2 with License Number 254509) on PC-Intel(R) Core(TM) i7-3630QM CPU 2.40 GHz, 8 GB RAM. Here we use *Iters* and *Trr* to denote the number of iterations and $\log_{10}$ of

Table 3
Comparison results for Example 4.1 with different parameter settings in Table 2. (The values in bold indicate the best therein.)

Method		BiCORSTAB		QMRCORSTAB		QMRCGSTAB	
Parameter setting		*Iters*	*Trr*	*Iters*	*Trr*	*Iters*	*Trr*
Set 1 (γ)	50	**101**	−8.001	104.5	−8.5407	132.5	−8.0433
	60	90	−8.0127	**84.5**	−8.0097	106	−8.2199
	70	96.5	−8.0307	**89.5**	−8.1069	113.5	−8.5615
	80	110.5	−8.0916	**94.5**	−8.1603	125.5	−8.2991
Set 2 (β)	−100	**101**	−8.001	104.5	−8.5407	132.5	−8.0433
	−200	**134.5**	−8.6469	146	−8.1091	211.5	−8.1112
	−300	336.5	−8.189	**210.5**	−8.0177	673	−8.0056
	−400	–	−5.8637	–	−6.4624	–	−1.496
Set 3 (*gridsize*)	15	**101**	−8.001	104.5	−8.5407	132.5	−8.0433
	17	61.5	−8.8828	**58.5**	−8.3004	160	−8.0513
	19	174.5	−8.3285	**157**	−8.0968	217.5	−9.8828
	21	97.5	−8.0487	**93.5**	−8.7525	259.5	−8.2377

the final true relative residual 2-norm defined as $\log_{10} \frac{\|b-Ax_{\text{final}}\|_2}{\|r_0\|_2}$, respectively. It is noted from Table 1 that the number of *MVPs* is twice *Iters*. In all the context, a zero initial guess is taken. The stopping criterion used here is that the 2-norm of the approximate residual be reduced by a factor (referred to as targeted backward error *Tol*) of the 2-norm of the initial residual, i.e., $\|r_k\|_2/\|r_0\|_2 \leq Tol$, or when *Iters* exceeded the maximal iteration number (referred to as *MAXIT*). Whenever the considered problem contains no right-hand side to the original linear system $Ax = b$, let $b = Ae$, where e is the $n \times 1$ vector whose elements are all equal to unity, such that $x = (1, 1, \ldots, 1)^T$ is the exact solution. The convergence histories show *MVPs* (on the horizontal axis) versus 2-norms of the approximate relative residuals (on the vertical axis) in all figures. A symbol "–" is used to indicate that the method did not meet the required *Tol* before *MAXIT* or did not converge at all.

4.1. A three-dimensional convection–diffusion problem

In this problem, the necessity for the development of the QMRCORSTAB method as well as its efficiency obtained will be justified in comparison with its counterpart—the QMRCGSTAB method [23] and its basic parent—the BiCORSTAB method [13] by contriving some testing matrices generated by the central finite difference scheme to discretize a three-dimensional (3D) convection–diffusion equation [23]

$$L(u) = -\Delta u + \gamma(xu_x + yu_y + zu_z) + \beta u$$

on the unit cube with different settings for the parameters γ, β and *gridsize* (representing grid size) involved. The resulting coefficient matrix is then of order $n = gridsize^3$. For typically comprehensive comparison, three sets of parameter settings are designed by varying the parameters γ, β and *gridsize* separately as shown in Table 2. For the convenience of observing influences of different values of these three parameters on the solvability of these involving three iterative solvers, each set starts with the basic parameter setting of $\gamma = 50$, $\beta = -100$, and $gridsize = 15$ as taken for the Example 4 in [23]. As will be noticed for these specifically designed experiments that for this operator, not only large values in magnitude of γ but also β will add the unfavorable solvable difficulty for the BiCGSTAB-type methods; see [23,33,34] for related discussions.

No preconditioning was used. Here, we set $Tol = 10^{-8}$ and $MAXIT = 2000$. The comparative figures are listed in Table 3 and the convergence histories are displayed in Figs. 1–3 correspondingly to the three sets of parameter settings. The QMRCORSTAB method outperforms the QMRCGSTAB method in all cases in terms of *MVPs*, especially when $\beta = -300$ in *Set* 2 (β) and $gridsize = 21$ in *Set* 3 (*gridsize*). For the comparison between the QMRCORSTAB and BiCORSTAB methods, the former indeed adds dramatically favorable smoothing effect to the latter, since the latter has irregular convergence behaviors along its convergence history. And with the increasing values in magnitude of these three parameters, the QMRCORSTAB method turns to be better than the BiCORSTAB method in terms of *MVPs*. In addition, it is observed in the last plot of Fig. 2 and the last line for *Set* 2 (β) in Table 3, all the three solvers cannot converge to the targeted accuracy because of stagnancy for the case of $\beta = -400$ in *Set* 2 (β), which means the yielding problem associated with this case becomes more difficult to solve for the three solvers. However, the QMRCORSTAB method can converge to the accuracy of $Tol = 10^{-6}$ with $MVPs = 1788$ while the other two solvers cannot in this case.

4.2. Electromagnetic problems

The second set of linear systems arises from the numerical solution of boundary integral equations in radar-cross-section (RCS) calculation of 3D objects in electromagnetic scattering. The underlying mathematical model is called the *Electric Field Integral Equation* (EFIE). We point the reader to, e.g., [35, page 25] for a thorough description of this model. The EFIE formulation can be applied to arbitrary geometries, including those with open surfaces and cavities, hence it is very popular in scattering analysis. The Method of Moments discretization gives rise to indefinite and ill-conditioned

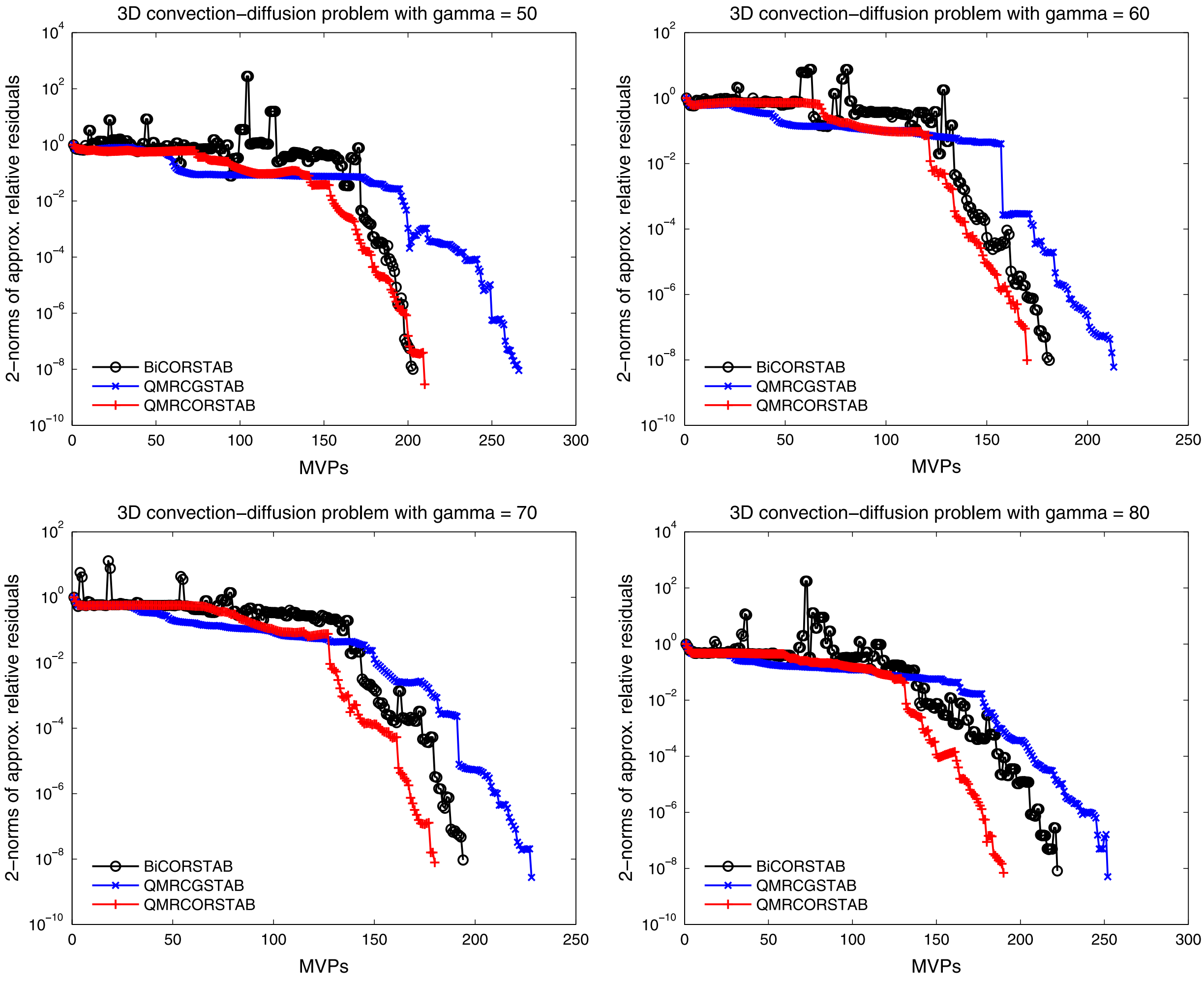

Fig. 1. Convergence histories of Example 4.1 with *Set* 1 (γ) in Table 2.

Table 4
Set and characteristics of test matrix problems (listed in order of increasing size).

Example	Size	Frequency (MHz)	Geometry
1	1080	47.47	Guide
2	1699	57.14	Satellite
3	1980	710.87	Paraboloid
4	4932	158.34	Cylinder

systems that are notoriously tough to solve by iterative methods, compared to other surface integral formulations for the same application [36]. The pertinent data matrix A is dense, complex, non-Hermitian; the right-hand side b varies with the frequency and the direction of the illuminating wave.

In earlier work, the authors have reported on the remarkable robustness and efficiency of the BiCORSTAB method for solving this problem class [15,37]. In this section we analyse numerically the effect of the quasi-minimal residual strategy by running MATLAB experiments on selected matrix problems. In Table 4 we summarize the characteristics of the linear systems that we considered in our experiments, corresponding to electromagnetic scattering from four different geometries. Although not large, their solution demanded considerable computer resources in MATLAB. e.g., storing the pertinent linear system for the cylinder problem (Example 4) requires around 370 Mb RAM when symmetry is not exploited. Larger problems would need a Fortran implementation of the solvers. However, the selected problems are representative of realistic RCS calculation and are difficult to solve for iterative methods due to indefiniteness of the data matrix. On the small satellite problem (Example 2, $n = 1699$), the un-preconditioned BiCORSTAB method required 1305.5 iterations to reduce the initial residual by eight orders of magnitude starting from the zero vector and using a physical right-hand side; the QMRCORSTAB method required 1289 iterations and the QMRCGSTAB method 1406 iterations. On the largest system,

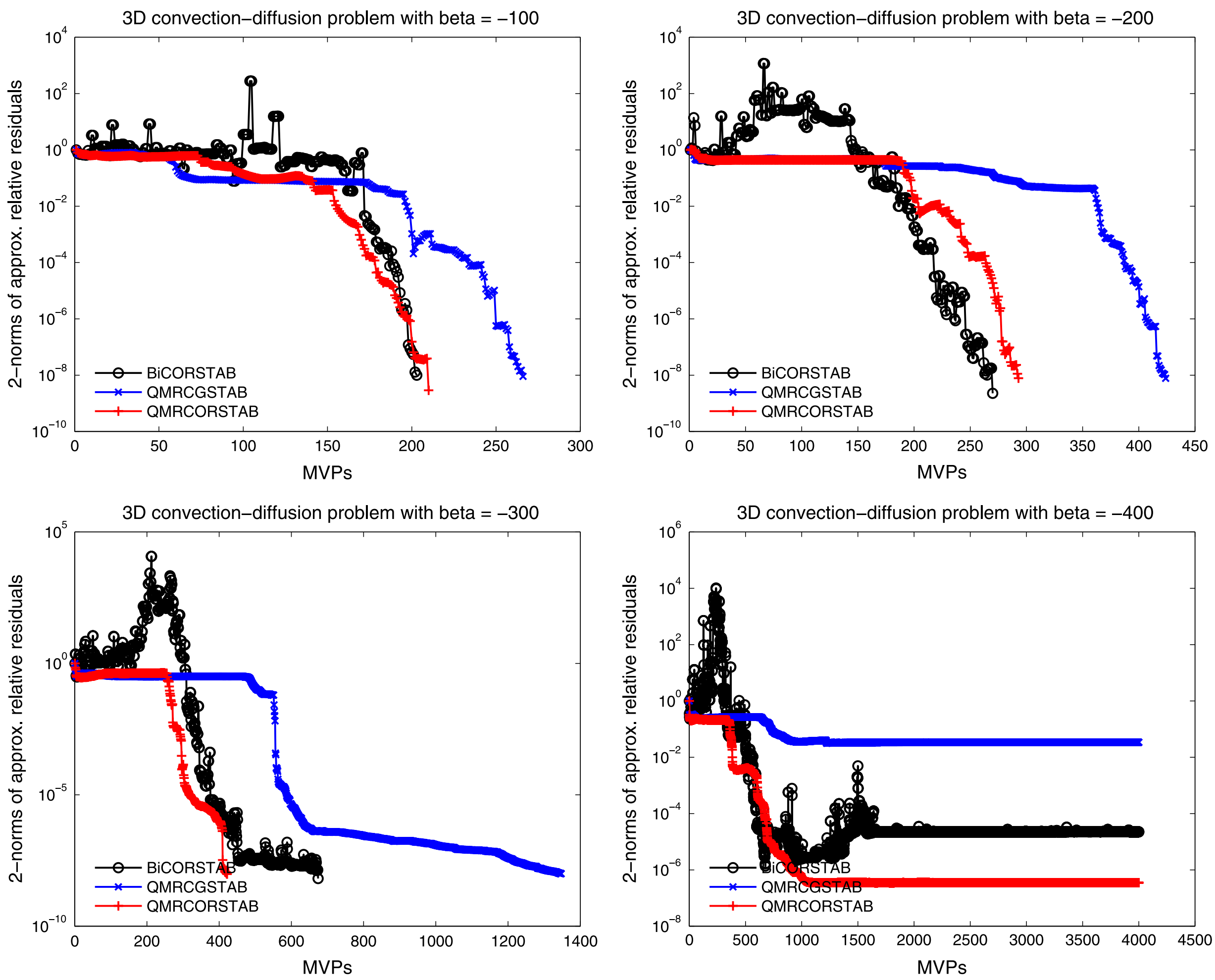

Fig. 2. Convergence histories of Example 4.1 with *Set* 2 (β) in Table 2.

Table 5
Comparison results for solving dense problems in electromagnetics in Example 4.2. (The values in bold indicate the best therein.)

Method	BiCORSTAB		QMRCORSTAB		QMRCGSTAB	
Example	*Iters*	*Trr*	*Iters*	*Trr*	*Iters*	*Trr*
1	90	−8.0300	**84.5**	−8.3221	89.5	−8.0658
2	246.5	−8.0444	**244**	−8.0613	255.5	−8.0015
3	168.5	−8.2342	**144.5**	−8.0544	156.5	−8.3216
4	82.5	−8.1291	**81.5**	−8.0029	87	−8.0148

the cylinder problem (Example 4), the BiCORSTAB method required 1540.5 iterations, the QMRCORSTAB method 1461.5 iterations and the QMRCGSTAB method 1493.5 iterations. Therefore, preconditioning is critically needed. The choice of effective preconditioning methods for boundary integral equations is a difficult issue on its own hand, see e.g. discussions in [36]. In our experiments, we preconditioned the linear system by using a multilevel inverse-based incomplete LU factorization. We point the reader to [38] for a detailed description of this preconditioner, and to [39] for an assessment of its performance for solving the EFIE formulation. The preconditioner was computed from a sparse approximation to the dense coefficient matrix, constructed by retaining the, say k, largest entries per column of A. We chose $k < 100$ for every problem, and tuned this parameter according to the size of the system to solve.

We clearly see from the results of Table 5 and from the convergence histories shown in Fig. 4 the good potential of the quasi-minimal residual strategy to smooth the residual and to improve the performance of the BiCORSTAB method to some extent. In most of our runs, the QMRCORSTAB method outperformed the QMRCGSTAB method.

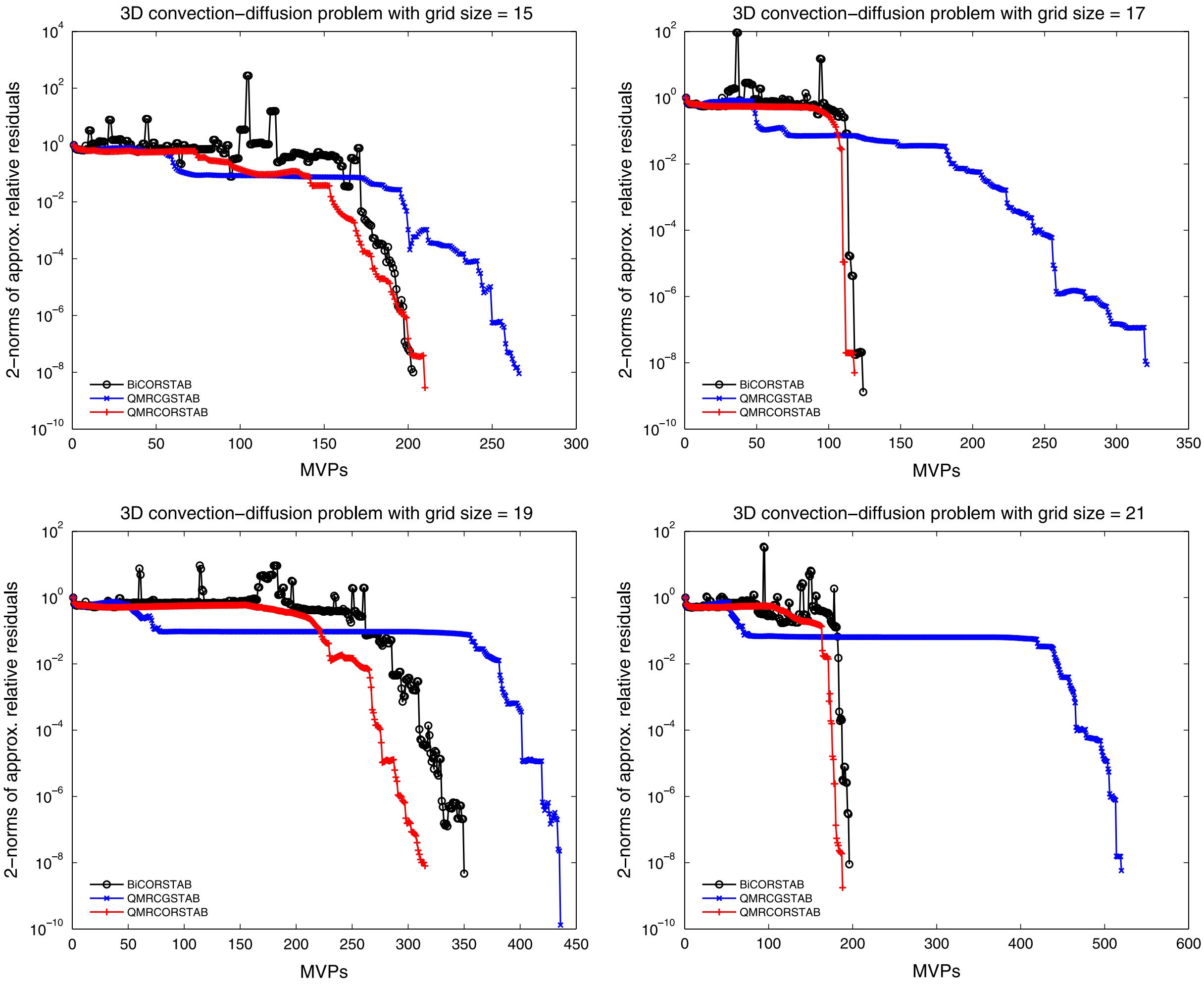

Fig. 3. Convergence histories of Example 4.1 with *Set* 3 (*gridsize*) in Table 2.

Table 6
Set and characteristics of test matrix problems (listed in alphabetical order).

Group and name	Size	Nonzeros	Problem kind
Bai/rdb5000	5000	29,600	Computational fluid dynamics problem
HB/sherman5	3312	20,793	Computational fluid dynamics problem
Quaglino/viscoplastic1	4326	61,166	Viscoplastic collision problem
HB/young3c	841	3,988	Acoustics problem

4.3. Test problems from University of Florida Sparse Matrix Collection

Finally, in order to further demonstrate the advantages gained by applying the quasi-minimal residual strategy to the BiCORSTAB method, a few more but not extensive test problems as listed in Table 6 are borrowed in the Matrix Market format from the University of Florida Sparse Matrix Collection [17] to compare the QMRCORSTAB method with the BiCORSTAB and QMRCGSTAB methods. The targeted backward error is $Tol = 10^{-8}$ and $MAXIT = 4000$ for all tests. No preconditioning was used except for Quaglino/viscoplastic1, where an ILU(0) preconditioner was implemented to ensure the convergence within a reasonable amount of iterations.

Numerical results are presented in Table 7 and the corresponding convergence histories are shown in Fig. 5. It is demonstrated again that the QMRCORSTAB method shares much smoother convergence behavior than the BiCORSTAB method and it is better than (at least almost the same as) the latter in terms of *MVPs*. Moreover, the QMRCORSTAB method is quite competitive with its counterpart—the QMRCGSTAB method in such kinds of problems.

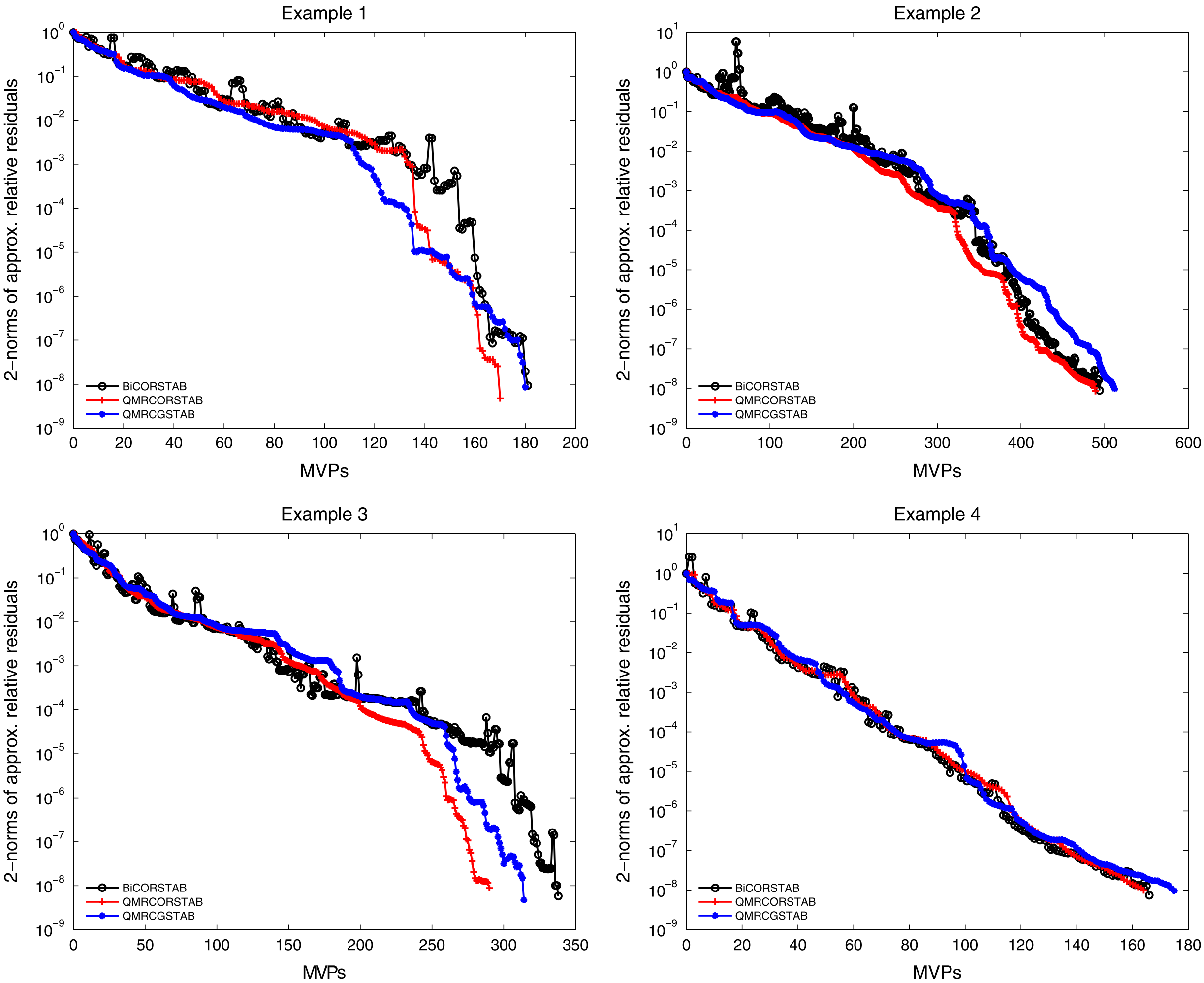

Fig. 4. Convergence histories on the dense matrix problems from Electromagnetics applications in Example 4.2.

Table 7
Comparison results for Example 4.3. (The values in bold indicate the best therein.)

Method	BiCORSTAB		QMRCORSTAB		QMRCGSTAB	
Matrix problem	*Iters*	*Trr*	*Iters*	*Trr*	*Iters*	*Trr*
Bai/rdb5000	306.5	−8.2221	**234.5**	−8.0325	263	−8.006
HB/sherman5	2719.5	−8.1856	**2670**	−8.0568	3412.5	−8.0293
Quaglino/viscoplastic1	**403.5**	−8.0804	408.5	−8.0966	404.5	−8.1061
HB/young3c	947.5	−8.8996	**897**	−8.3103	954.5	−8.041

5. Conclusions

In this paper, we have developed a QMRCORSTAB method based on the quasi-minimization of the residual using standard Givens rotations that lead to methods with short-term recurrences and smooth convergence curves. The numerical experiments verified that the QMRCORSTAB method converges more smoothly than the basic BiCORSTAB method, especially gives better performance when the BiCORSTAB method has irregular convergence behaviors with oscillations along the convergence history. However, it is commented that along with our experimental implementation, the QMR-type variants are observed to be sensitive to system properties, right-hand side and preconditioner chosen; see [40–42] for relevant issues and analyses. Also as pointed out by Simoncini and Szyld [6] that it is still challenging to substantially improve the performance of Krylov subspace method on large applications, when no a priori information on the problem is available. By the way, all the above algorithms can be implemented with various deflated and augmented techniques to accelerate convergence speed [3,5], which is another further issue to be discussed.

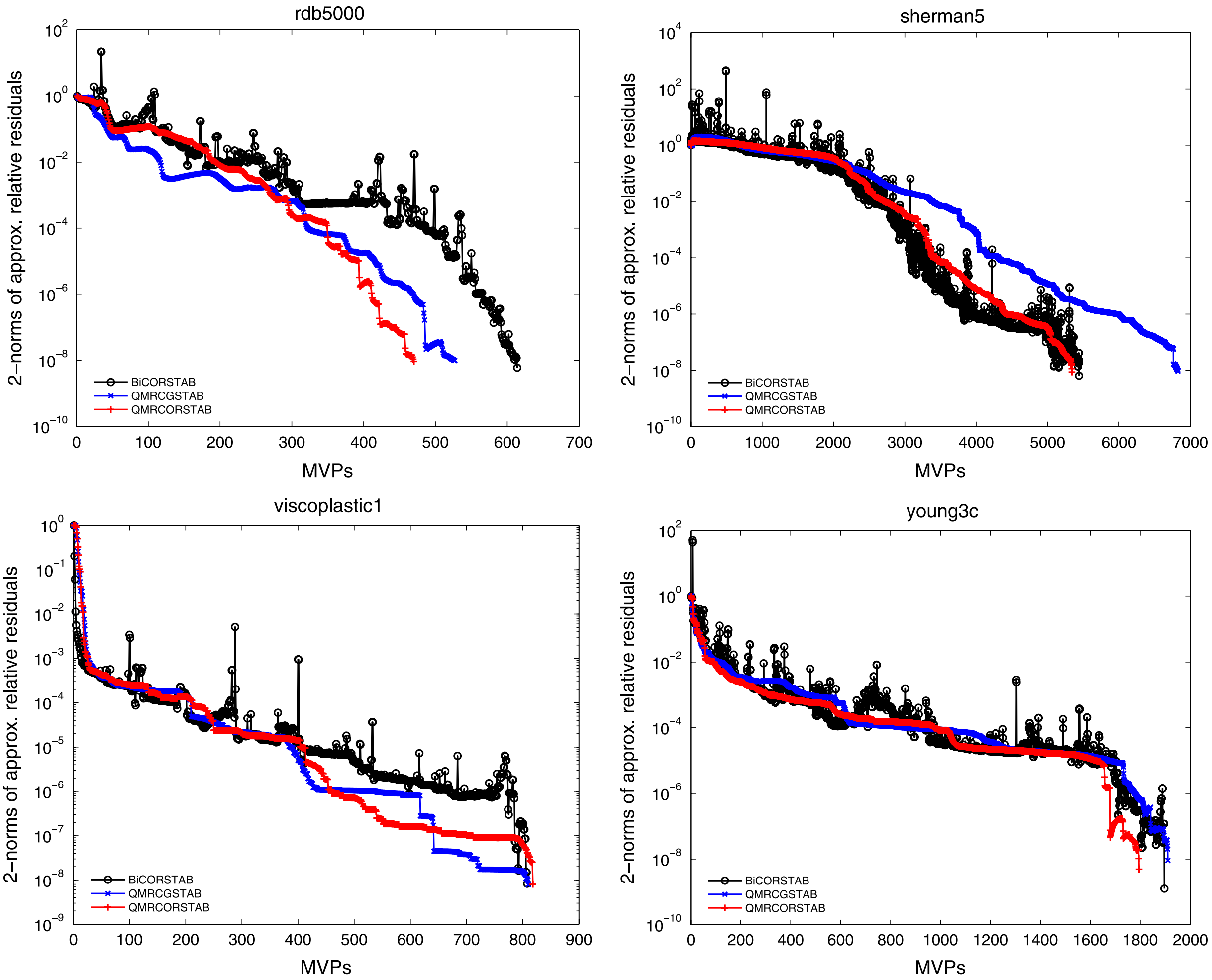

Fig. 5. Convergence histories of Example 4.3.

Acknowledgments

The authors would like to thank Professor Dr. Leszek Feliks Demkowicz and the anonymous reviewers for their valuable comments and suggestions. One of the reviewers drew our attention to [43], which contributed substantially to this work and added nicely to the presentation of the preconditioned QMRCORSTAB method. The second author is grateful to the Computational Mechanics and Numerical Mathematics Group at the Johann Bernoulli Institute for Mathematics and Computer Science, University of Groningen (the Netherlands) for warm hospitality.

This research is supported by 973 Program (2013CB329404), National Natural Science Foundation of China (61370147, 61170311, 61170309, 11201055), Chinese Universities Specialized Research Fund for the Doctoral Program (20110185110020, 20120185120026), the State Scholarship Fund from the China Scholarship Council, the Fundamental Research Funds for the Central Universities.

References

[1] Y. Saad, M.H. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (1986) 856–869.
[2] Y. Saad, Iterative Methods for Sparse Linear Systems, second ed., The PWS Publishing Company, Boston, 1996, SIAM: Philadelphia, PA, 2003.
[3] A. Gaul, M.H. Gutknecht, J. Liesen, R. Nabben, A framework for deflated and augmented Krylov subspace methods, SIAM J. Matrix Anal. Appl. 34 (2013) 495–518.
[4] L. Giraud, S. Gratton, X. Pinel, X. Vasseur, Flexible GMRES with deflated restarting, SIAM J. Sci. Comput. 32 (2010) 1858–1878.
[5] M.H. Gutknecht, Deflated and augmented Krylov subspace methods: a framework for deflated BiCG and related solvers, 2014. Available online at http://www.sam.math.ethz.ch/~mhg/pub/defl-MHG2/MHG2revcol.pdf.
[6] V. Simoncini, D.B. Szyld, Recent computational developments in Krylov subspace methods for linear systems, Numer. Linear Algebra Appl. 14 (2007) 1–59.
[7] R. Fletcher, Conjugate Gradient Methods for Indefinite Systems, in: Lecture Notes in Mathematics, vol. 506, Springer-Verlag, Berlin, Heidelberg, New York, 1976, pp. 73–89.

[8] P. Sonneveld, CGS: a fast Lanczos-type solver for nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 10 (1989) 36–52.
[9] H.A. van der Vorst, Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput. 13 (1992) 631–644.
[10] M.H. Gutknecht, Variants of BiCGSTAB for matrices with complex spectrum, SIAM J. Sci. Comput. 14 (1993) 1020–1033.
[11] G.L.G. Sleijpen, D.R. Fokkema, BiCGSTAB(l) for linear equations involving unsymmetric matrices with complex spectrum, Electron. Trans. Numer. Anal. 1 (1993) 11–32.
[12] S.-L. Zhang, GPBi-CG: generalized product-type methods based on Bi-CG for solving nonsymmetric linear systems, SIAM J. Sci. Comput. 18 (1997) 537–551.
[13] Y.-F. Jing, T.-Z. Huang, Y. Zhang, L. Li, G.-H. Cheng, Z.-G. Ren, Y. Duan, T. Sogabe, B. Carpentieri, Lanczos-type variants of the COCR method for complex nonsymmetric linear systems, J. Comput. Phys. 228 (17) (2009) 6376–6394.
[14] B. Carpentieri, Y.-F. Jing, T.-Z. Huang, The BiCOR and CORS iterative algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Comput. 33 (2011) 3020–3036.
[15] Y.-F. Jing, B. Carpentieri, T.-Z. Huang, Experiments with Lanczos biconjugate A-orthonormalization methods for MoM discretizations of Maxwell's equations, Prog. Electromagn. Res., PIER 99 (2009) 427–451.
[16] Y.-F. Jing, T.-Z. Huang, Y. Duan, B. Carpentieri, A comparative study of iterative solutions to linear systems arising in quantum mechanics, J. Comput. Phys. 229 (22) (2010) 8511–8520.
[17] T.A. Davis, Y. Hu, The University of Florida sparse matrix collection, ACM Trans. Math. Softw. 38 (1) (2011) 1:1–1:25.
[18] L. Zhao, T.-Z. Huang, Y.-F. Jing, L.-J. Deng, A generalized product-type BiCOR method and its application in signal deconvolution, Comput. Math. Appl. 66 (2013) 1372–1388.
[19] L. Zhao, T.-Z. Huang, A hybrid variant of the BiCOR method for a nonsymmetric linear system with a complex spectrum, Appl. Math. Lett. 26 (2013) 457–462.
[20] L. Du, T. Sogabe, S.-L. Zhang, A variant of the IDR(s) method with the quasi-minimal residual strategy, J. Comput. Appl. Math. 236 (2011) 621–630.
[21] R.W. Freund, N.M. Nachtigal, QMR: a quasi-minimal residual method for non-Hermitian linear systems, Numer. Math. 60 (1991) 315–339.
[22] R.W. Freund, A transpose-free quasi-minimal residual algorithm for non-Hermitian linear systems, SIAM J. Sci. Comput. 14 (1993) 470–482.
[23] T.F. Chan, E. Gallopoulos, V. Simoncini, T. Szeto, C. Tong, A quasi-minimal residual variant of the Bi-CGSTAB algorithm for nonsymmetric systems, SIAM J. Sci. Comput. 15 (1994) 338–347.
[24] Z.H. Cao, D.J. Evans, On quasi-minimal residual approach of iterative algorithms for solving nonsymmetric linear systems, Int. J. Comput. Math. 62 (3–4) (1996) 249–270.
[25] Z.H. Cao, On the QMR approach for iterative methods including coupled three-term recurrences for solving nonsymmetric linear systems, Appl. Numer. Math. 27 (2) (1998) 123–140.
[26] M.B. Van Gijzen, P. Sonneveld, Algorithm 913: an elegant IDR(s) variant that efficiently exploits bi-orthogonality properties, ACM Trans. Math. Softw. 38 (1) (2011) 5:1–5:19.
[27] M.B. Van Gijzen, G.L.G. Sleijpen, J.-P.M. Zemke, Flexible and multi-shift induced dimension reduction algorithms for solving large sparse linear systems, Report 11-06, DIAM, TU Deft, and Bericht 156, INS, TU Hamburg-Harburg, August 29, 2011.
[28] M. Hajarian, The generalized QMRCGSTAB algorithm for solving Sylvester-transpose matrix equations, Appl. Math. Lett. 26 (10) (2013) 1013–1017.
[29] K.J. Ressel, M.H. Gutknecht, QMR smoothing for Lanczos-type product methods based on three-term rrecurrences, SIAM J. Sci. Comput. 19 (1) (1998) 55–73.
[30] D.B. Szyld, J.A. Vogel, FQMR: a flexible quasi-minimal residual method with inexact preconditioning, SIAM J. Sci. Comput. 23 (2) (2001) 363–380.
[31] K. Abe, G.L.G. Sleijpen, BiCR variants of the hybrid BiCG methods for solving linear systems with nonsymmetric matrices, J. Comput. Appl. Math. 234 (2010) 985–994.
[32] T. Sogabe, M. Sugihara, S.-L. Zhang, An extension of the conjugate residual method to nonsymmetric linear systems, J. Comput. Appl. Math. 226 (1) (2009) 103–113.
[33] O.G. Ernst, M.J. Gander, Why it is difficult to solve Helmholtz problems with classical iterative methods, in: Numerical Analysis of Multiscale Problems, in: Lecture Notes in Computational Science and Engineering, vol. 83, 2012, pp. 325–363.
[34] M.M. Gupta, J. Zhang, High accuracy multigrid solution of the 3D convection–diffusion equation, Appl. Math. Comput. 113 (2000) 249–274.
[35] W.C. Gibson, The Method of Moments in Electromagnetics, Chapman & Hall/CRC, Boca Raton, FL, 2008, p. xv. 272 p. $ 119.95.
[36] B. Carpentieri, I.S. Duff, L. Giraud, G. Sylvand, Combining fast multipole techniques and an approximate inverse preconditioner for large electromagnetism calculations, SIAM J. Sci. Comput. 27 (3) (2005) 774–792.
[37] B. Carpentieri, Y.-F. Jing, T.-Z. Huang, W.-C. Pi, X.-Q. Sheng, Combining the CORS and BiCORSTAB iterative methods with MLFMA and SAI preconditioning for solving large linear systems in electromagnetics, Appl. Comput. Electromagn. Soc. J. 27 (2) (2012) 102–111.
[38] M. Bollhöfer, Y. Saad, Multilevel preconditioners constructed from inverse-based ILUs, SIAM J. Sci. Comput. 27 (5) (2006) 1627–1650.
[39] B. Carpentieri, M. Bollhöfer, Symmetric inverse-based multilevel ILU preconditioning for solving dense complex non-hermitian systems in electromagnetics, Prog. Electromagn. Res., PIER 128 (2012) 55–74.
[40] M.D. García, E. Flórez, A. Suárez, L. González, G. Montero, New implementation of QMR-type algorithms, Comput. Struct. 83 (28–30) (2005) 2414–2422.
[41] G.L.G. Sleijpen, H.A. van der Vorst, Maintaining convergence properties of BiCGSTAB methods in finite precision arithmetic, Numer. Algorithms 10 (2) (1995) 203–223.
[42] G.L.G. Sleijpen, H.A. van der Vorst, An overview of approaches for the stable computation of hybrid BiCG methods, Appl. Numer. Math. 19 (3) (1995) 235–254.
[43] G. Montero, R. Montenegro, J.M. Escobar, E. Rodríguez, Resolution of sparse linear systems of equations: the RPK strategy, in: Progress in Engineering Computational Technology, Saxe-Coburg Publications, 2004, pp. 81–110.