

USAGE OF THE CONVERGENCE TEST OF THE RESIDUAL NORM IN THE TSUNO-NODERA VERSION OF THE GMRES ALGORITHM

K. MORIYA¹ and T. NODERA²

(Received 13 October, 2005; revised 15 January, 2007)

Abstract

Tsuno and Nodera proposed a new variant of the GMRES(m) algorithm. Their algorithm is referred to as the GMRES($\leq m_{\max}$) algorithm and performs the restart process adaptively, considering the distribution of the zeros of the residual polynomial. However, unless the zeros of the residual polynomial are distributed uniformly, m_{\max} is always chosen and their algorithm becomes almost the same as the GMRES(m) algorithm with $m = m_{\max}$.

In this paper, we include a convergence test for the residual norm in the GMRES($\leq m_{\max}$) algorithm and propose a new restarting technique based on two criteria. Even if the distribution of zeros does not become uniform, the restart can be performed by using the convergence test of the residual norm. Numerical examples simulated on a Compaq Beowulf computer demonstrate that the proposed technique accelerates the convergence of the GMRES($\leq m_{\max}$) algorithm.

2000 *Mathematics subject classification*: primary 65F10; secondary 65M12.

Keywords and phrases: linear system of equations, GMRES(m) algorithm, parallel computer, adaptive restart.

1. Introduction

We study a linear system of equations

$$Ax = b, \quad A \in R^{n \times n}, \quad x, b \in R^n, \quad (1.1)$$

where the coefficient matrix A is large, sparse, nonsingular and nonsymmetric. It is well known that the GMRES algorithm [12] is one of the major iterative solvers for System (1.1). It is difficult to implement this algorithm because of the computational

¹Ohi-Branch, Nikon System Inc., Japan; email: kmoriya@nikon-sys.co.jp.

²Department of Mathematics, Keio University, Japan; email: nodera@math.keio.ac.jp.

© Australian Mathematical Society 2007, Serial-fee code 1446-1811/07

cost and the difficulty of orthogonalization. The GMRES(m) algorithm – the restart version of the GMRES algorithm – is usually used in order to reduce the cost of orthogonalization. It is important to choose a suitable restart cycle m to avoid both the stagnation of the convergence of the residual norm and excessive orthogonalization cost. Up to now, many techniques that improve the GMRES(m) algorithm have been studied. Techniques that augment the Krylov subspace by using eigenvectors have been proposed [3, 8, 15]. Morgan [9] performs the restart using Harmonic Ritz values. Bağlama *et al.* [2] compute the preconditioner with spectral information obtained from an Arnoldi process. Moreover, Zítko [14] uses a pre-iteration technique before starting the GMRES(m) algorithm. Moriya and Nodera [10] studied an adaptive procedure that combines the DEFLATED-GMRES(m, k) algorithm and the determination of a restart frequency m automatically. Tsuno and Nodera [13] proposed a new variant of the GMRES(m) algorithm in which m is varied adaptively. In their technique, the restart occurs when the zeros of the residual polynomial are distributed on the complex plane uniformly. The restart must be carried out with the restart cycle $m = m_{\max}$, and the orthogonalization cost becomes excessive unless the distribution of the zeros becomes uniform. In this paper, we study a parameter for the convergence test of the residual norm and apply it to the restarting technique of Tsuno and Nodera [13]. The main objective of the proposed technique is to decrease the number of restart cycles.

A brief introduction to the residual polynomials and their zeros is included in Section 2. In Section 3, we combine a new criterion of the restart with the GMRES($\leq m_{\max}$) algorithm and describe the selection process for m . In Section 4, three examples are implemented on a Compaq MIMD parallel Beowulf computer, and a PC with a single processor. The numerical results for these examples are also reported and demonstrate the effectiveness of the proposed technique. Finally, we give some concluding remarks in Section 5.

2. Residual polynomials and their zeros

The GMRES and GMRES($\leq m_{\max}$) algorithms are introduced briefly, using their residual polynomials. The computation of the zeros of the residual polynomial which the GMRES($\leq m_{\max}$) algorithm employs is also studied.

2.1. Residual polynomial of the GMRES algorithm The ideal residual polynomial $\Psi^e(t)$ of the GMRES algorithm is defined as

$$\Psi^e(t) = \prod_j \left(1 - \frac{t}{\lambda_j^e}\right), \quad (2.1)$$

where λ_j^e are the zeros of the polynomial (2.1) and the exact eigenvalues of A . The zeros of this residual polynomial are harmonic Ritz values [4]. The ℓ -th dimensional

residual polynomial $\Psi_\ell^G(t)$ of the GMRES algorithm is defined as

$$\Psi_\ell^G(t) = \prod_{j=0}^{\ell} \left(1 - \frac{t}{\lambda_j^G}\right). \quad (2.2)$$

By using the residual polynomial (2.2), the ℓ -th residual vector of the GMRES algorithm is described as

$$\mathbf{r}_\ell = \Psi_\ell^G(t) \mathbf{r}_0,$$

where the zeros λ_j^G are computed so that the residual norm $\|\mathbf{r}_\ell\|_2$ is minimized.

The zeros λ_j^G are chosen to minimize $\|\mathbf{r}_\ell\|_2$ and are not always chosen to be close to λ_j^e . However, as the dimension ℓ gets larger, the number of λ_j^G increases, and it is more likely that λ_j^G will be close to λ_j^e . It may be effective to increase the dimension of the polynomial (2.2) when λ_j^G are not sufficiently close to λ_j^e . In the following sections, λ_j^e are referred to as the “ideal zeros”. Generally speaking, it is almost impossible to compute the ideal zeros of the residual polynomial of the GMRES algorithm because the eigenvalues of A are required.

2.2. Residual polynomial of the GMRES($\leq m_{\max}$) algorithm We now assume that the restart has been performed i times and k iterations have been executed after the i -th restart. We describe the total number of iterations before the i -th restart as $\tilde{\ell} = \sum_{j=1}^i m_j$, where m_j is the j -th restart cycle and $m_j \leq m_{\max}$. The j -th restart cycle is the number of iterations from the $(j-1)$ -th to the j -th restart. Here

$$\Psi_k^{(i+1)}(t) = \prod_{s_1=1}^k \left(1 - \frac{t}{\lambda_{s_1}^{(i+1)}}\right), \quad (1 \leq k \leq m_{\max}) \quad \text{and} \quad (2.3)$$

$$\Psi_{m_j}^{(j)}(t) = \prod_{s_2=1}^{m_j} \left(1 - \frac{t}{\lambda_{s_2}^{(j)}}\right), \quad (1 \leq m_j \leq m_{\max}, j = 1, 2, \dots, i) \quad (2.4)$$

are two types of residual polynomials of the GMRES($\leq m_{\max}$) algorithm, whose dimensions are k and m_j , respectively. While the residual polynomial (2.3) is computed before the i -th restart, the residual polynomial (2.4) is obtained after the i -th restart. By using the product of these two polynomials, the ℓ -th ($= \sum_{j=1}^i m_j + k$) residual vector \mathbf{r}_ℓ is characterized as

$$\mathbf{r}_\ell = \Psi_k^{(i+1)}(A) \prod_{j=1}^i \Psi_{m_j}^{(j)}(A) \mathbf{r}_0,$$

where $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$ are the zeros of the residual polynomials (2.3) and (2.4), respectively.

If A is nonsymmetric, some of the ideal zeros λ_j^e are complex. In order to approach the ideal zeros, the dimensions of the residual polynomials (2.3) and (2.4) must be more than two. The simplest way to ensure this is that the dimensions of both the residual polynomials are chosen as two. However, if the residual vector r_ℓ is computed by only quadratic residual polynomials, it is likely that $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$ are not close to the ideal zeros at all. In this case, it may be of benefit to increase the dimensions of the residual polynomials (2.3) and (2.4) so that $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$ often become closer to the ideal zeros. However, the disadvantage of increasing the dimensions is that the computation cost per iteration becomes excessive. Therefore, it is important to choose an appropriate dimension of the residual polynomial. In the following sections, $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$ are referred to as the ‘‘approximate zeros’’ or ‘‘zeros’’.

2.3. Computation of zeros We show the implementation of the residual polynomial of the GMRES($\leq m_{\max}$) algorithm. The residual polynomial (2.3) is described as

$$\Psi_k^{(i+1)}(t) = 1 - \alpha_0 t - \alpha_1 t^2 - \dots - \alpha_{k-1} t^k. \tag{2.5}$$

We consider the computation of the coefficients of the residual polynomial (2.5) $\alpha_0, \alpha_1, \dots, \alpha_{k-1}$ based on Nachtigal *et al.* [11]. The orthonormal basis of the Krylov subspace $K_k(r_{\tilde{\ell}}, A)$ is defined as

$$V_k = [v_0, v_1, \dots, v_{k-1}],$$

where $r_{\tilde{\ell}}$ is the $\tilde{\ell}$ -th residual vector obtained when the last restart occurred. Two matrices R_k and C_k are also defined as

$$R_k = [r_{\tilde{\ell}}, Ar_{\tilde{\ell}}, \dots, A^{k-1}r_{\tilde{\ell}}] \quad \text{and} \quad C_k = \begin{bmatrix} c_{11} & \dots & c_{1k} \\ & \ddots & \vdots \\ 0 & & c_{kk} \end{bmatrix}.$$

Since V_k is the orthonormal basis of $K_k(r_{\tilde{\ell}}, A)$,

$$V_k = R_k C_k \tag{2.6}$$

is satisfied, where C_k is the upper triangular matrix.

The formula for the Arnoldi process can be described as

$$\begin{cases} v_{k+1} = h_{k+1,k}^{-1}(Av_k - V_k h_k) \\ h_k = [h_{1,k}, h_{2,k}, \dots, h_{k,k}]^T \end{cases}, \tag{2.7}$$

where $h_{i,j}$ is the i, j -th entry of the $(k + 1) \times k$ upper Hessenberg matrix

$$\bar{H}_k = \begin{bmatrix} h_{1,1} & h_{1,2} & \dots & h_{1,k} \\ h_{2,1} & h_{2,2} & \dots & h_{2,k} \\ & \ddots & \ddots & \vdots \\ 0 & & h_{k,k-1} & h_{k,k} \\ & & & h_{k+1,k} \end{bmatrix}.$$

The left-hand side of Equation (2.7) is described as

$$v_{k+1} = R_{k+1} [c_{1,k+1}, c_{2,k+1}, \dots, c_{k+1,k+1}]^T \tag{2.8}$$

by using Equation (2.6). From Equation (2.7),

$$Av_k = AV_k e_k = AR_k [c_{1,k}, c_{2,k}, \dots, c_{k,k}]^T = R_{k+1} [0, c_{1,k}, c_{2,k}, \dots, c_{k,k}]^T \tag{2.9}$$

is satisfied. Equation (2.6) can be transformed into

$$V_k = R_{k+1} \begin{bmatrix} C_k \\ 0 \end{bmatrix}. \tag{2.10}$$

By multiplying both sides of Equation (2.10) by h_k , we obtain

$$V_k h_k = R_{k+1} \begin{bmatrix} C_k h_k \\ 0 \end{bmatrix}. \tag{2.11}$$

Substituting (2.8), (2.9) and (2.11) in (2.7), we obtain the recurrences

$$\begin{bmatrix} c_{1,k+1} \\ c_{2,k+1} \\ \vdots \\ c_{k+1,k+1} \end{bmatrix} = h_{k+1,k}^{-1} \left(\begin{bmatrix} 0 \\ c_{1,k} \\ \vdots \\ c_{k,k} \end{bmatrix} - \begin{bmatrix} C_k h_k \\ 0 \end{bmatrix} \right).$$

Since $v_0 = c_{11} r_{\bar{i}}$ is satisfied from Equation (2.6), $c_{11} = 1/\|r_{\bar{i}}\|_2$ is also obtained. The ℓ -th residual vector of the GMRES($\leq m_{\max}$) algorithm is characterized as

$$r_\ell = r_{\bar{i}} - AV_k y_k, \tag{2.12}$$

where y_k is computed so that $\|r_\ell\|_2$ is minimized. Substituting (2.6) in (2.12),

$$r_\ell = r_{\bar{i}} - AR_k C_k y_k = r_{\bar{i}} - [Ar_{\bar{i}}, A^2 r_{\bar{i}}, \dots, A^k r_{\bar{i}}] C_k y_k$$

is obtained. Setting $C_k y_k = [\alpha_0, \alpha_1, \dots, \alpha_{k-1}]$, the coefficients of the residual polynomial $\Psi_k^{(i+1)}(t)$ are obtained from $C_k y_k$.

We consider the computation of the zeros of the residual polynomial (2.5), which has the same zeros as the k -dimensional monic polynomial

$$\bar{P}_k(t) = t^k + \sum_{j=1}^{k-1} \frac{\alpha_{j-1}}{\alpha_j} t^{j-1} - \frac{1}{\alpha_{k-1}}.$$

Since $\bar{P}_k(t)$ is also the characteristic polynomial of the matrix

$$D_k = \begin{bmatrix} 0 & \dots & 0 & 1/\alpha_{k-1} \\ 1 & \ddots & \vdots & \vdots \\ & \ddots & 0 & -\alpha_{k-3}/\alpha_{k-1} \\ 0 & & 1 & -\alpha_{k-2}/\alpha_{k-1} \end{bmatrix},$$

the zeros of the residual polynomial (2.5) are just the eigenvalues of matrix D_k . Since D_k is an upper Hessenberg matrix, QR decomposition is a typical scheme we can use for computing the eigenvalues. Another procedure for computing the zeros of the residual polynomial (2.5) was also studied by Freund [5].

3. Choosing the restart cycle

We propose a new restart technique, in which the convergence test of the residual norm is combined with the GMRES($\leq m_{\max}$) algorithm. The restart is determined based on two criteria.

3.1. Restart based on the distribution of zeros The restart technique proposed by Tsuno and Nodera [13] is presented. In order to accelerate the convergence of the residual norm, the approximate zeros are required to be close to the ideal zeros. The most appropriate approach is to perform the restart when the distribution of the approximate zeros is similar to that of the ideal zeros. However, it is impossible to ascertain the distribution of the ideal zeros. It would be significant to achieve a uniform distribution of the approximate zeros on the complex plane such that they overlap all the ideal zeros. Now, let us discuss two examples. We assume a distribution of the ideal zeros “o” as given in Figures 1(a) and (b). If all the approximate zeros are restricted to only a small region of the complex plane, as in Figure 1(a), the distribution of the approximate zeros may not encompass the distribution of the ideal zeros. On the other hand, if the approximate zeros are distributed uniformly, as shown in Figure 1(b), it becomes more likely that the approximate zeros are close to the ideal zeros. As the approximate zeros are distributed more uniformly, it further increases the possibility that the distribution of the approximate zeros overlaps that of the ideal zeros. In some cases, the ideal zeros may be in close proximity in a small region. However, as the number of approximate zeros increases and they are distributed more widely, the possibility that some of them get close to the ideal zeros would increase.

The approximate zeros are of two types, $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$. The first type, $\lambda_{s_1}^{(i+1)}$, are the zeros of the residual polynomial (2.3) and are varied as the iterations are increased. The second type, $\lambda_{s_2}^{(j)}$, are the zeros of the residual polynomial (2.4). Their values are fixed before the i -th restart is performed and are never varied after the i -th restart.

We now assume that there are ℓ approximate zeros on the complex plane. We also define M_{re} and M_{im} as the ranges of the real and imaginary directions of the rectangular region in which the approximate zeros exist, respectively. If all the approximate zeros are distributed uniformly on a region of size $M_{\text{im}} \times M_{\text{re}}$, they exist uniformly in the direction of either the imaginary or real axis. If the distribution of the approximate zeros is uniform in the direction of the real axis, the interval between two zeros is more than $M_{\text{re}}/(\ell - 1)$. Similarly, this interval is more than $M_{\text{im}}/(\ell - 1)$ in the direction of

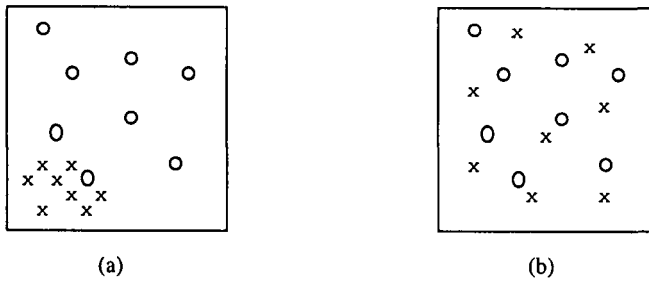


FIGURE 1. Distributions of ideal zeros (o) and approximate zeros (x). We illustrate an example of (a) nonuniform and (b) uniform distribution of the approximate zeros.

the imaginary axis. On the other hand, if the distance between two arbitrary zeros is less than $M_{re}/(\ell - 1)$ and $M_{im}/(\ell - 1)$ in the direction of the real axis and imaginary axis, respectively, the approximate zeros are not distributed uniformly. Based on the above information, Tsuno and Nodera [13] determined the restart when the following condition is satisfied.

CONDITION 1. For all $\lambda_{s_1}^{(i+1)}$, there are no zeros $\lambda_{s_2}^{(j)}$ in the rectangular region

$$T(\lambda_{s_1}^{(i+1)}) := \left\{ z \in \mathbb{C} \mid \text{re} |\lambda_{s_1}^{(i+1)} - z| < \frac{1}{2} \frac{M_{re}}{(\ell - 1)}, \quad \text{im} |\lambda_{s_1}^{(i+1)} - z| < \frac{1}{2} \frac{M_{im}}{(\ell - 1)} \right\},$$

where

$$M_{re} = \max_{s_1, s_2, j} \{ \text{re}(\lambda_{s_1}^{(i+1)}), \text{re}(\lambda_{s_2}^{(j)}) \} - \min_{s_1, s_2, j} \{ \text{re}(\lambda_{s_1}^{(i+1)}), \text{re}(\lambda_{s_2}^{(j)}) \} \quad \text{and}$$

$$M_{im} = \max_{s_1, s_2, j} \{ \text{im}(\lambda_{s_1}^{(i+1)}), \text{im}(\lambda_{s_2}^{(j)}) \} - \min_{s_1, s_2, j} \{ \text{im}(\lambda_{s_1}^{(i+1)}), \text{im}(\lambda_{s_2}^{(j)}) \}.$$

In Figure 2 there are some fixed and new zeros. In this figure, there are also some rectangulars $T(\lambda_{s_1}^{(i+1)})$ whose central points are new zeros $\lambda_{s_1}^{(i+1)}$. The rectangulars are not clustered. In order to distribute all zeros uniformly, there must be only one zero in each rectangular. If fixed zeros $\lambda_{s_2}^{(j)}$ exist in these rectangulars, it follows that there are more than two zeros in such rectangulars. Therefore, when Condition 1 is satisfied, all the zeros are distributed uniformly and k new zeros $\lambda_{s_1}^{(i+1)}$ should be fixed. Figure 2(b) illustrates the relation between $\lambda_{s_1}^{(i+1)}$ and $\lambda_{s_2}^{(j)}$ when Condition 1 is satisfied. In Figure 2(a), it is obvious that $\lambda_{s_2}^{(j)}$ exists in the rectangular region $T(\lambda_{s_1}^{(i+1)})$ and Condition 1 is not satisfied.

We study the reduction in the computational cost of the zeros. If the coefficient matrix A is real, the coefficients of the residual polynomial are also real, and all the zeros are distributed symmetrically about the real axis. Therefore, only the zeros whose imaginary parts are non-negative are required for the determination of the restart. If the restart is performed at an odd step, at least one zero is real. In this

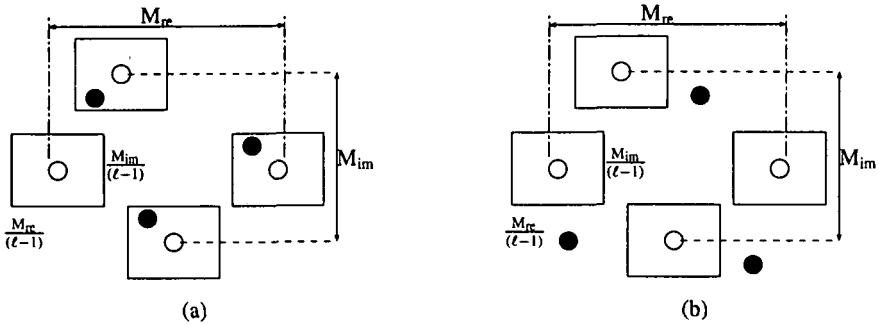


FIGURE 2. Interval between $\lambda_{s_1}^{(i+1)}$ (o) and $\lambda_{s_2}^{(j)}$ (•). We show the cases where (a) the zeros are not distributed uniformly and (b) the zeros are distributed uniformly. The rectangles with central points “o” are $T(\lambda_{s_1}^{(i+1)})$ of order $M_{re}/(\ell-1) \times M_{im}/(\ell-1)$.

case, if all the ideal zeros are complex, the real zeros do not approximate any ideal zeros. Therefore, the determination of the restart is carried out only at even steps. Since there are no zeros on the complex plane at the start of the iteration, the restart is always performed at the 2nd iteration step in order to produce the initial distribution of the zeros.

3.2. The convergence test of the residual norm The $GMRES(\leq m_{max})$ algorithm has the drawback that it becomes necessary to perform the restart with m_{max} compulsorily if the approximate zeros do not become distributed uniformly. In order to improve this drawback, we apply the convergence test of the residual norm to the original $GMRES(\leq m_{max})$ algorithm. The main objective of the proposed technique is to reduce the occurrence of the restart cycle reaching m_{max} .

We study the $\ell (= \tilde{\ell} + k)$ -th residual vector of the $GMRES(\leq m_{max})$ algorithm

$$\begin{cases} r_\ell = r_{\tilde{\ell}} - d_k \\ d_k = AV_k y_k \end{cases}$$

where $\tilde{\ell}$ is the iteration at which the last restart was performed and d_k is the search vector. Here V_k is the orthonormal subspace of order k and is obtained by the Arnoldi process [1]. The parameter y_k is determined in order to minimize the following least squares problem of order k :

$$\|b - Ax_\ell\|_2^2 = \|\beta e_1 - \bar{H}_k y_k\|_2^2$$

where \bar{H} is the $(k + 1) \times k$ Hessenberg matrix and $\beta = \|r_{\tilde{\ell}}\|_2$. The parameter for the

convergence test of the residual norm

$$\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k) = \frac{|(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)|}{\|\mathbf{r}_{\bar{\ell}}\|_2 \|\mathbf{d}_k\|_2} \quad (3.1)$$

$$= \sqrt{1 - \frac{\|\mathbf{r}_{\ell}\|_2^2}{\|\mathbf{r}_{\bar{\ell}}\|_2^2}} \quad (3.2)$$

is defined. The formulae (3.1) and (3.2) show the inner product of $\mathbf{r}_{\bar{\ell}}$ and \mathbf{d}_k . Since $\|\mathbf{r}_{\ell}\|_2$ and $\|\mathbf{r}_{\bar{\ell}}\|_2$ are often computed during the iterations, we use formula (3.2). The residual norm converges more rapidly as the value of $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ increases. We consider that the convergence of the residual norm does not stagnate if the formula

$$\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k) > \varepsilon \quad (3.3)$$

is satisfied. Even if Condition 1 is not satisfied, we perform the restart when formula (3.3) is satisfied. Unfortunately, since it is impossible to choose ε mathematically, we will consider varying ε adaptively during the iterations using the results of the restart determined by Condition 1. The details are presented in Subsection 3.3.

3.3. A new approach to determine the restart We apply the convergence test of the residual norm to the GMRES($\leq m_{\max}$) algorithm. This new technique determines the restart cycle based on Condition 1 and formula (3.3). The restart is performed when either Condition 1 or formula (3.3) is satisfied.

We determine ε used in formula (3.3) adaptively. Since the determination of the restart is carried out at every even step, $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is also computed at each of these steps. If the restart occurs according to Condition 1, $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ computed at this step is assigned to ε . At the next even step, $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is recomputed. If the latter $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is greater than the former one, we consider that the previous restart was performed effectively. In this case at least, there is the result that the restart was carried out according to Condition 1, when the former $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ was computed. If formula (3.3) is satisfied when the latter $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is computed, it is evident that the convergence of the residual norm is improved. Therefore, we also perform the restart when the latter $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is larger than the former one. We notice that ε is not updated if the restart is carried out according to formula (3.3).

In summary, we determine the restart in the following sequence.

Step 1 When the number of iterations is even, k approximate zeros and the current $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ are computed after the Arnoldi process. The current $\cos_{\ell}(\mathbf{r}_{\bar{\ell}}, \mathbf{d}_k)$ is assigned to ρ .

Step 2 The determination of the restart is carried out.

Step 2.1 If Condition 1 is satisfied or the total number of iterations is two, the restart is performed, setting $\varepsilon = \rho$.

Step 2.2 If formula (3.3) is satisfied or the restart cycle is m_{\max} , the restart is performed without setting $\varepsilon = \rho$.

Just like the GMRES($\leq m_{\max}$) algorithm, the determination of the restart is carried out at each even step.

In Figure 3, we show the new GMRES($\leq m_{\max}$) algorithm with the convergence test of the residual norm. The determination of the restart is carried out after the approximate solution and the residual vector are updated. The notation of “Fix k approximate zeros $\lambda_{s_2}^{(i+1)}$ ” means that the values of k approximate zeros $\lambda_{s_2}^{(i+1)}$ are fixed and are never varied later. In this paper, the GMRES($\leq m_{\max}$) algorithm with the adaptive restarting process based on these two criteria is referred to as the BC-GMRES($\leq m_{\max}$) algorithm, where “BC” means “Bi-Conditions”.

BC-GMRES($\leq m_{\max}$) algorithm

Choose x_0
 $r_0 = b - Ax_0$,
 $\ell = 0, \tilde{\ell} = 0, k = 1$
start:
 $\ell = \ell + 1$
Compute V_k through the Arnoldi process.
Update x_ℓ, r_ℓ and d_k .
if $\|r_\ell\|_2$ is sufficiently small **then**
 Stop iterations
endif
 ((Beginning of the determination of the restart))
if $(k \bmod 2) = 0$ **then**
 Compute k approximate zeros $\lambda_{s_2}^{(i+1)}$ and $\cos_\ell(r_\ell, d_k)$.
 Set $\rho = \cos_\ell(r_\ell, d_k)$
 if “Condition 1” is satisfied or $\ell = 2$ **then**
 Fix k approximate zeros $\lambda_{s_2}^{(i+1)}$
 $k = 1, \tilde{\ell} = \ell, x_{\tilde{\ell}} = x_\ell, r_{\tilde{\ell}} = r_\ell, \varepsilon = \rho$
 goto start
 endif
 if condition (3.3) is satisfied or $k = m_{\max}$ **then**
 Fix k approximate zeros $\lambda_{s_2}^{(i+1)}$
 $k = 1, \tilde{\ell} = \ell, x_{\tilde{\ell}} = x_\ell, r_{\tilde{\ell}} = r_\ell$
 goto start
 endif
 ((End of the determination of the restart))
 $k = k + 1$
goto start

FIGURE 3. The BC-GMRES($\leq m_{\max}$) algorithm.

4. Numerical Examples

We solve the linear system (1.1) in order to compare the performance of the BC-GMRES($\leq m_{\max}$) algorithm with the GMRES($\leq m_{\max}$) and GMRES(m) algorithms. There are two numerical examples in this section.

4.1. Examples for a PDE boundary value problem Two examples for a PDE boundary value problem are implemented on a Compaq Beowulf computer with a WS cluster system. The specifications of this system are as follows.

- **CPU:** Alpha 600 MHz
- **Local main memory per CPU:** 1 GB
- **OS:** Alpha Linux
- **Floating point:** Double precision arithmetic (64 bits)
- **The communication library:** MPI (Message Passing Interface)

We used 16 CPUs for these two numerical examples. In both examples, the initial approximate solution is chosen as $x_0 = 0$, and the stopping criterion for solving the linear system is

$$\frac{\|r_\epsilon\|_2}{\|b\|_2} < 1.0 \times 10^{-12}. \quad (4.1)$$

The iterations are terminated as soon as stopping criterion (4.1) is satisfied.

EXAMPLE 1. We consider the following two-dimensional problem in the region $\Omega = [0, 1]^2$ (see Joubert [6]):

$$\begin{aligned} -u_{xx} - u_{yy} + D \{ (y - 1/2)u_x + (x - 2/3)(x - 1/3)u_y \} &= f \quad \text{on } \Omega \\ u(x, y)|_{\partial\Omega} &= 1 + xy, \end{aligned}$$

where f is determined so that the exact solution is $u = 1 + xy$. We discretize this problem using the five-point central differential scheme with 512×512 grid points to produce a linear system of order 262,144. In Table 1, we list the numerical results for four kinds of values of Dh , where h is the length between two grid points and $h = 1/513$. The notation "time" and "iter" in Table 1 mean the computation time and iterations required for satisfying stopping criterion (4.1), respectively. In all cases, the residual norm of the BC-GMRES($\leq m_{\max}$) algorithm converges more rapidly than that of the other two algorithms. In most of the cases, the computation time required by the GMRES($\leq m_{\max}$) algorithm is less than 50% of that required by the GMRES(m) algorithm.

With regard to $Dh = 2^{-5}$ and $m, m_{\max} = 30$, Figure 4 plots the behaviour of the residual norm and ϵ . In Figure 4(a), the residual norm of the BC-GMRES(≤ 30)

TABLE 1. Numerical results in Example 1 (time: computation time (s), iter: iterations). An entry ... means that the stopping criterion (4.1) could not be satisfied in one hour.

Algorithm	Dh							
	2^{-5}		2^{-4}		2^{-3}		2^{-2}	
	time	iter	time	iter	time	iter	time	iter
GMRES(10)
GMRES(20)	2162.0	41150	1365.0	26890	1232.0	24449	1069.0	21255
GMRES(30)	1546.0	21478	1088.0	15510	1490.0	20181	1014.0	14034
GMRES(40)	1297.0	14666	1316.0	14272	1112.0	12562	921.0	10460
GMRES(50)	2044.0	18521	1160.0	10335	1107.0	9866	1203.0	10213
GMRES(≤ 10)	549.0	16351	557.0	16455	414.0	12751	625.0	18267
GMRES(≤ 20)	669.0	12837	737.0	14158	719.0	13411	822.0	14675
GMRES(≤ 30)	690.0	9745	601.0	8609	663.0	9635	788.0	10921
GMRES(≤ 40)	581.0	7803	634.0	8553	731.0	9651	716.0	9453
GMRES(≤ 50)	630.0	7717	723.0	9056	757.0	9449	836.0	10288
BC-GMRES(≤ 10)	305.0	12345	271.0	11040	275.0	11088	271.0	11676
BC-GMRES(≤ 20)	232.0	8497	300.0	11254	231.0	8975	254.0	10626
BC-GMRES(≤ 30)	267.0	9843	302.0	11254	248.0	9764	223.0	9446
BC-GMRES(≤ 40)	229.0	7958	297.0	11254	254.0	9764	226.0	9446
BC-GMRES(≤ 50)	231.0	7958	298.0	11254	252.0	9764	227.0	9446

algorithm converges about twice and five times as fast as the GMRES(≤ 30) and GMRES(30) algorithms, respectively. In Figure 4(b), ε is varied adaptively from 1.0×10^{-2} to 1.0×10^0 and ε does not have to be determined by the user. For the same case, we also show the number of occurrences of the restart cycle in the GMRES(≤ 30) and BC-GMRES(≤ 30) algorithms in Table 2. While the compulsory restarts with m_{\max} are never performed in the BC-GMRES(≤ 30) algorithm, about 50% of the restart cycles reach m_{\max} in the GMRES(≤ 30) algorithm. The restarts in the GMRES(≤ 30) algorithm are not always performed effectively.

4.2. An example from the Matrix Market series In this subsection, an example from the Matrix Market series [7] is implemented on a PC with a single processor. This example need not be parallelized because the order of the linear system is not as large as that in the previous example. The specifications of this PC are as follows.

- **Processor:** Pentium4 2.66GHz
- **Main memory:** 512MB
- **OS:** RedHat linux 9
- **Floating point:** Double precision arithmetic (64 bits)

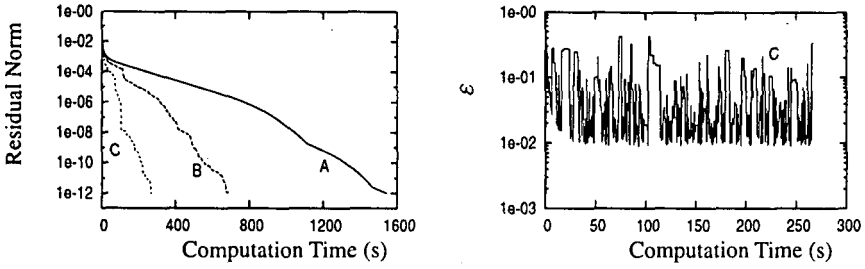


FIGURE 4. Example 1: Behaviours of the residual norm and ϵ ($Dh = 2^{-5}$). A: GMRES(30), B: GMRES(≤ 30), C: BC-GMRES(≤ 30).

TABLE 2. Classification of the frequency in use for the two restarting criteria in Example 1 ($Dh = 2^{-5}$), A: Restart based on the distribution of zeros, B: Restart based on the convergence test of the residual norm, C: Compulsory restart with m_{max} .

Restart cycle	Occurrences					
	BC-GMRES(≤ 30)			GMRES(≤ 30)		
	A	B	C	A	B	C
2	883	703	-	168	-	-
4	23	918	-	74	-	-
6	2	115	-	10	-	-
8	0	74	-	20	-	-
10	0	29	-	2	-	-
12	0	35	-	3	-	-
14	0	21	-	0	-	-
16	0	9	-	0	-	-
18	0	4	-	0	-	-
20	0	7	-	0	-	-
22	0	2	-	0	-	-
24	0	0	-	1	-	-
26	0	0	-	0	-	-
28	0	1	-	2	-	-
30	0	2	4	73	-	218

EXAMPLE 2. We now consider the linear system with the coefficient matrix “MEM-PLUS”, one of the Matrix Market series. The order of the coefficient matrix is 17,758 and its nonzero entries are 126,150. The right-hand side of the linear system is defined such that all the entries of the exact solution are 1.0. We show the computation time and iterations required to satisfy stopping criterion (4.1) in Table 3.

The convergence of the GMRES(10) algorithm is not possible for iterations that are of the order of the coefficient matrix. The residual norm of the BC-GMRES($\leq m_{max}$)

TABLE 3. Classification of the frequency in use for the two restarting criteria in Example 2.

A: Restart based on the distribution of zeros, B: Restart based on the convergence test of the residual norm, C: Compulsory restart with m_{\max} .

Restart cycle	occurrences					
	BC-GMRES(≤ 40)			GMRES(≤ 40)		
	A	B	C	A	B	C
2	91	174	-	36	-	-
4	1	275	-	13	-	-
6	0	31	-	1	-	-
8	0	15	-	3	-	-
10	0	10	-	0	-	-
12	0	5	-	0	-	-
14	0	7	-	0	-	-
16	0	13	-	0	-	-
18	0	15	-	1	-	-
20	0	10	-	1	-	-
22	0	9	-	2	-	-
24	1	1	-	0	-	-
26	1	0	-	1	-	-
28	0	6	-	1	-	-
30	2	4	-	0	-	-
32	1	8	-	1	-	-
34	1	1	-	0	-	-
36	1	2	-	0	-	-
38	0	1	-	0	-	-
40	0	1	0	0	-	74

algorithm converges more rapidly than in the two other algorithms. While the required computation time of the GMRES(≤ 40) algorithm is about 50% of that of the GMRES(40) algorithm, the computation time of the BC-GMRES(≤ 40) algorithm is only about 30% of that of the GMRES(40) algorithm. The BC-GMRES(≤ 40) algorithm requires a larger number of iterations than the GMRES(≤ 40) algorithm. However, the former involves a comparatively lower computation cost. In this context, we analyze the occurrences of the restart cycle in Table 3. The BC-GMRES(≤ 40) algorithm generally uses less than the restart cycle of $m = 20$ and the maximum restart cycle m_{\max} is used only once. On the other hand, in the GMRES(≤ 40) algorithm, the restart cycle $m = 40$ is used more frequently than the other values. Therefore, despite the lower number of iterations, the orthogonalization of the GMRES(≤ 40) algorithm is more expensive than for the BC-GMRES(≤ 40) algorithm. The behaviours of the residual norm and ε for the GMRES(40), GMRES(≤ 40) and BC-GMRES(≤ 40) algorithms

are shown in Figure 5.

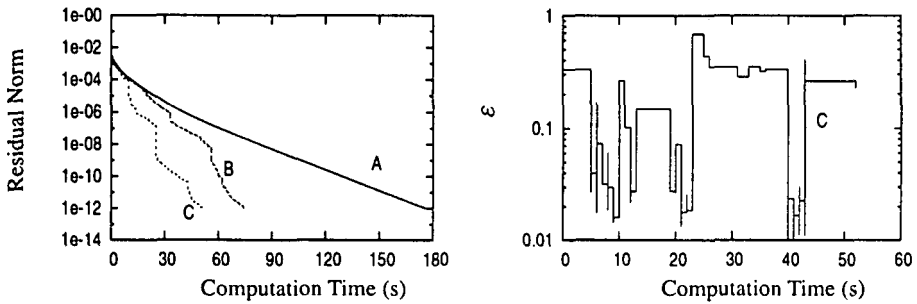


FIGURE 5. Example 2: Behaviours of the residual norm and ε , A: GMRES(40), B: GMRES(≤ 40), C: BC-GMRES(≤ 40).

The first 30 seconds of the computations of the GMRES(40) and GMRES(≤ 40) algorithms are identical, and the speedup of the GMRES(≤ 40) algorithm occurred after this interval. The convergence of the BC-GMRES(≤ 40) algorithm is almost identical to that of the GMRES(40) algorithm during the first 15 seconds. After this interval, the residual norm of the BC-GMRES(≤ 40) algorithm converges faster than that of the GMRES(40) algorithm.

5. Conclusions

We have proposed the BC-GMRES($\leq m_{\max}$) algorithm. The proposed technique applies the convergence test of the residual norm to the original GMRES($\leq m_{\max}$) algorithm. We have also analyzed its performance, using two numerical examples in Section 4. In most of the cases considered in the two examples, the residual norm converges more rapidly in the BC-GMRES($\leq m_{\max}$) algorithm than in the GMRES(m) and GMRES($\leq m_{\max}$) algorithms. We have also shown that the proposed technique reduces the number of restarts with $m = m_{\max}$, and a shorter restart becomes possible. Therefore, the BC-GMRES($\leq m_{\max}$) algorithm can be considered as an alternative to the original GMRES($\leq m_{\max}$) algorithm in some boundary value problems for partial differential equations.

References

- [1] W. Arnoldi, "The principle of minimized iterations in the solution of the matrix eigenvalue problem", *Quart. Appl. Math.* **9** (1951) 17–29.
- [2] J. Baglama, D. Calvetti, G. H. Golub and L. Reichel, "Adaptively preconditioned GMRES algorithms", *SIAM J. Sci. Comput.* **20** (1998) 373–390.
- [3] A. Chapman and Y. Saad, "Deflated and augment Krylov subspace techniques", *Numer. Lin. Algebra Appl.* **4** (1997) 43–66.
- [4] M. Eiermann, O. G. Ernst and O. Schneider, "Analysis of acceleration strategies for restarted minimal residual methods", *J. Comput. Appl. Math.* **123** (2000) 261–292.
- [5] R. W. Freund, "Quasi-kernel polynomials and their use in non-Hermitian matrix iterations", *J. Comp. Appl. Math.* **43** (1992) 135–158.
- [6] W. Joubert, "Lanczos methods for the solution of nonsymmetric systems of linear equations", *SIAM J. Matrix. Anal. Appl.* **13** (1992) 928–943.
- [7] MatrixMarket, "Url = <http://math.nist.gov/matrixmarket/>".
- [8] R. B. Morgan, "A restarted GMRES method augmented with eigenvectors", *SIAM J. Matrix Anal. Appl.* **16** (1995) 1154–1171.
- [9] R. B. Morgan, "Implicitly restarted GMRES and Arnoldi methods for nonsymmetric systems of equations", *SIAM J. Matrix Anal. Appl.* **21** (2000) 1112–1135.
- [10] K. Moriya and T. Nodera, "The DEFLATED-GMRES(m, k) method with switching the restart frequency dynamically", *Numer. Lin. Algebra Appl.* **7** (2000) 569–584.
- [11] N. M. Nachtigal, L. Reichel and L. N. Trefethen, "A hybrid GMRES algorithm for nonsymmetric linear systems", *SIAM J. Matrix. Anal. Appl.* **13** (1992) 796–825.
- [12] Y. Saad and M. K. Schultz, "GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems", *SIAM J. Sci. Stat. Comput.* **7** (1986) 856–869.
- [13] N. Tsuno and T. Nodera, "The speedup of the GMRES(m) method using the early restarting procedure", *Trans. Inform. Process. Soc. Japan* **40** (1999) 1760–1773, (in Japanese).
- [14] J. Zítko, "Using successive approximations for improving the convergence of GMRES method", *Appl. Math.* **43** (1998) 321–350.
- [15] J. Zítko, "Convergence condition for restarted GMRES method augmented with eigenspaces", *Numer. Lin. Algebra Appl.* **12** (2005) 373–390.