International Journal of Computer Mathematics
Vol. 87, No. 9, July 2010, 2101–2112

Accelerate weighted GMRES by augmenting error approximations

Qiang Niu[a]*, Linzhang Lu[a,b] and Jituan Zhou[c]

[a]*School of Mathematical Sciences, Xiamen University, Xiamen, 361005, China;* [b]*School of Mathematics and Computer Science, Guizhou Normal University, Guiyang, 550001, China;* [c]*Department of Mathematics and Physics, Wuyi University, Jiangmen, 529020, China*

(*Received 05 October 2007; revised version received 16 March 2008; accepted 13 November 2008*)

By augmenting error approximations at every restart cycle, this paper presents an accelerating strategy for restarted weighted generalized minimum residual (GMRES) method. We show that the procedure can effectively correct the occurrence of small skip D-angles, which indicates a slow convergent phase. Numerical results show that the new method converges much regular and faster than the weighted GMRES method. Finally, comparisons are made between the new and the recently proposed LGMRES methods.

Keywords: linear systems; iterative method; Arnoldi process; GMRES; WGMRES

2000 AMS Subject Classification: 65F10

C.R. Category: G.1.3

1. Introduction

We consider a large sparse linear system

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

where $A \in \mathcal{R}^{n\times n}$ is non-singular, and $b \in \mathcal{R}^n$ is the right-hand side. The generalized minimum residual (GMRES) method [13] is one of the most efficient large sparse linear system solvers. It has been widely studied, see e.g. [3,6,8,11], and some accelerating strategies have been widely discussed in [2,6,9,10,12,14]. We refer the readers to [15] for a survey of recent development of Krylov subspace linear system solvers.

By defining a different inner product in the Arnoldi process, Essai [7] presented two new iterative methods named weighted FOM (WFOM) and weighted GMRES (WGMRES) for solving non-symmetric linear systems. Cao and Yu [4] discussed the convergence behaviour of the preconditioned WFOM and WGMRES methods by numerical examples, and pointed out that

*Corresponding author. Email: kangniu@gmail.com

ISSN 0020-7160 print/ISSN 1029-0265 online

DOI: 10.1080/00207160802617004
http://www.informaworld.com

the preconditioned WFOM and WGMRES methods usually do not have a good performance compared with the preconditioned FOM and GMRES methods. In this paper, we intend to accelerate the convergence of WGMRES method, by using an augmenting strategy recently proposed by Baker *et al.* [2].

By defining a D-angle, we observed that the sequential D-angles, i.e. the angles between the successive residual vectors can be kept at a reasonable large degree. However, skip D-angles of WGMRES method, i.e. the D-angles between every other residual vectors, can become relatively small. The same phenomenon also appeared in the GMRES method, which enlightened the authors of [2] to retain the error approximations for subsequent cycles.

Generally speaking, the occurrence of small skip angles indicates that the slow convergence phase may be encountered in the iterative process. Then, by adding some previous error approximations into the next approximate Krylov subspace, we reveal that both the skip and sequential D-angles can be maintained at a relatively large degree, which implies that the convergence rate is maintained at a reasonable level. Numerical results show that the new method is much faster than WGMRES.

It is an interesting job to compare the new method discussed in this paper with the LGMRES method [2]. We compared both methods by numerical tests. Most of the results show that there is no big difference between the two methods. Examples are given to show that either method can be better than another one. We point out the potential merits of the new method and give some illuminations for future improvements.

Throughout the paper, we denote by $\mathcal{K}_m(A, v_1) = \text{span}\{v_1, Av_1, \ldots, A^{m-1}v_1\}$ the m-dimensional Krylov subspace generated by A and v_1, f^T denotes the transpose of vector f. By $\mathcal{R}^m$, we denote the m-dimensional real space, and by e_m we denote the mth coordinate vector of dimension m. Norm $\|\cdot\|_2$ denotes the Euclidean vector norm and the induced matrix norm. Matlab notations and commands, e.g. diag, abs and so on are used for convenience.

2. Analysis of the WGMRES(m) method

Let $D = \text{diag}(d_1, \ldots, d_n)$ be diagonal with $d_i > 0$, $i = 1, \ldots, n$. For any vector $u, v \in \mathcal{R}^n$, the D-scalar product can be defined as follows:

$$(u, v)_D = v^T Du = \sum_{i=1}^{n} d_i u_i v_i.$$

By the definition of norm, it is not difficult to see that a D-norm associated with this inner product can be defined as follows:

$$\|u\|_D = \sqrt{(u, u)_D} = \sqrt{\sum_{i=1}^{n} d_i u_i^2}.$$

Based on these definitions, the weighted Arnoldi process is proposed in [7], which can be described as follows.

Algorithm 1 m-step-weighted Arnoldi process

(1) for $j = 1 : m$ do:
(2) $\quad v = Av_j$;
(3) $\quad$ for $i = 1 : j$ do:
(4) $\quad\quad h_{i,j} = (v, v_i)_D$;

(5) $v = v - h_{i,j}v_i$;
(6) end
(7) $h_{j+1,j} = \|v\|_D$;
(8) $v_{j+1} = v/h_{j+1,j}$;
(9) end

Let $V_m = [v_1, \ldots, v_m]$, then Algorithm 1 reveals that

$$AV_m = V_m H_m + h_{m+1,m} v_{m+1} e_m^T = V_{m+1}\tilde{H}_m, \tag{2}$$

where $\tilde{H}_m$ is the $(m+1) \times m$ upper Hessenberg matrix, which is the same with H_m except for an additional row whose only non-zero entry is $h_{m+1,m}$ in the position $(m+1, m)$.

It is easy to show that columns of V_m form D-orthogonal basis for the Krylov subspace $\mathcal{K}_m(A, v_1) = \text{span}\{v_1, Av_1, \ldots, A^{m-1}v_1\}$, in the sense that

$$V_m^T D V_m = I_m.$$

At the $(k+1)$th cycle of the iteration, the WGMRES(m) seeks the approximation x_{k+1}, which minimizes the residual D-norm in $x_k + \mathcal{K}_m(A, r_k)$, i.e.

$$x_{k+1} = arg \min_{x \in x_k + \mathcal{K}_m(A, r_k)} \|b - Ax\|_D, \tag{3}$$

which is equivalent to

$$b - Ax_{k+1} \perp_D A\mathcal{K}_m(A, r_k). \tag{4}$$

Based on this m-step-weighted Arnoldi process, the restarted WGMRES method (WGMRES(m)) can be defined as the follows.

Algorithm 2 The WGMRES(m) algorithm

(1) Start: *Choose a stopping criterion ϵ and an initial guess x_0, compute $r_0 = b - Ax_0$, set the size of the Krylov subspace to be m.*
(2) Initialization: *Choose a diagonal matrix D with non-negative diagonal entries, compute $\beta = \|r_0\|_D$ and set $v_1 = r_0/\beta$.*
(3) Iteration: *Run Algorithm 1 to form D-orthonormal basis for the approximate Krylov subspace.*
(4) Form the approximate solution: *Solving $y_m = \text{argmin}_{y \in \mathcal{R}^m} \|\beta e_1 - \tilde{H}_m y\|_2$, and form $x_m = x_0 + V_m y_m$.*
(5) Restart: *Compute the residual vector $r_m = b - Ax_m$, and relative residual norm $\|r_m\|_2/\|b\|_2$, if satisfied then stop, else set $x_0 = x_m$, $r_0 = r_m$ and go to step (2).*

The details on the practical implementation of this algorithm are discussed in [4,7].

Remarks

1. The diagonal elements of D in step (2) of the algorithm are heuristically chosen as $d_i = \sqrt{n}|r_0(i)|/\|r_0\|_2$ in [7]. However, Cao and Yu [4] pointed out a scaling invariant property of the WGMRES method, so that d_i can be set to be $|r_0(i)|$ without incurring any difference in the sense of exact arithmetic. We remark that when the method nearly converges, the absolute value of elements of the residual vector would become very small (smaller than the residual norm). Though both approaches are same in the sense of exact computation, the occurrence of very small entries of r_0 may make the inner products lose precision. However, little difference is observed in our test.

2. The D-inner product $(u, v)_D = \sum_{i=1}^{n} d_i u_i v_i$ can be formed cheaply by using Matlab command

$$(u, v)_D = (u. * d) * v \quad \text{or} \quad (u, v)_D = u * (v. * d),$$

where d is a column vector with its ith element $d(i) = D(i, i)$. Compared with the regular inner product, one D-inner product only adds n times multiplications. Noticing that one inner product is needed for generating every element $h_{i,j}$, so the weighted Arnoldi algorithm needs $(m^2 + 3m - 1)n/2$ more flops than the regular Arnoldi algorithm. However, this difference is negligible compared with the entire orthogonalization costs of the Arnoldi process, which is $O(mn^2)$, because $m \ll n$ typically. The added storage cost is only one vector d, which is used to form the new inner products. So the additional storage is marginal compared with the overall storage.

3. In step (4) of the previous algorithm, the relative norm employed is two-norm, which is used to check the convergence.

In the next paragraph, we analyse the convergence behaviour of the WGMRES(m) method, and introduce a way to improve the convergence. First, we define the D-angle spanned by two vectors, u and v

$$\cos \angle_D(u, v) = \frac{(u, v)_D}{\|u\|_D \|v\|_D}. \tag{5}$$

Then the following theorems reveal that the convergence of WGMRES(m) is related to the D-angle of the sequential and every other residual vectors. The theorem is the natural generalization of the results in [2]. The proofs are similar, except replacing the regular inner product and norm by the D-inner product and D-norm, and so we omitted them here.

THEOREM 1 *Let r_{k+1} and r_k be the $(k+1)$-th and the k-th residual vectors from WGMRES. Then the angle between the two residual vectors is given by*

$$\cos \angle_D(r_{k+1}, r_k) = \frac{\|r_{k+1}\|_D}{\|r_k\|_D}, \tag{6}$$

where $D = \text{diag}(\text{abs}(r_k))$.

THEOREM 2 *Let r_{k+1} and r_{k-1} be the $(k+1)$th and the $(k-1)$th residual vectors from WGMRES. Then the angle between the two residual vectors is given by*

$$\cos \angle_D(r_{k+1}, r_{k-1}) = \frac{\|r_{k+1}\|_D}{\|r_{k-1}\|_D} - \frac{(Az_k, r_k - Az_{k+1})_D}{\|r_{k+1}\|_D \|r_{k-1}\|_D}, \tag{7}$$

where $D = \text{diag}(\text{abs}(r_k))$, $x_{k+1} = x_k + z_{k+1}$ and $x_k = x_{k-1} + z_k$ are the corresponding approximate solutions.

Notice that Theorem 2 is slightly different from Theorem 5 in [2]. This is due to the weighted matrix D that generally (except the case of stagnation) changes from cycle to cycle, so that for the next weighted matrix D used in the $(k+1)$th restart cycle, the D-orthogonal relationship between r_k and Az_k does not hold.

Theorem 1 reveals that the convergence of WGMRES correlates the size of the sequence angles. If the sequence residual vectors are nearly D-orthogonal to each other, then faster convergence would be expected. However, if the skip angles are small, then it indicates that the WGMRES method would build nearly the same approximate subspaces at every other restart cycle and hence

Table 1. WGMRES tested on sherman1.

k	15	20	25	30
$\angle_D(r_k, r_{k-2})$	11.1	16.3	10.1	15.4
$\angle_D(r_k, r_{k-1})$	45.5	51.2	67.2	46.1

slow convergence would appear. We will present similar theorems for the new algorithm discussed in the next section and point out the differences.

To understand the convergence behaviour of the WGMRES method, we investigated the sequence and skip D-angles by numerical experiments. The periodic alternative phenomenon of the directions of the residual vectors are observed. In Table 1, we report some tested results for WGMRES(20) on matrix sherman1 from the matrix market [1]. For different k, the skip and sequence angles, $\angle_D(r_k, r_{k-2})$ and $\angle_D(r_k, r_{k-1})$, are recorded.

3. The weighted GMRES method augmented with error approximations

Suppose $\hat{x}$ be the exact solution of Equation (1) and now we have the approximate solution x_k at hand. Then the kth error is defined by $e_k = \hat{x} - x_k$. It is well known that the linear system would be solved exactly if the next approximate subspace includes the error e_k. However, $\hat{x}$ is not explicitly known, and the best approximation to the kth error at hand is $z_k = x_k - x_{k-1}$. Based on the idea proposed in [2], we proposed to add the most recently generated p error approximations z_j, $j = k - p + 1, \ldots, k$, into the next approximation subspace of the WGMRES(m) method. Then at the $(k+1)$th cycle of the iteration, the new method would seek the approximate solution in the augmented Krylov subspace

$$\mathcal{W}_m = \mathcal{K}_{m-p}(A, r_k) \cup \{z_j\}_{j=k-p+1}^{k} = \text{span}\{r_k, Ar_k, \ldots, A^{m-p-1}r_k, z_{k-p+1}, \ldots, z_k\}, \tag{8}$$

i.e. the $(k+1)$th approximate solution has the form

$$x_{k+1} = x_k + q_{m-p-1}^{k+1}(A)r_k + \sum_{j=k-p+1}^{k} \alpha_{ij} z_j,$$

and can be formed by satisfying the following optimal property

$$x_{k+1} = \arg \min_{x \in x_k + \mathcal{W}_m} \|b - Ax\|_D, \tag{9}$$

which is equivalent to

$$b - Ax_{k+1} \perp_D A\mathcal{W}_m, \tag{10}$$

where q_{m-p-1}^{k+1} is a polynomial of degree at most $m - p - 1$, and it is to be determined together with α_{ij} such that the norm of r_{k+1} is minimized. We refer to this procedure as WGMRES-E(m, p), where (m, p) means the dimension of the approximate subspace is m with p error approximations added and restarted. The version of the new method can be described as follows.

Algorithm 3 The WGMRES-E(m, p) algorithm

(1) Start: *Given r_k, choose the weighted matrix D, compute $\beta = \|r_k\|_D$, $v_1 = r_k/\beta$, set the dimension of the approximate subspace to be m.*

(2) Orthogonalizing process: *The same as Algorithm* 1, *except changing the second step as: if* $j \leq m - p$, *then* $w = Av_j$, *else* $w = Az_{j-m+k}$.
(3) Form the approximate solution: *Find* $\tilde{y}_{k+1} = \arg\min_{y \in \mathcal{R}^m} \|\beta e_1 - \tilde{H}_m y\|_2$, *form* $x_{k+1} = x_k + W\tilde{y}_{k+1}$; *where* $W = [v_1, \cdots, v_{m-p}, z_k, \cdots, z_{k-p+1}]$, *elements of* $\tilde{H}_m$ *are generated by the orthogonalization process in Algorithm 1.*
(4) *Form* $z_{k+1} = W\tilde{y}_{k+1}$.
(5) *Compute* $r_{k+1} = b - Ax_k$, *if* $\|r_{k+1}\|_2/\|b\|_2 <$ tol, *then stop; else* $k = k + 1$, *goto step 1.*

The optimal values for p are typically very small, generally $p \leq 3$. We refer to [2] for some practical implementation details. The following results indicate that the convergence of WGMRES-E correlates both the sequential and skip D-angles.

THEOREM 3 *Let* r_{k+1}, r_k, *and* r_{k-1} *be the residual vectors from WGMRES-E. Then the angles between these residual vectors are given by*

$$\cos \angle_D(r_{k+1}, r_k) = \frac{\|r_{k+1}\|_D}{\|r_k\|_D}, \tag{11}$$

$$\cos \angle_D(r_{k+1}, r_{k-1}) = \frac{\|r_{k+1}\|_D}{\|r_{k-1}\|_D}, \tag{12}$$

where $D = \text{diag}(\text{abs}(r_k))$.

Table 2 reports some D-angle information of WGMRES-E(20,1) on the same matrix sherman1. Compared with Table 1 we can see that the new method has obviously improved the skip D-angles, and the skip angles are generally larger than the sequential angles. Hence, much faster convergence behaviour is expected.

4. Numerical experiments and further discussions

In this section, WGMRES-E(m, p) method is tested and compared with the WGMRES(m) and LGMRES methods. All numerical tests have been carried out on an INTEL PENTIUM IV 2.4 GHz with main memory 256 MB and the machine precision $eps = 2.22 \times 10^{-16}$ using MATLAB 6.5 on a Windows XP-based system. In all tables of this paper, m is the dimension of the Krylov subspace, p is the approximate error added, Iter denotes the number of restarts and CPU is the CPU timings in seconds. In the following examples, unless special statement is given, we use $p = 1$, and set $D = \text{diag}(\text{abs}(r_k))$ at the $(k + 1)$th cycle of the iteration. The stopping criterion tol is 10^{-10}.

4.1 *Comparisons between WGMRES with WGMRES-E*

Example 1 The matrix `sherman1` and the right hand-side b are all from Harwell–Beoing sparse matrix collection [5]. The matrix is real unsymmetric, of size 1000 with 3750 non-zero entries,

Table 2. WGMRES-E tested on sherman1.

k	15	20	25	30
$\angle_D(r_k, r_{k-2})$	63.6	58.8	66.1	68.1
$\angle_D(r_k, r_{k-1})$	58.2	46.3	55.4	42.1

Table 3. Tested results for Example 1.

	WGMRES(m)		WGMRES-E(m, 1)	
m	Iter	CPU	Iter	CPU
10	329	4.6	151	3.1
20	96	2.6	67	2.3
30	54	2.3	43	2.2
40	38	2.5	29	2.4

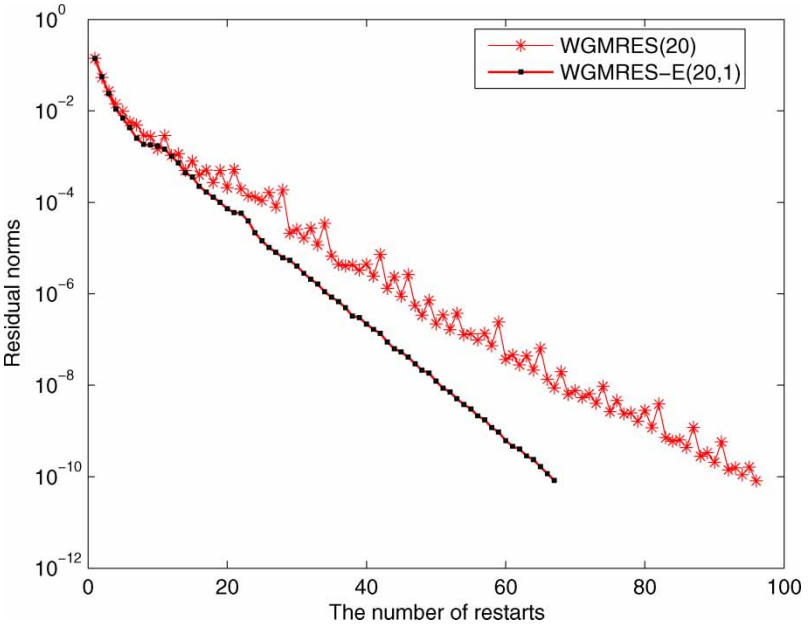

Figure 1. The convergence curves tested on matrix `sherman1`.

generated in oil reservoir simulation. In Table 3, we show the tested results for different subspace dimensions, and Figure 1 depicts the convergence curves of both methods when $m = 20$. From Figure 1 and Table 3, we can see that the WGMRES-E method converges more regular and faster than the WGMRES method. Better performance can be expected from Table 2, where the skip and sequential angles of WGMRES-E method are tested.

Example 2 The matrix `cdde1` from the matrix market [1] is tested in this example. The size of the matrix is 961, with 4681 non-zero entries. It is generated in computational fluid dynamics. Table 4 and Figure 2 show the tested results when $m = 20$.

Table 4. Tested results on Example 2.

	WGMRES(m)		WGMRES-E(m, 1)	
m	Iter	CPU	Iter	CPU
20	125	2.5	54	1.8

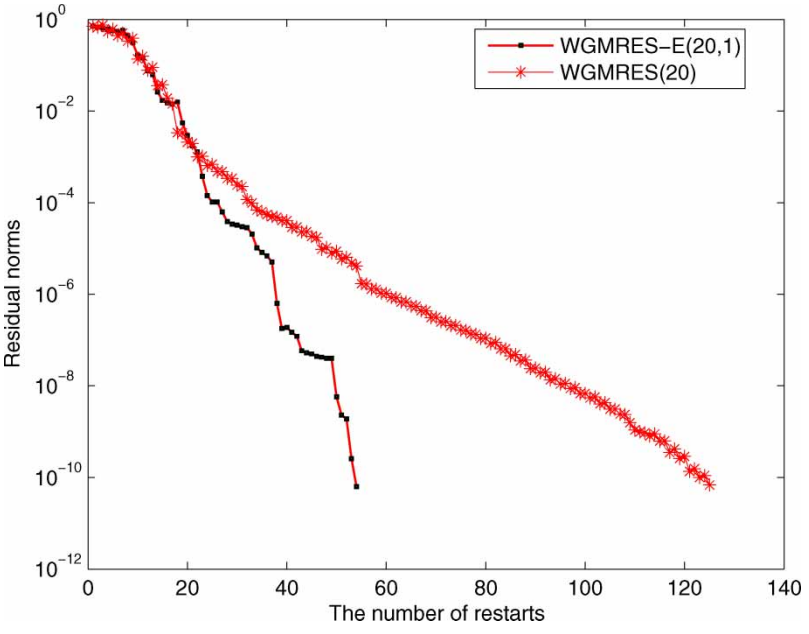

Figure 2. The convergence curves tested on matrix `cdde1`.

Table 5. Tested results for Example 3.

	WGMRES(m)		WGMRES-E(m, 1)	
m	Iter	CPU	Iter	CPU
20	168	138.5	141	126.6

Example 3 The matrix `memplus` from matrix market [1] is tested. It is of size 17,758 with 1,26,150 non-zero entries. This matrix is generated in the Computer component design memory circuit. Table 5 and Figure 3 give the convergence behaviour of both methods when $m = 20$ and $p = 1$.

From Tables 3, 4 and Figures 1–3 we have the following conclusions.

1. The irregular convergence behaviour generally appears in the WGMRES(m) method. This is different from the GMRES(m) method, in which the residual norms are non-increasing. However, this does not necessarily mean that the irregular convergence curve is bad, as sometimes WGMRES(m) can break through the stagnation phase, which may appear in the GMRES(m) method. See [7] for more detailed analysis.
2. The WGMRES-E generally converges faster than the WGMRES method in the sense that it needs less iteration numbers whereas the new method needs a little more CPU timing per iteration.
3. The convergence curve of WGMRES can be irregular, whereas the new method converges much smoothly by and regularly.

Until now, we have shown that the WGMRES-E method converges more faster and smoother than the WGMRES method. Another very interesting and meaningful task would be to compare

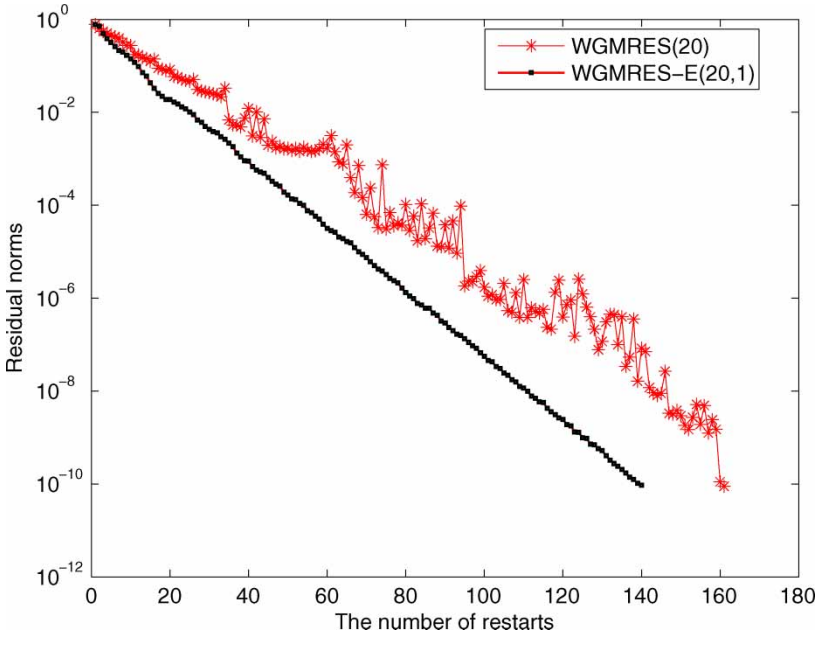

Figure 3. The convergence curves tested on matrix `memplus`.

the WGMRES-E with the LGMRES method [2] proposed by Baker *et al*. We have performed this work in the subsection.

4.2 *Comparisons between WGMRES-E and LGMRES*

In the following examples, we intend to compare the performance of WGMRES-E(m, p) with LGMRES(m, p) proposed in [2], where (m, p) in LGMRES(m, p) means the restarted LGMRES in m-dimensional approximate subspace with p error approximations added, so the same size approximate subspaces are used in both the methods.

Example 4 In this example, we use matrix `sherman4` and its corresponding right-hand side b from [5]. The matrix is of size 1104 with 3786 non-zeros. In Table 6, we show the convergence history of both the methods. From the table, we can see that when subspace dimension increases, there is a small difference between the two methods, and WGMRES-E is slower than the LGMRES method in both CPU timing and iteration numbers. Similar convergence comparisons can be found in some other matrices. However, this is not always the case. For most of the cases, there

Table 6. Tested results for Example 4.

	WGMRES(m, 1)		LGMRES(m, 1)	
m	Iter	CPU	Iter	CPU
10	44	0.48	32	0.31
20	19	0.52	18	0.45
30	13	0.89	11	0.81
40	9	1.2	7	0.95

are only small differences between the two methods. There also exist cases where WGMRES-E is considerably better than LGMRES. See the following examples.

Example 5 We tested on matrix `bfw782a` from [1] in this example. The matrix is of size 782 with 7514 non-zeros.

Example 6 We tested on matrix fs_541_2 from [1] in this example. The matrix is of size 541 with 4285 non-zeros.

In Figures 4 and 5, we record the convergence curve of two matrices, respectively. From Figure 4, we can see that the convergence behaviour of both methods are similar. Sometimes one of the methods is a little better. This phenomenon is typical from our test experience. The reason is that there is no big difference between the approximate Krylov subspaces used by the two methods, if the same starting vector is used. The merit of the WGMRES type method is that by using a different type of orthonormal basis, the method may shrink or break through the stagnation phase. Moreover, there is some flexibility of the method by changing different types of the weighted matrices, although choosing the optimal weighted matrix is a difficult job. Here, we heuristically use the weighted matrix constructed by the most recently generated residual vector. We also tested other weighted matrices at every restart. Particularly, we sometimes use the residual vector plus the error approximation to build a weighted matrix that may produce better performance, but not always. However, investigating the potential better and even nearly optimal weighted matrix at every restart is an interesting work that needs further investigation.

Specially, we also have an example (see Figure 5) where WGMRES-E is considerably better than the LGMRES method. This matrix is also tested in [7]. Compared with the FOM method, the problem `bfw782a` seems more difficult for the GMRES method, which would stagnate for low-dimension Krylov subspace; WGMRES method converges, whereas takes time

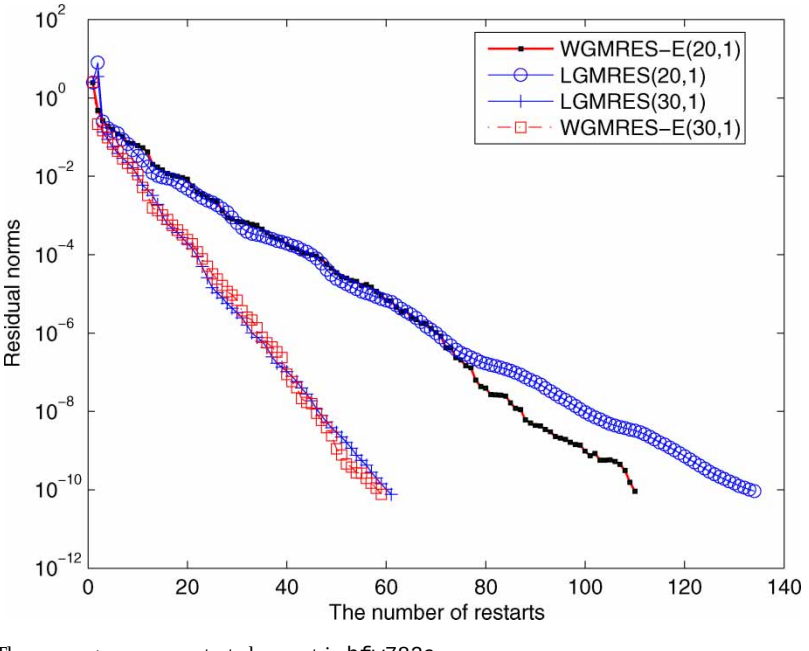

Figure 4. The convergence curves tested on matrix `bfw782a`.

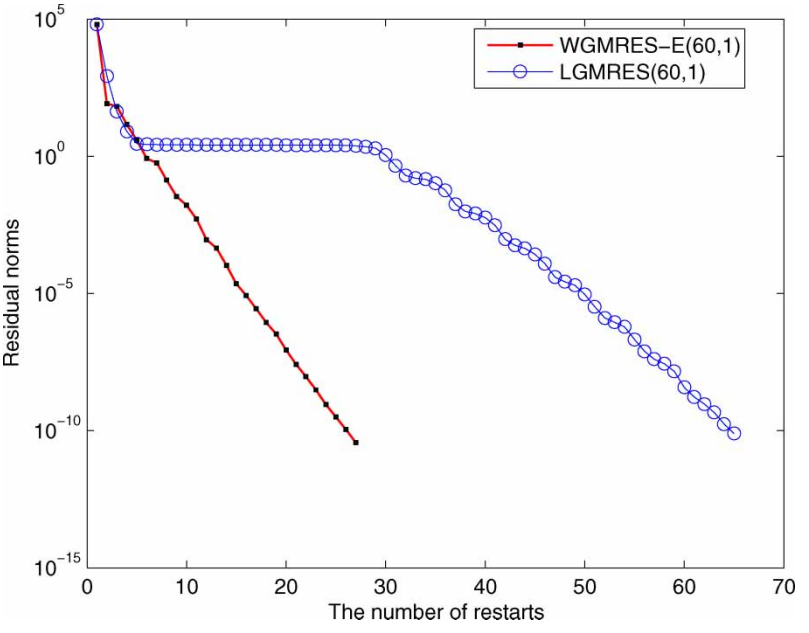

Figure 5. The convergence curves tested on matrix `fs_541_2`.

to break through the stagnation. After augmenting with error approximations, both LGMRES and WGMRES-E converge. However, a phase of stagnation still exists for the LGMRES method, whereas WGMRES-E can break through easily, and converges much faster. Although this example is not typical, still we can see there are some merits of the new method.

5. Concluding remarks

We have discussed an acceleration strategy applied to the WGMRES method, which shows that the new method converges more smoothly and quickly than the WGMRES method. Comparisons show that the new method behaves more or less compared with the LGMRES method, and sometimes the new method is comparable with the LGMRES method. Finally, we give impersonal remarks and further research lines.

Acknowledgements

The authors thank the anonymous referees for their valuable comments on how to improve the presentation. This work was supported by the National Natural Science Foundation of China, nos. 10531080. The first author was supported by the China Scholarship Council.

References

[1] Z. Bai, R. Barret, D. Day, J. Demmel, and J. Dongarra, *Test matrix collection for non-Hermitian eigenvalue problems*, Technical Report, Department of Mathematics, University of Kentucky, US, 1995. Available at http://math.nist.gov/MatrixMarket/.

[2] A. Baker, L. Jessup, and T. Manteuffel, *A technique for accelerating the convergence of restarted GMRES*, SIAM J. Matrix Anal. Appl. 26 (2005), pp. 962–984.

[3] B. Barret, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, and H.A. van der Vorst (eds.), *Templates for the Solution of Linear Systems: Building Blocks for Iterative Systems*, SIAM, Philadelphia, 1994.

[4] Z.H. Cao and X.Y. Yu, *A note on the weighted FOM and GMRES for solving nonsymmetric linear systems*, Appl. Math. Comput. 151 (2004), pp. 719–727.
[5] I.S. Duff, R.G. Grimes, and J.G. Lewis, *Sparse matrix test problems*, ACM Trans. Math. Soft. 15 (1989), pp. 1–14.
[6] M. Eiermann, O.G. Ernst, and O. Schneider, *Analysis of acceleration stategies for restarted minimal residual methods*, J. Comput. Appl. Math. 123 (2000), pp. 261–292.
[7] A. Essai, *Weighted FOM and GMRES for solving nonsymmetric linear systems*, Numer. Algor. 18 (1998), pp. 277–292.
[8] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.
[9] G. Gu, *Restarted GMRES augmented with harmonic Ritz vectors for shifted linear systems*, Int. J. Comput. Math, 82 (2005), pp. 837–849.
[10] R.B. Morgan, *A restarted GMRES method augmented with eigenvectors*, SIAM J. Matrix Anal. Appl. 16 (1995), pp. 1154–1171 .
[11] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.
[12] Y. Saad, *Analysis of augmented Krylov subspace methods*, SIAM J. Matrix Anal. Appl. 18 (1997), pp. 435–449.
[13] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J. Sci. Stat. Comput. 7 (1986), pp. 856–869.
[14] V. Simoncini and D.B. Szyld, *On the occurrence of superlinear convergence of exact and inexact Krylov subspace methods*, SIAM Rev. 47 (2005), pp. 247–272.
[15] V. Simoncini and D.B. Szyld, *Recent computational developments in Krylov subspace methods for linear systems*, Numer. Linear Algebra Appl. 14 (2007), pp. 1–59.