ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 1, No. 2, 2006, *pp*. 110-114

Simpler Hybrid GMRES

Hua-lei Liu [+]

Department of Mathematics, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China

(*Received Dec.10, 2005, accepted March. 02, 2006*)

Abstract. Hybrid GMRES algorithms are effective for solving large nonsymmetric linear systems. GMRES is employed at the first phase to produce iterative polynomials, which will be used at the second phase to implement the Richardson iteration. In the process of GMRES, a least squares problem needs to be solved which involves an upper Hessenberg factorization. Instead of using GMRES, we may use simpler GMRES. Correspondingly, simpler hybrid GMRES algorithms are formulated. It is described how to construct the iterative polynomials from simpler GMRES. The new algorithms avoid the upper Hessenberg factorization so that they are easier to program and require a less amount of work. Numerical examples are conducted to illustrate the good performance of the new algorithms.

Key words: linear systems, iterative methods, GMRES, hybrid algorithm.

1. Introduction

Suppose we are given a large nonsymmetric system of linear equations of the form

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

where A may be the matrix after preconditiong. Hybrid GMRES algorithms [1, 2] are effective methods for solving such systems. Unlike other hybrid algorithms, which first estimate eigenvalues and then apply this knowledge in further iterations, these ones avoid eigenvalues estimate. [1] first propose the hybrid GMRES algorithm, which runs GMRES until the residual norm drops by a certain amount, then re-applies the GMRES residual polynomial via a Richardson iteration with Leja ordering. A product hybrid GMRES algorithm is proposed in [2], which significantly improves the convergence of Richardson iteration. The underlying idea of the product hybrid scheme is the complementary behavior of restarted GMRES, which is first observed in [3] and has attracted wide interest at present [4, 5]. It has been observed that the residual polynomials resulted from successive restarting cycles of GMRES(m) may differ from one another meaningfully. In [2], it has further shown that these polynomials can complement each other harmoniously in reducing the iterative residual, a natural choice of the residual polynomial needed in Richardson iteration is the product of these polynomials. The hybrid GMRES algorithm [1] is sketched as follows:

Algorithm 1: Hybrid GMRES algorithm

Start with a random initial guess x_0

Phase I: Run GMRES until $\|r_n\|$ drops by a suitable amount. Set $m = n$.

Phase II: Re-apply the residual polynomial $P_m(z)$ constructed by GMRES

cyclically via a Richardson iteration with Leja ordering.

In the algorithm, GMRES [6] is taken to acquire enough information needed in Phase II. Central to the usual implementation of GMRES is the Arnoldi process. r_0 (r_0 denotes the initial residual $b - Ax_0$) is chosen as the initial vector of this process. By shifting the Arnoldi process to begin with Ar_0 instead of r_0, we obtain the simpler GMRES [7]. The sketch of restarted version of the algorithm denoted by SGMRES(m) is presented below.

Algorithm 2: Restarted simpler GMRES algorithm

[+] *E-mail address*: liuhualei99@163.com.

(1) Given x_0, compute $r_0 = b - Ax_0$ and set $\beta = \|Ar_0\|, v_1 = Ar_0$.

(2) Iterate, for $j = 1, \cdots, m$, do

- $\rho_{ij} = (Av_j, v_i) \quad i = 1, \cdots, j-1, \quad j > 1$
- $\hat{v}_j = Av_{j-1} - \sum_{i=1}^{j-1} \rho_{ij} v_i \quad j > 1$
- $\rho_{jj} = \|\hat{v}_j\|$, and $v_j = \hat{v}_j / \rho_{jj}$
- $\xi_j = r_{j-1}^T v_j$ and $r_j = r_0 - \sum_{i=1}^{j} \xi_i v_i$

The ρ_{ij}' s constitute the (possibly) nonzero elements of the matrix R_j and $V_j = (r_0, v_1, \cdots, v_{j-1})$.

- If the residual is sufficiently small, go to (3).

(3) Form the approximate solution

$$\tilde{x} = x_0 + V_j y_j$$

where $y_j = R_j^{-1} w_j$, $w_j = (\xi_1, \cdots, \xi_j)^T$.

If $\tilde{x}$ is sufficiently accurate, accept $\tilde{x}$ and exit; otherwise, $x_0 = \tilde{x}$ and go to (1).

The advantage is that the problem for finding the minimal residual solution reduces to an upper triangular least-squares problem instead of the upper-Hessenberg least-squares problem of GMRES. Because upper-Hessenberg factorization is unnecessary, it is simpler to program and requires $O(m^2)$ fewer arithmetic operations over each iteration cycle of m steps than the usual implementation of GMRES. The two algorithms have almost the same accuracy and efficiency. If we replace the GMRES in the hybrid algorithms with simpler GMRES, we obtain the simpler hybrid GMRES algorithms. In view of the better performance of the product hybrid GMRES, we compare the product version of the new algorithms with the hybrid GMRES algorithm and SGMRES(m) in Section 3.

2. Hybrid algorithms based on simpler GMRES

We first describe how to construct the residual polynomial $P_m(z)$ from m steps simpler GMRES. Let $K_m = (r_0, Ar_0, \cdots, A^{m-1} r_0)$. The m steps simpler GMRES constructs an $m \times n$ matrix of vectors spanning the same space $V_m = (r_0, v_1, \cdots, v_{m-1})$. According to [1], in order to determine the coefficients of $P_m(z)$, we need to compute the upper-triangular matrix C_m that satisfies

$$V_m = K_m C_m.$$

Denote $L_{m-1} = (Ar_0, \cdots, A^{m-1} r_0)$ and $W_{m-1} = (v_1, \cdots, v_{m-1})$. B_{m-1} satisfies

$$W_{m-1} = L_{m-1} B_{m-1}. \tag{1}$$

Obviously, $C_1 = (1)$, $B_1 = (1/\|Ar_0\|)$, and

$$C_m = \begin{pmatrix} 1 & 0 \\ 0 & B_{m-1} \end{pmatrix}.$$

Let $\rho_{m-1} = (\rho_{1,m-1}, \cdots, \rho_{m-2,m-1})^T$ and $B_i^{(j)}$ denotes the jth column of B_i. According to (1), $v_{m-1} = L_{m-1} B_{m-1}^{(m-1)}$. By applying the formula

$$v_{m-1} = Av_{m-2} - \sum_{i=1}^{m-2} \rho_{i,m-1} v_i,$$

we have

$$\rho_{m-1,m-1}^{-1} (Av_{m-2} - \sum_{i=1}^{m-2} \rho_{i,m-1} v_i) = L_{m-1} B_{m-1}^{(m-1)}.$$

Then

$$\rho_{m-1,m-1}^{-1}(AL_{m-2}B_{m-2}^{(m-2)}-W_{m-2}\rho_{m-1})=L_{m-1}B_{m-1}^{(m-1)};$$

equivalently,

$$\rho_{m-1,m-1}^{-1}(L_{m-1}\begin{pmatrix}0\\B_{m-2}^{(m-2)}\end{pmatrix}-L_{m-1}\begin{pmatrix}B_{m-2}\rho_{m-1}\\0\end{pmatrix})=L_{m-1}B_{m-1}^{(m-1)}.$$

It follows that

$$B_{m-1}^{(m-1)}=\rho_{m-1,m-1}^{-1}\begin{pmatrix}0\\B_{m-2}^{(m-2)}\end{pmatrix}-\rho_{m-1,m-1}^{-1}\begin{pmatrix}B_{m-2}\rho_{m-1}\\0\end{pmatrix}. \tag{2}$$

By inserting the calculation (2) to the simpler GMRES iteration, we generate the elements of B_{m-1} column by column as the iteration proceeds. From C_m, we can obtain the coefficients of the residual polynomial $P_m(z)$ easily according to [1].

Algorithm 3: Simpler hybrid GMRES algorithm

Phase I: Run SGMRES until $\|r_n\|$ drops by a suitable amount. Set $m=n$.

Phase II: Re-apply the simpler GMRES residual polynomial $P_m(z)$ cyclically via a Richardson iteration with Leja ordering.

If we use the restarted simpler GMRES in Phase I, we obtain a product simpler hybrid GMRES algorithm. $P_{m,k}(z)\quad(k=1,\cdots,s)$ denote the residual polynomials resulted from successive restarting cycles of SGMRES(m). s denotes the number of these residual polynomials.

Algorithm 4: Product simpler hybrid GMRES algorithm

Phase I: Run SGMRES(m) until $\|r_{km}\|$ drops by a suitable amount. Set $s=k$ and construct the simpler GMRES residual polynomials $P_{m,k}(z),\quad k=1,\cdots,s$.

Phase II: Re-apply the residual polynomial $\Pi_s(z)=P_{m,s}(z)P_{m,s-1}(z)\cdots P_{m,1}(z)$ cyclically via a Richardson iteration with Leja ordering.

The principal feature of simpler hybrid GMRES or product simpler hybrid GMRES is when Phase I should be terminated, in other words, the choice of m or m and s. Firstly, we have the following theorem:

Theorem 1: The residual polynomials constructed by m steps simpler GMRES and m steps GMRES are equivalent.

Proof. Let $P_m(z)$ and $Q_m(z)$ be the residual polynomials constructed by m steps simpler GMRES and GMRES respectively. Then for any $w\in AK_m$, we have

$$(P_m(A)r_0,w)=0 \text{ and } (Q_m(A)r_0,w)=0.$$

Hence, $((P_m(A)-Q_m(A))r_0,w)=0$ .Considering that $(P_m(A)-Q_m(A))r_0\in AK_m$ and $\deg((P_m(z)-Q_m(z))\le m$, it follows that

$$P_m(z)=Q_m(z).$$

Based on the assumptions adopted in [1], the m steps simpler GMRES is equivalent to the m steps GMRES implementation. Therefore, it is natural for us to choose the same switching criterions, see [1, 2].

3. Numerical experiments

In this section, we apply the new algorithm to some test examples and compare its performance with the hybrid GMRES algorithm (HGMRES(m)) and SGMRES(m). In view that the product version of the hybrid GMRES performs better than the usual implementation of hybrid GMRES, we just describe the numerical results of the product simpler hybrid GMRES (P-SHGMRES(s, m)). For each example, we present a plot shows $\log_{10}\|r_{sm}\|$ as a function of work measured by vector operations. In each example, the right-hand side is chosen as $b=(1,\cdots,1)^T$, the initial guess x_0 is taken to be zero and the convergence tolerance is $\varepsilon=10^{-10}$.

Example 1 This example is taken from [1]**.** A is a large upper-triangular Toeplitz matrix of the form

$$A=\begin{pmatrix} 1 & 1 & 1/2 & & \\ & 1 & 1 & \ddots & \\ & & 1 & \ddots & 1/2 \\ & & & \ddots & 1 \\ & & & & 1 \end{pmatrix}_{1000\times 1000}$$

In Figure 1, we can see that HGMRES(2) performs ideally well, but P-SHGMRES(2,2) is further ahead. SGMRES(2) lags behind the two algorithms. The observation shows the efficiency of the new algorithm.

Example 2: This is a realistic test problem taken from the Harwell-Boeing collection. The matrix(GRE115) was produced from runs of package QNAP written by CII-HB for simulation modeling of computer systems and used as a test bed for ordering codes. In this example, HGMRES(33) diverges for all possible choices of m , but P-SHGMRES(2,33) converges well.

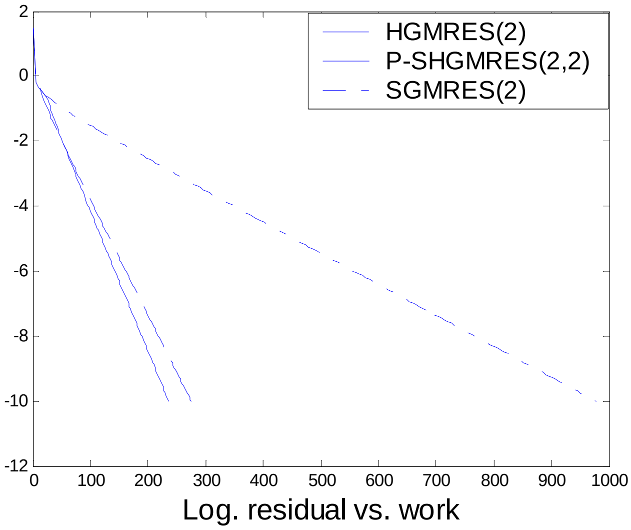

Fig. 1: Example 1

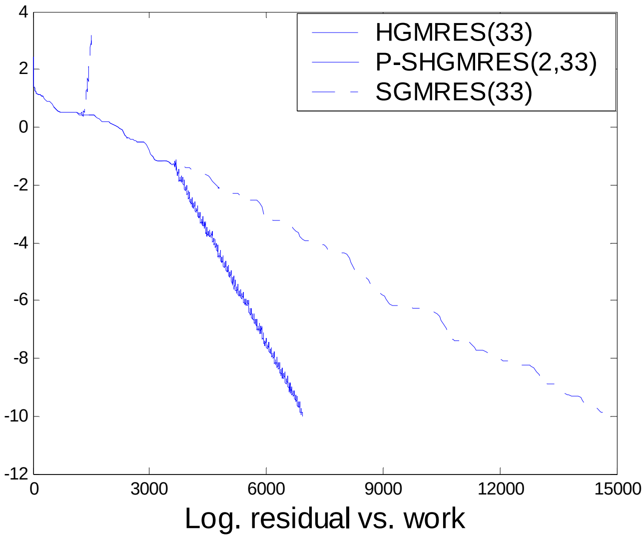

Fig. 2: Example 2

4. Conclusion

It has been shown in [4] that simpler GMRES may not maintain accuracy well near the limits of residual norm reduction. However, in our product simpler hybrid GMRES algorithm, because we use the Richardson iteration in the further steps, the accuracy is maintained well. Generally speaking, the product simpler hybrid GMRES performs better than the hybrid GMRES [1]. Furthermore, it is simpler to program.

5. Acknowledgement

The author wishes to thank Dr. Bao-jiang Zhong for many helpful comments on this subject.

6. References

[1] N. M. Nachtigal, L. Reichel, L. N. Trefethen, A hybrid GMRES algorithm for nonsymmetric linear systems, SIAM J. Matrix Anual. Appl., 13(1992), 796-825.

[2] B. J. Zhong, A product hybrid GMRES algorithm for nonsymmetric linear systems, Journal of Computational Mathematics, 23(2004), 83-92.

[3] B. J. Zhong, A flexible hybrid GMRES algorithm (in Chinese), Numer. Math. –A J. of Chinese Universities, 23(2001), 261-272.

[4] B. J. Zhong and Ronald B. Morgan., Complementary cycles of restarted GMRES, Technical report, Department of Mathematics, Nanjing University of Aeronautics and Astronautics, 2004.

[5] Alison H. Baker, Elizabeth R. Jessup, and Thomas Manteuffel, A technique for accelerating the convergence of restarted GMRES, SIAM J. Matrix Anal. Appl., 26(2005), 962-984.

[6] Y. Saad, M.H. Schults, GMRES, A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Statist. Comput., 7(1986), 856-869.

[7] H. F. Walker, L. Zhou, A simpler GMRES, Numer. Lin. Alg. Appl., 1(1994), 571-581.