New insights in GMRES-like methods with variable preconditioners

Report 93-10

Kees Vuik

TU Delft

Faculteit der Technische Wiskunde en Informatica
Faculty of Technical Mathematics and Informatics

Technische Universiteit Delft
Delft University of Technology

ISSN 0922-5641

Copies of these reports may be obtained from the bureau of the Faculty of Technical Mathematics and Informatics, Julianalaan 132, 2628 BL Delft, phone +3115784568.
A selection of these reports is available in PostScript form at the Faculty's anonymous ftp-site. They are located in the directory /pub/publications/tech-reports at ftp.twi.tudelft.nl

New insights in GMRES-like methods with variable preconditioners

Kees Vuik
Faculty of Technical Mathematics and Informatics,
Delft University of Technology,
Mekelweg 4, 2628 CD Delft
The Netherlands

Abstract

In this paper we compare two recently proposed methods, FGMRES [5] and GMRESR [7], for the iterative solution of sparse linear systems with an unsymmetric nonsingular matrix. Both methods compute minimal residual approximations using preconditioners, which may be different from step to step. The insights resulting from this comparison lead to better variants of both methods.

Keywords: FGMRES, GMRESR, non symmetric linear systems, iterative solver.

AMS(MOS) subject classification. 65F10

1 Introduction

Recently two new iterative methods, FGMRES [5] and GMRESR [7] have been proposed to solve sparse linear systems with an unsymmetric and nonsingular matrix. Both methods are based on the same idea: the use of a preconditioner, which may be different in every iteration. However, the resulting algorithms lead to somewhat different results.

In [5] the GMRES method is given for a fixed preconditioner. Thereafter, it is shown that a slightly adapted algorithm: FGMRES can be used in combination with a variable preconditioner. Finally a wide class of possible preconditioners is given.

In [7] GMRESR is presented as a slightly adapted version of the GCR method [2]. Again a variable preconditioner can be used. A special choice of the preconditioner, m steps of GMRES [6] or one LSQR step [4] , is investigated in more detail. In [9] GMRESR is compared with other iterative methods. For the given class of problems in [9] GMRESR is feasible if the matrix vector product is expensive with respect to a vector update and the number of iterations is not too large.

A short comparison of FGMRES and GMRESR has been given in [9]. The results of this comparison may be summarized as follows. FGMRES may break down, and can only be restarted in the outer loop. GMRESR does not break down and can be restarted and truncated. In general, the search directions used in both methods are different, but the convergence behaviour is approximately the same. The required amount of memory and work for a given number of iterations without restarting or truncation are comparable.

In this paper we give a more detailed comparison of FGMRES and GMRESR. We describe both methods in Section 2, and compare them in Section 3.1. In Section 3.2 we specify another method called FFOM and show that the FGMRES search directions are constructed from the FFOM residuals. This relation can be used to avoid breakdown and to stop in the inner loop. In Section 4 an FGMRES variant is given which is equal to GMRESR. In Section 5.1 the reverse is shown: a GMRESR variant, which is equal to FGMRES. Finally in Section 5.2 a cheaper implementation of GMRESR is given.

2 FGMRES and GMRESR

In this section we describe the FGMRES [5] and the GMRESR method [7]. These are iterative solution methods for the non singular linear system $Ax = b$. Furthermore we give some definitions to facilitate comparison of both methods in the following sections.

In ([5]; Algorithm 2.2) the Flexible GMRES algorithm (FGMRES) is defined as follows:

<u>FGMRES algorithm</u>

1. Start: Select x_0, tol, and compute $r_0 = b - Ax_0$,
   $\beta = \|r_0\|_2,\ v_1 = r_0/\beta$ and set $k = 0$;
2. Iterate: while $\|r_k\|_2 > \text{tol}$ do
   $k = k+1,\ z_k = M_k(v_k),\ w = Az_k;$
   for $i = 1, ..., k$ do
   $h_{i,k} = w^T v_i,\ w = w - h_{i,k} v_i;$
   $h_{k+1,k} = \|w\|_2,\ v_{k+1} = w/h_{k+1,k};$
3. Form the approximate solution:

   Define $Z_k := [z_1, ..., z_k]$ en $\bar{H}_k := \{h_{i,j}\}_{\substack{1 \le i \le j+1,\\ 1 \le j \le k,}}$

   Compute $x_k = x_0 + Z_k y_k$ where $y_k = arg \min_{y \in I\!R^k} \|\beta e_1 - \bar{H}_k y\|_2$

   and $e_1 = [1, 0, ..., 0]^T \in I\!R^{k+1}$.

In this algorithm the non-linear operator M_k is an approximation of A^{-1}. M_k can be seen as a variable preconditioner of the system $Ax = b$. Comparing GMRES and FGMRES it appears that, besides the variable preconditioner M_k, the only further change is that the search directions z_k should be kept in memory. Many relations used in GMRES can also be proved for FGMRES, for instance: the computation of y_k and the estimate of $\|r_k\|_2$ during the iteration process. In Section 3 we shall show that FGMRES and GMRES have different properties with respect to breakdown.

In [7] the GMRES Recursive algorithm (GMRESR) is proposed:

GMRESR algorithm

1. Start: select x_0, tol;
   compute $r_0 = b - Ax_0$, and set $k = 0$;
2. Iterate: while $\|r_k\|_2 > \text{tol}$ do
   - $k = k + 1,\ u_k^{(1)} = M_k(r_{k-1}),\ c_k^{(1)} = Au_k^{(1)}$;
   - for $i = 1, ..., k-1$ do
     - $\alpha_i = c_i^T c_k^{(i)},\ c_k^{(i+1)} = c_k^{(i)} - \alpha_i c_i,\ u_k^{(i+1)} = u_k^{(i)} - \alpha_i u_i$;
   - $c_k = c_k^{(k)}/\|c_k^{(k)}\|_2,\ u_k = u_k^{(k)}/\|c_k^{(k)}\|_2$;
   - $x_k = x_{k-1} + u_k c_k^T r_{k-1}$;
   - $r_k = r_{k-1} - c_k c_k^T r_{k-1}$;

Again the operator M_k is an approximation of A^{-1}.

In [7] this method is analysed for a special choice of M_k:

> The search direction $u_k^{(1)}$ is obtained as an approximation to the solution of $Ay = r_{k-1}$ using m steps of GMRES. This inner iteration is always started with $y_0 = 0$ as initial guess. In order to avoid breakdown we use an LSQR switch: if $u_k^{(1)} = 0$ then $u_k^{(1)} = A^T r_{k-1}$ (compare [9]).

In the remainder of this paper we compare FGMRES with GMRESR where both use the same choice of M_k. In order to avoid confusion, we distinguish vectors by a superscript if necessary. For instance r_k^{FG} denotes the FGMRES residual, and r_k^{GR} denotes the GMRESR residual.

3 Comparing the search directions used by FGMRES and GMRESR

3.1 Differences between FGMRES and GMRESR

In this subsection the comparison of FGMRES and GMRESR is started by choosing the operators M_k equal to a linear operator M for every k. Thereafter we show that if M_1 and M_2 are different then, in general, after the first iteration the residuals of FGMRES and GMRESR are different. Further we specify an example, where FGMRES breaks down. We end this subsection with an application of FGMRES and GMRESR to a testproblem.

In this paragraph we choose $M_k = M$ where M is a linear operator. It is easily seen [5] that for this choice FGMRES is equal to GMRES applied to

$$AM(M^{-1}x) = b, \tag{3.1}$$

and GMRESR is equal to GCR (for GCR see [2]) applied to (3.1). For a comparison of GMRES and GCR we refer to [6]. Note that for this choice the computed solutions are the same. However even for this choice there are differences between FGMRES and GMRESR, because GCR can have a breakdown in contrast with GMRES. Furthermore, GCR costs more work and memory than GMRES. An advantage for GCR is that it can be restarted and truncated whereas GMRES can only be restarted. We shall see that for variable M_k the comparison is more favourable for GMRESR.

In general the operators M_k vary from step to step and are non-linear. From the algorithms it follows that for every choice of M_1, $u_1 \in span\ \{z_1\}$ so $r_1^{FG} = r_1^{GR}$. However in the second step $z_2 = M_2(v_2)$, where v_2 is the component of Az_1 perpendicular to r_0 and $u_2^{(1)} = M_2(r_1^{GR})$, where r_1^{GR} is the component of r_0 perpendicular to $Au_1 \in span\ \{Az_1\}$. Since M_1 and M_2 are different and/or nonlinear, in general, the *span* of $\{z_1, z_2\}$ is different from the *span* of $\{u_1^{(1)}, u_2^{(1)}\}$. This is illustrated by the following examples, where always $\|r_1^{FG}\|_2 = \|r_1^{GR}\|_2$ but $\|r_2^{FG}\|_2 \neq \|r_2^{GR}\|_2$.

In the Example 1 we show that FGMRES and GMRESR have different properties with respect to breakdown. In this example FGMRES breaks down in the second iteration, whereas $\|r_2^{GR}\|_2 = 0$.

Example 1

Take $A = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$, $x = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and $x_0 = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$. Further we choose $M_1 = I$, and $M_2 = A^2$. Note that M_2 is equal to A^{-1} and A^T.

Applying FGMRES leads to:

$v_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $z_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $x_1^{FG} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ and $r_1^{FG} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, in the second step

$v_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$, $z_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $\bar{H}_2 = \begin{pmatrix} 0 & 0 \\ 1 & 1 \\ 0 & 0 \end{pmatrix}$ and v_3 is undefined. Since $x_2^{FG} = x_0 + \alpha z_1 + \beta z_2$ is follows that $x_2^{FG} \neq x$, so this is a serious breakdown.

Applying GMRESR we obtain:

$u_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $c_1 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$, $x_1^{GR} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$ and $r_1^{GR} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, in the second step

$u_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$, $c_2 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, $x_2^{GR} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$ and $r_2^{GR} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$. So GMRESR has computed the exact solution after two iterations.

Finally we apply FGMRES and GMRESR to a linear system obtained from a discretization of the following pde:

$$-\left(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}\right)+\beta\left(\frac{\partial u}{\partial x}+\frac{\partial u}{\partial y}\right)=f \text{ on } \Omega,$$

$$u|_{\partial\Omega}=0,$$

where Ω is the unit square. The exact solution u is given by $u(x,y)=\sin(\pi x)\sin(\pi y)$. In the discretization we use the standard five point finite difference approximation. The stepsizes in x- and y-direction are equal to h. As innerloop we take one step of GMRES(10) in both methods. The results for $\beta = 1$ and $h = 1/50$ are given in Figure 3.1. As expected only $\|r_0\|_2$ and $\|r_1\|_2$ are the same for both methods. Note that the convergence behaviour is approximately the same.

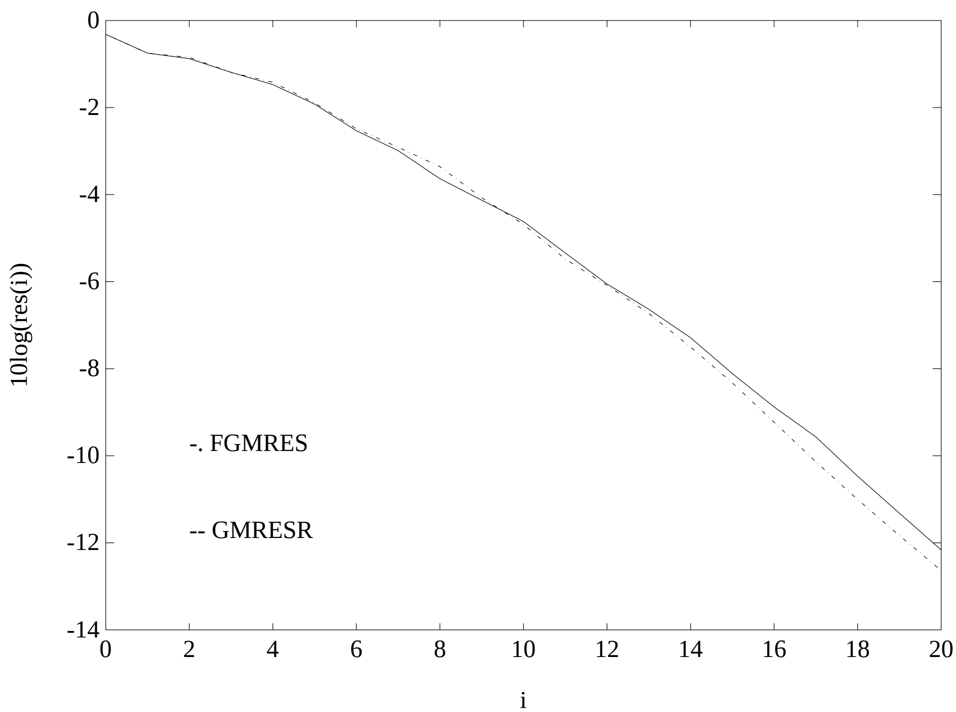

Figure 3.1: The norm of the residuals for $\beta = 1$ and $h = 1/50$

Conclusions

We have seen that if the operators M_k are all equal to the same linear operator M, then FGMRES is equal to GMRES and GMRESR is equal to GCR. In this case the computed solutions are the same but GCR may have a breakdown and is more expensive than GMRES with respect to work and memory.

In the general case, M_k variable and nonlinear, the results are different for $k \geq 2$. Moreover the costs of FGMRES and GMRESR are approximately the same. From the given example it appears that FGMRES breaks down, whereas GMRESR converges. In [7] it is proved that GMRESR (with LSQR switch) does not breakdown. So for variable M_k the comparison is more favourable for GMRESR.

3.2 The search directions of FGMRES are constructed from FFOM residuals

In this subsection we define the FFOM method. The relation with FGMRES is comparable with the relation between FOM [6] and GMRES (see [6], [1], [8], and [3]). It appears that the vector v_{k+1} is equal to a constant times the k-th FFOM residual. This relation gives us a better insight in the FGMRES method and the differences with the GMRESR method. These insights are used to avoid breakdown of FGMRES and to determine a termination criterion for the innerloop iteration such that the required accuracy is obtained.

Below we describe the FFOM method. Vectors related to the FFOM method are denoted by a superscript FF. We define the FFOM approximation by $x_k^{FF} = x_0 + Z_k y_k^{FF}$ and $V_k = [v_1, ..., v_k]$. The vector $y_k^{FF} \in \mathbb{R}^k$ is chosen such that $r_k^{FF} = r_0 - AZ_k y_k^{FF}$ is perpendicular to *span* $\{v_1, ..., v_k\}$. Using the relation $r_0 = \beta v_1$ it follows from

$$V_k^T(r_0 - AZ_k y_k^{FF}) = 0,$$

that $V_k^T AZ_k y_k^{FF} = \beta e_1$, where $e_1 \in \mathbb{R}^k$. Define $H_k \in \mathbb{R}^{k\times k}$ by $H_k = \{h_{i,j}\}_{\substack{1 \le i \le j \\ 1 \le j \le k}}$.

The relation

$$AZ_k = V_{k+1}\bar{H}_k, \tag{3.2}$$

given in ([5]; equation1.1) implies that:

$$V_k^T AZ_k = V_k^T V_{k+1}\bar{H}_k = \begin{bmatrix} & 0 \\ I_k & \vdots \\ & 0 \end{bmatrix} \bar{H}_k = H_k.$$

So the vectors y_k^{FF} satisfy $H_k y_k^{FF} = \beta e_1$. If H_k is non singular then x_k^{FF} exists and is given by $x_k^{FF} = x_0 + Z_k H_k^{-1}\beta e_1$.

In order to prove that $v_{k+1} \in$ *span* $\{r_k^{FF}\}$ we give some definitions. The matrix $\bar{H}_k$ can be factorized by Givens rotations into $\bar{H}_k = Q_k^T R_k$ where $Q_k \in \mathbb{R}^{k+1\times k+1}$, $Q_k^T Q_k = I_{k+1}$ and $R_k \in \mathbb{R}^{k+1\times k}$ is an upper triangular matrix. The matrix Q_k is formed by the product $F_k ... F_1$

here the matrix $F_j \in I\!R^{k+1 \times k+1}$ is the following Givens rotation

$$F_j = \begin{bmatrix} 1 & & & & & & & \\ & \ddots & & & & \emptyset & & \\ & & 1 & & & & & \\ & & & c_j & -s_j & & & \\ & & & s_j & c_j & & & \\ & & & & & 1 & & \\ & \emptyset & & & & & \ddots & \\ & & & & & & & 1 \end{bmatrix} .$$

The product $F_{k-1}...F_1\bar{H}_k = \begin{bmatrix} * & \dots & * & * \\ & \ddots & \vdots & \vdots \\ \emptyset & & * & * \\ & & 0 & \rho_k \\ & & 0 & h_{k+1,k} \end{bmatrix}$, where an asterisk stands for a non zero element, implies that c_k and s_k should be chosen as follows:

$$c_k = \rho_k / \sqrt{\rho_k^2 + h_{k+1,k}^2} \text{ and } s_k = -h_{k+1,k} / \sqrt{\rho_k^2 + h_{k+1,k}^2} .$$

<u>Lemma 1</u>
If $c_k \neq 0$ then the FFOM residual satisfies the relation:

$$r_k^{FF} = (s_1...s_k \|r_0\|_2 / c_k) v_{k+1} .$$

<u>Proof</u>
The relation $r_k^{FF} = r_0 - AZ_k y_k^{FF}$ combined with (3.2) gives

$$r_k^{FF} = r_0 - V_{k+1}\bar{H}_k y_k^{FF} = r_0 - V_{k+1} \begin{pmatrix} H_k y_k^{FF} \\ h_{k+1,k}\, e_k^T y_k^{FF} \end{pmatrix} .$$

Since $H_k y_k^{FF} = \|r_0\|_2 e_1$ we obtain

$$r_k^{FF} = -h_{k+1,k}\, e_k^T y_k^{FF} v_{k+1} .$$

Multiplication of $H_k y_k^{FF} = \|r_0\|_2\, e_1$ with Q_{k-1} gives

$$\begin{bmatrix} & * \\ R_{k-1} & \vdots \\ & * \\ & \rho_k \end{bmatrix} y_k^{FF} = Q_{k-1} \|r_0\|_2 e_1 .$$

Since R_{k-1} is upper triangular the last equation is equivalent to:

$$\rho_k e_k^T y_k^{FF} = s_1...s_{k-1} \|r_0\|_2 .$$

The assumption $c_k \neq 0$ implies that $\rho_k \neq 0$ so

$$r_k^{FF} = -s_1 ... s_{k-1} \frac{h_{k+1,k}}{\rho_k} \|r_0\|_2 v_{k+1} = (s_1 ... s_k \|r_0\|_2 / c_k) v_{k+1} \ ,$$

which proves the lemma. □

Remarks

- An overview of related results for Krylov subspace methods, with a constant preconditioner, is given in [3].

- From this relation, it appears that if the operators M_k are scaling invariant ($M_k(\alpha v) = \alpha M_k(v)$) then the search directions $z_k = M_k(v_k)$ are elements of *span* $\{M_k(r_{k-1}^{FF})\}$. Now the difference between FGMRES and GMRESR is clear: in FGMRES one calculates in the outer loop the minimal residual using search directions constructed from the FFOM residuals, whereas in GMRESR one calculates in the outer loop the minimal residual using search directions constructed from the GMRESR residuals. Note that the FGMRES and GMRESR residuals are the same if one uses the same search directions.

- Combination of Lemma 1 and the relation $\|r_k^{FG}\|_2 = |s_1|...|s_k| \ \|r_0\|_2$ leads to the relation (compare [1]):

$$\|r_k^{FG}\|_2 = |c_k| \ \|r_k^{FF}\|_2 \ .$$

This suggests that if there is a fast convergence ($c_k \simeq 1$) then $r_k^{FG} \simeq r_k^{FF}$, so we expect that the convergence behaviour of FGMRES and GMRESR are approximately the same. This is studied in more detail in Section 4.

The relation given in Lemma 1 can be used to specify a termination criterion for the innerloop, such that the outer loop residual has a prescribed accuracy.

Lemma 2

Suppose that H_k is non singular and $\|r_k^{FG}\|_2 > 0$. If the search direction $z_{k+1} = M_{k+1}(v_{k+1})$ satisfies the inequality

$$\|Az_{k+1} - v_{k+1}\|_2 < eps \ |c_k| / \|r_k^{FG}\|_2,$$

and x_{k+1}^{FG} exists then $\|r_{k+1}^{FG}\|_2 < eps$.

Proof

Since H_k is non singular x_k^{FF} exists. Using the auxiliary vector $\tilde{x}_{k+1} = x_k^{FF} + s_1 ... s_k \|r_0\|_2 / c_k z_{k+1}$ it follows from Lemma 1 that:

$$\begin{aligned} \|\tilde{r}_{k+1}\|_2 &= \|b - A\tilde{x}_{k+1}\|_2 = \|r_k^{FF} - (s_1 ... s_k \|r_0\|_2 / c_k) A z_{k+1}\|_2 \\ &= \|r_k^{FF}\|_2 \|v_{k+1} - Az_{k+1}\|_2 < \frac{eps \ |c_k| \ \|r_k^{FF}\|_2}{\|r_k^{FG}\|_2} = eps. \end{aligned}$$

Using the optimality property for the outer loop residual of FGMRES the result $\|r_{k+1}^{FG}\|_2 \leq \|\tilde{r}_{k+1}\|_2 < eps$ is proved. □

Note that $|c_k|$ and $\|r_k^{FG}\|_2$ are available, so this leads to a cheap termination criterion for the innerloop iteration. This termination criterion prevents too much iterations in the final innerloop.

We know that FGMRES only breaks down if $h_{k+1,k} = 0$. In the case that $h_{k+1,k} = 0$ and H_k is non singular we have a lucky breakdown: $x_k^{FF} = x_k^{FG} = x$ [5] Proposition 2.2 (compare [6], p. 864). So serious breakdown is only possible if H_k is singular and $h_{k+1,k} = 0$. This is illustrated by Example 1, where H_2 is singular, $h_{3,2} = 0$ and serious breakdown occurs. In GMRESR breakdown is avoided by choosing one LSQR step. If the current choice of z_k in FGMRES leads to breakdown a first idea could be to choose $z_k = A^T v_k$. However this is not a good idea. A counterexample is again Example 1 where $M_2 = A^2 = A^T$ and breakdown occurs.

In the following lemma we shall give a sufficient condition such that FGMRES has no breakdown. Before stating the lemma we note that the equation:

$$H_k = Q_{k-1}^T \begin{bmatrix} & * \\ R_{k-1} & \vdots \\ & * \\ & \rho_k \end{bmatrix}$$

implies that if $x_{k-1}^{FG} \neq x$ and $c_k \neq 0$ then H_k is non singular ([6], p. 864).

<u>Lemma 3</u>
Suppose that $c_1 \neq 0, ..., c_k \neq 0$ and $x_k^{FG} \neq x$. If the search direction z_{k+1} is such that

$$\|Az_{k+1} - v_{k+1}\|_2 < |c_k|$$

then H_{k+1} is non singular.

<u>Proof</u>
For $\tilde{x}_{k+1} = x_k^{FF} + (s_1...s_k\|r_0\|_2/c_k)z_{k+1}$ we obtain $\|\tilde{r}_{k+1}\| < \|r_k^{FG}\|_2$ (compare the proof of Lemma 2). This together with the optimality property of FGMRES gives

$$Az_{k+1} \notin span \; \{Az_1, ..., Az_k\}.$$

We shall now prove that the assumption "H_{k+1} is singular" leads to a contradiction. If H_{k+1} is singular, there is a vector $u \in I\!R^{k+1}$ such that $u \neq 0$ and $H_{k+1}u = 0$. From the definition of H_{k+1} it follows that

$$V_{k+1}^T A Z_{k+1} u = 0. \tag{3.3}$$

Since H_k is non singular and $Az_{k+1} \notin span \; \{Az_1, ..., Az_k\}$ the vector $\tilde{u} := AZ_{k+1}u \in span \; \{v_1, ..., v_{k+1}, Az_{k+1}\}$ is not equal to zero. Equation (3.3) implies $v_i^T \tilde{u} = 0$ for $i = 1, ..., k+1$, so there is a nonzero vector $\tilde{u} \in span \; \{v_1, ..., v_{k+1}, Az_{k+1}\}$ perpendicular to $span \; \{v_1, ..., v_{k+1}\}$, and thus $h_{k+2,k+1} \neq 0$. This implies that x_{k+1}^{FG} exists and $\|r_{k+1}^{FG}\|_2 \leq \|\tilde{r}_{k+1}\|_2 < \|r_k^{FG}\|_2$. This leads to $s_{k+1} < 1$, and thus $c_{k+1} \neq 0$. So H_{k+1} is non-singular, which is a contradiction. □

This inequality implies that the norm of the final residual of the inner loop is $|c_k|$ times the norm of the initial residual. Choosing GMRES in the inner loop, this inequality is easily satisfied for a large class of problems.

4 FGMRES with the search directions of GMRESR

In this section we show that it is possible to compute the GMRESR search directions in a cheap way during the FGMRES process. A consequence of this is that we can use a combination of FGMRES and GMRESR search directions in the FGMRES method.

Definition 1
The vectors w_k are defined by the following recurrence

$$w_1 = v_1$$

and

$$w_{k+1} = s_k w_k + c_k v_{k+1}, \quad k \geq 1,$$

where $\{v_k\}$ is given in the FGMRES algorithm.

It follows from Definition 1 that $w_k \in span\ \{v_1, ..., v_k\}$. Since $v_{k+1} \perp span\{v_1, ..., v_k\}$ the norm of w_{k+1} is given by

$$\|w_{k+1}\|_2 = (s_k^2 \|w_k\|_2^2 + c_k^2 \|v_{k+1}\|_2^2)^{1/2} = 1.$$

Note that the vectors w_k can be calculated in the FGMRES algorithm by one extra vectorupdate.

In the following lemma we give a relation between the vector w_{k+1} and the FGMRES residual r_k^{FG} (compare [3]).

Lemma 4
If the FGMRES approximation x_k^{FG} exists then the equation

$$r_k^{FG} = s_1 ... s_k \|r_0\|_2 w_{k+1} \text{ holds .}$$

Proof
From [5] it follows that

$$r_k^{FG} = r_0 - AZ_k y_k^{FG} = r_0 - V_{k+1} \bar{H}_k y_k^{FG}.$$

This can also be written as:

$$r_k^{FG} = r_0 - V_{k+1} Q_k^T Q_k \bar{H}_k y_k^{FG}.$$

The vector y_k^{FG} is computed such that (compare [6], p. 862):

$$Q_k \bar{H}_k y_k^{FG} = Q_k \|r_0\|_2\ e_1 - e_{k+1}^T Q_k \|r_0\|_2\ e_1\ e_{k+1}.$$

Combination of these expressions gives

$$r_k^{FG} = e_{k+1}^T Q_k \|r_0\|_2 e_1 V_{k+1} Q_k^T e_{k+1}.$$

It is easy to see that $V_{k+1}Q_k^T e_{k+1} = w_{k+1}$ and $e_{k+1}^T Q_k \|r_0\|_2 \; e_1 = s_1 ... s_k \|r_0\|_2$. This proves the lemma. □

In the original FGMRES scheme $\|r_k^{FG}\|_2$ is known but r_k^{FG} is not available. Using Lemma 4 (which is also valid for GMRES) r_k^{FG} can be calculated, so it is possible to inspect the residual during the computation, or to use other norms in the termination criterion.

Note that as a consequence of Lemma 4 we can use the GMRESR search directions in FGMRES by choosing $z_k = M_k(w_k)$. This again follows from the fact that the FGMRES and GMRESR residuals are the same if one uses the same search directions. So FGMRES can use a combination of FGMRES and GMRESR search directions. Using $z_k = M_k(s_1 ... s_k \|r_0\|_2 w_k)$ we can use the same termination criterion in the inner loop as GMRESR ([7]; Lemma 3).

In the following lemma we show that breakdown of FGMRES can be avoided by using an LSQR switch (for LSQR see [4]).

Definition 2
The LSQR switch is defined as follows: if the FGMRES search direction z_{k+1} leads to a singular matrix H_{k+1}, then use the following search direction $z_{k+1} = A^T w_{k+1}$.

Lemma 5
FGMRES with LSQR switch does not breakdown.

Proof
Suppose the current choice z_{k+1} leads to a singular matrix H_{k+1}. Then the search direction is replaced by $z_{k+1} = A^T w_{k+1}$. Since r_k^{FG} has the minimal residual property r_k^{FG} is perpendicular to *span* $\{Az_1, ..., Az_k\}$. From $(r_k^{FG})^T A z_{k+1} = s_1 ... s_k \|r_0\|_2 w_{k+1}^T A A^T w_{k+1} \neq 0$ it follows that $Az_{k+1} \notin$ *span* $\{Az_1, ..., Az_k\}$. This combined with

$$\|\tilde{r}_{k+1}\|_2 := \|r_k^{FG} - (r_k^{FG})^T A z_{k+1} / \|Az_{k+1}\|_2^2 A z_{k+1}\|_2 < \|r_k^{FG}\|_2$$

implies that H_{k+1} is non singular (compare the proof of Lemma 3) and so no serious breakdown occurs. □

The relation $r_k^{FG} = s_1 ... s_k \|r_0\| w_{k+1}$ can be used to give a further explanation of the differences between FGMRES and GMRESR. In the second iteration we have:

$$x_2^{FG} \in span \; \{M_1(r_0), M_2(v_2)\}$$

and

$$x_2^{GR} \in span \; \{M_1(r_0), M_2(s_1 r_0 / \|r_0\|_2 + c_2 v_2)\},$$

where we use $r_1^{GR} = r_1^{FG} = s_1 \|r_0\|_2 w_2 = s_1 \|r_0\|_2 (s_1 v_1 + c_1 v_2)$. Now it is clear that only if the operators M_k are all equal to the same linear operator M then the FGMRES and GMRESR results may be the same.

Conclusions

If $c_k \neq 0$ the relations

$$r_k^{FF} = s_1 ... s_k / c_k \|r_0\|_2 v_{k+1} \quad \text{(Lemma 1)},$$

$$r_k^{FG} = s_1 ... s_k \|r_0\|_2 w_{k+1} \quad \text{(Lemma 4)},$$

and

$$w_{k+1} = s_k w_k + c_k v_{k+1} \quad \text{(Definition 1)},$$

can be combined to

$$r_k^{FG} = s_k^2 r_{k-1}^{FG} + c_k^2 r_k^{FF} \quad .$$

So if $|s_m| \ll 1$ then the FFOM and FGMRES residual are close together, independent of the values of $s_1, ..., s_{k-1}$. Furthermore if $|s_k| \ll 1$ for all $k \geq 1$, then $M_k(v_k)$ and $M_k(r_{k-1}^{FG}) = M_k(w_k)$ are close together. So if the convergence of FGMRES is fast we expect that the convergence behaviours of both methods (without restarting) are comparable. However, in the case of slow convergence there may be a large difference between FGMRES and GMRESR (this difference depends on $s_1, ..., s_k$).

5 New results for GMRESR

In Subsection 5.1 we consider a variant of GMRESR, where the search directions can be chosen equal to the FGMRES search directions. Thereafter we specify in Subsection 5.2 a slightly cheaper implementation of the GMRESR method.

5.1 GMRESR with the search direction of FGMRES

In this subsection the expression x "is equal to" y means $x \neq 0$ and $x \in span\ \{y\}$. Furthermore we assume that no breakdown occurs. Considering the FGMRES algorithm we note that v_{k+1} "is equal to" the component of Az_k perpendicular to $span\ \{r_0, Az_1, ..., Az_{k-1}\}$. If we choose $u_k^{(i)}$ "equal to" z_k it follows that c_k "is equal to" the component of $Au_k^{(1)} = Az_k$ perpendicular to $span\ \{Az_1, ..., Az_{k-1}\}$. Since r_{k-1}^{GR} "is equal to" the component of r_0 perpendicular to $span\ \{Az_1, ..., Az_{k-1}\}$ it is easily seen that the vector p_k defined by

$$p_k := c_k - \frac{c_k^T r_{k-1}^{GR}}{\|r_{k-1}^{GR}\|_2^2} r_{k-1}^{GR}, \tag{5.1}$$

"is equal to" v_k. So if we choose $p_1 = r_1^{GR}$ and

$$u_k^{(1)} = M_k(p_k) \tag{5.2}$$

the algorithms FGMRES and GMRESR lead to the same results in exact arithmetic. Note that the calculation of p_k costs only one extra vector update per outer iteration.

Using the relations (5.1) and (5.2) it appears that GMRESR can also use a combination of FGMRES and GMRESR search directions. GMRESR combined with (5.1), (5.2) and truncation is a new method because there is no truncated FGMRES variant.

5.2 A faster implementation of GMRESR

Comparing FGMRES and GMRESR it appears that the number of vector updates in the outer loop of GMRESR is two times as large as for FGMRES. In this subsection we give a GMRESR version, where the number of vector updates in the outer loop is halved, and thus comparable with FGMRES.

We give an implementation of GMRESR, such that in the outer loop only the vectors $u_k^{(1)}$ and c_k are calculated. In the final iteration the approximate solution is calculated using the vectors $u_k^{(1)}$. This implementation can be used in combination with restarting and the truncfirst truncation strategy (see [9]; Section 3). The number of vectors used in the truncation is denoted by: $ntrunc$.

<u>Definition 3</u>
The following quantities are defined for the GMRESR algorithm:

$$\alpha_{k,i} = c_i^T c_k^{(i)},\ \gamma_k = 1/\|c_k\|_2 \text{ and } \delta_k = c_k^T r_{k-1}.$$

We define $\beta_{k,j}$ such that

$$u_k^{(k)} = \sum_{j=1}^{k} \beta_{k,j} u_j^{(1)} \text{ for } k < ntrunc$$

and

$$u_k^{(k)} = \tilde{u}_k + \sum_{j=1}^{ntrunc-1} \beta_{k,j} u_j^{(1)} \text{ for } k \geq ntrunc,$$

where $\tilde{u}_{ntrunc-1} = 0$ and

$$\tilde{u}_k = u_k^{(0)} - \alpha_{k,k-1}\, \gamma_{k-1}\, \tilde{u}_{k-1} \text{ for } k \geq ntrunc.$$

Combination of the relations given in Definition 3 leads to the following expressions for $\beta_{k,j}$:

$$\begin{aligned} \beta_{k,k} &= 1 \\ \beta_{k,j} &= -\sum_{i=j}^{k-1} \alpha_{k,i}\, \gamma_i\, \beta_{i,j} \text{ for } j = 1, ..., k-1,\ k < ntrunc, \end{aligned}$$

whereas

$$\beta_{k,j} = -\sum_{i=j}^{ntrunc-1} \alpha_{k,j}\, \gamma_i\, \beta_{i,j} - \alpha_{k,k-1}\, \gamma_{k-1}\, \beta_{k-1,j} \text{ for } j = 1, ..., ntrunc-1,\ k \geq ntrunc.$$

This enables us to calculate $\beta_{k,j}$. Finally we give a relation to calculate the approximation x_l from the vectors $u_k^{(1)}$. From the GMRESR algorithm it appears that

$$x_l = x_0 + \sum_{k=1}^{l} \delta_k\, \gamma_k\, u_k^{(k)}.$$

Substituting the relation given in Definition 3 into this equation leads to:

$$\begin{aligned} x_l &= x_0 + \sum_{k=1}^{ntrunc-1} \delta_k \, \gamma_k \sum_{i=1}^{k} \beta_{k,i} \; u_i^{(1)} \\ &+ \sum_{k=ntrunc}^{l} \delta_k \, \gamma_k (\tilde{u} + \sum_{i=1}^{ntrunc-1} \beta_{k,i} \; u_i^{(1)}). \end{aligned}$$

This can be implemented using the following extra memory: one n-vector to store $\tilde{u}_k$, three vectors with length $ntrunc$ for $\alpha_{k,i}$, γ_k and δ_k and a 2-dimensional array with dimensions $ntrunc$ to store $\beta_{k,j}$. Besides the work to calculate c_k in the outer loop we use for $l \geq ntrunc$ two vectorupdates to calculate $\tilde{u}_k$ and update x_0 per outer iteration. Finally the approximation is formed by $ntrunc$ vectorupdates. Note that the amount of memory and work of this GMRESR variant is comparable with FGMRES.

This approach seems not feasible for other truncation strategies. To illustrate this we look at the trunclast strategy ([9], Section 3). In this strategy u_1 and c_1 are discarded after $ntrunc$ iterations. However, since u_1 is used in the construction of $u_2, .., u_{ntrunc}$ these vectors should be adapted. This costs $ntrunc$ extra vectorupdates, which is as expensive as the original GMRESR algorithm.

In the order to compare the original FGMRES method and both GMRESR variants (where the original GMRESR search directions are used) we apply the methods to the test problem given in Section 3.1. In the following experiments we take $h = 1/50$ and $\beta = 1$. The new version, without calculation of the vectors u_k, is denoted by GMRESR_new. We always apply one GMRES(m) step as inner iteration process. The results are given in Table 5.1.

m	FGMRES	GMRESR	GMRESR_new
2	1.5	2.1	1.5
3	0.89	1.13	0.85
4	0.69	0.84	0.64
5	0.59	0.69	0.57
6	0.53	0.63	0.53
7	0.49	0.59	0.52
8	0.53	0.56	0.52
9	0.56	0.59	0.54
10	0.54	0.60	0.57

Table 5.1: CPU times for different methods and different values of m.

The CPU time is measured in seconds using 1 processor of a Convex C3820. It appears that the CPU time of GMRESR_new is comparable with FGMRES. For small m the CPU time is much less than for GMRESR. For m in the vicinity of the optimal value ($m = 8$) the difference in CPU time is small.

Finally we compare the truncfirst version of GMRESR and GMRESR_new, and the restarted version of FGMRES. The results for different values of *ntrunc* or *nstart* are given in Figure 5.1. Note that GMRESR_new is again faster than GMRESR, and restarted FGMRES.

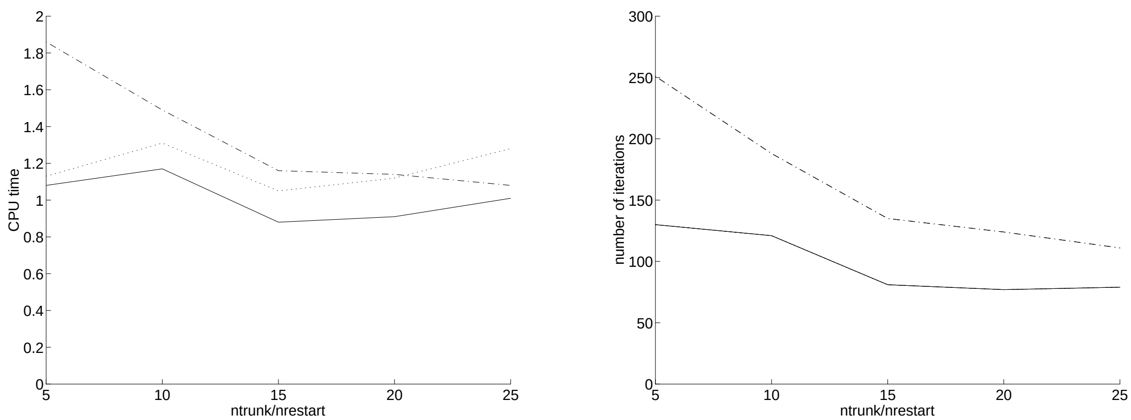

Figure 5.1: The CPU time and number of equations for GMRESR (..), GMRESR_new (–), and FGMRES (-.).

In both experiments the number of iterations and the approximations of GMRESR and GMRESR_new are the same.

6 Conclusions

We describe and compare the FGMRES and GMRESR methods. To facilitate the comparison we describe a new method, FFOM, related to FGMRES. This method is used to show that the FGMRES search directions are constructed from the FFOM residuals. This insight can be used to avoid breakdown and to give a termination criterion for the inner loop. Furthermore it enables us to give a detailed comparison of FGMRES and GMRESR. It appears that if the convergence of FGMRES is fast then the convergence behaviour of both methods is comparable.

A variant of FGMRES is given which uses the search directions of GMRESR and vice versa. Both methods can also use a combination of search directions, for instance the first iterations GMRESR search directions and then FGMRES search directions. Furthermore, if one method is implemented then a small change is sufficient to obtain results for the other method.

In the original GMRESR method one uses two times as much vector updates in the outer loop as FGMRES. We give a new implementation of GMRESR, which uses the same amount of work in the outer loop as FGMRES. This implementation can be combined with restarting and the truncfirst truncation strategy.

References

[1] P.N. Brown,
A theoretical comparison of the Arnoldi and GMRES algorithms.
SIAM J. Sci. Statist. Comput., **12** (1991), pp.58-78.

[2] S.C. Eisenstat, H.C. Elman and M.H. Schultz,
Variational iterative methods for non symmetric systems of linear equations,
SIAM J. Numer. Anal., **20** (1983), pp.345-357.

[3] M.H. Gutknecht,
Changing the norm in conjugate gradient type algorithms,
SIAM J. Num. Anal., **30** (1993), pp.40-56.

[4] C.C. Paige and M.A. Saunders,
LSQR: An algorithm for sparse linear equations and sparse least squares,
ACM Trans. Math. Soft., **8** (1982),pp.43-71.

[5] Y. Saad,
A flexible Inner-Outer preconditioned GMRES algorithm,
SIAM J. Sci. Stat. Comput. **14** (1993),pp.461-469.

[6] Y. Saad and M.H. Schultz,
GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems,
SIAM J. Sci. Statist. Comput., **7** (1986), pp.856-869.

[7] H.A. van der Vorst and C. Vuik,
GMRESR: a family of nested GMRES methods,
Report 91-80, Faculty of Technical Mathematics and Informatics,
Delft University of Technology,
J. Num. Lin. Alg. Appl., to appear, (1993).

[8] H.A. van der Vorst and C. Vuik,
The superlinear convergence behaviour of GMRES,
Report 90-77, Faculty of Technical Mathematics and Informatics,
Delft University of Technology,
J. Comp. Appl. Math., to appear, (1993).

[9] C. Vuik,
Further experiences with GMRESR,
Supercomputer, to appear, (1993).