Variants of the Residual Minimizing Krylov Space Method

Rozložník, Miroslav
1994

VARIANTS OF THE RESIDUAL MINIMIZING KRYLOV SPACE METHODS

M. ROZLOŽNÍK, Z. STRAKOŠ [*]

June 28, 1994

Institute of Computer Science, Academy of Sciences of the Czech Republic
Pod vodárenskou věží 2, 182 07 Praha 8, Czech Republic
e-mail: miro@uivt.cas.cz, strakos@uivt.cas.cz

Abstract.

Several variants of the GMRES method for solving linear nonsingular systems of algebraic equations are described. These variants differ in building up different sets of orthonormalized vectors used for the construction of the approximate solution. A new $A^T A$ variant of GMRES is proposed and optimal implementations of the algorithms are thoroughly discussed. It is shown that the described implementations are superior to widely used schemes ORTHODIR, ORTHOMIN and their relatives.

1 Introduction

Let $Ax = b$ be a system of linear algebraic equations, where A is a real nonsingular N by N matrix and b N-dimensional real vector. Many iterative methods for solving this system start with an initial guess x^0 for the solution and seek the n-th approximate solution x^n in the linear variety

$$x^n \in x^0 + K_n(A, r^0) \tag{1.1}$$

where $r^0 = b - Ax^0$ is the initial residual and $K_n(A, r^0)$ is the $n-th$ Krylov subspace generated by A, r^0,

$$K_n(A, r^0) = span\{r^0, Ar^0, \ldots, A^{n-1}r^0\}. \tag{1.2}$$

Then, the $n-th$ error and the $n-th$ residual are written in the form

$$x - x^n \; = \; p_n(A)(x - x^0) \tag{1.3}$$

[*]Part of this work was performed while visiting Department of Mathematics, Linköping University, Sweden

$$r^n \;=\; p_n(A)r^0, \tag{1.4}$$

where $p_n \in \mathcal{P}_n$, $\mathcal{P}_n$ denotes the set of polynomials of the degree at most n with the constant term equal to one. (1.1) defines Krylov space (or subspace) methods; based on (1.3) or (1.4), methods of this class are also referred to as polynomial methods.

Among the broad variety of the Krylov space methods (surveys can be found, e.g., in [18], [51], [50], [45], [44], [14], [30], [28], [11], [22]) we will concentrate especially on those which minimize the Euclidean norm of the residual

$$\|b - Ax^n\| = \min_{u \in x^0 + K_n(A,r^0)} \|b - Au\|, \tag{1.5}$$

or, equivalently, for which the n-th residual r^n is orthogonal to the shifted n-th Krylov subspace $K_n(A, Ar^0) \equiv AK_n(A, r^0)$

$$r^n \perp AK_n(A, r^0). \tag{1.6}$$

For A symmetric, the area is well mapped by the several classical papers [27], [37], [16], [15]. In [50] and [51], various methods are examined in a very clear exposition, and the most efficient and stable versions are described. In the nonsymmetric case the state of the art is not so advanced. Numerous algorithms generating the optimal approximation x^n determined by (1.5) were proposed and named ORTHODIR, ORTHOMIN, GMRES etc. [46], [65], [54], [2], [14]. The approximation defined by (1.5) always exist and is unique, therefore all the methods designed for computing this approximation at any step n are mathematically, i.e., in the exact arithmetic, equivalent. In the finite precision arithmetic, hovewer, they may have quite different convergence behavior. Surprisingly many variants are used in practical computations, including the numerically unstable or inefficient ones. Recently, Walker and his collaborators presented several papers studying the efficiency and numerical stability of the GMRES implementations [60], [61], [53], and proposed new, simpler implementation of the GMRES method [62]. GMRES and related methods have also been thoroughly studied and interesting generalizations were proposed by van der Vorst and others [56], [57], [58], [29].

In this paper, we describe several variants of the GMRES method. Our exposition is based on the following observation: For any minimal residual method, the approximate solution is constructed via some set of orthogonal vectors (note that these vectors may not span the subspaces $K_n(A, r^0)$ from (1.1) and may be orthogonal with respect to some other than the Euclidean innerproduct); *the choice of this set of orthogonal vectors together with the way in which the orthogonality is enforced determine the numerical stability of the algorithm.* We demonstrate it also on the relations of the described algorithms to the other mathematically equivalent or related methods. We point out the effective and stable versions and indicate the weak points of the unstable ones. The paper is organized as follows. Section 2 is devoted to variants of GMRES, including a new variant based on the A^TA-orthonormal basis of the Krylov subspaces $K_n(A, r^0)$. Section 3 describes implementation details. Relation to other methods are specified in Section 4. Section 5 presents results of numerical experiments. Section 6 contains concluding remarks.

Sometimes it is useful to consider the following generalization of (1.1)

$$x^n \in x^0 + K_n(B, Cr^0), \tag{1.7}$$

where $K_n(B, Cr^0)$ is the n-th Krylov subspace generated by B, Cr^0; B and C are nonsingular N by N matrices. For an example we refer to the Generalized mimimal error method (GMERR) introduced by Weiss in [63], [64]. In the forthcoming paper [39], analogical description of the variants of GMERR is given and the relation to an other error minimization methods is discussed.

As it is well known [19], see also [20], [31], [55], methods based on (1.5) require, in general, full term recurrences. Recently, a lot of effort has been devoted to the methods for nonsymmetric (non-Hermitian) systems using short recurrences (see [18] and the papers referred to there) which generate in most cases good, but not optimal, approximate solution. These methods are not within the subject of this paper.

We describe variants of various methods and point out relations between them using the key words "minimization property, orthogonality, normalization, bases of the Krylov subspaces" etc., which implicitly assume, of course, exact arithmetic. Any switch to the discussion of the actual finite precision results is always explicitly mentioned in the text so that any confusion is avoided. It should also be mentioned that we consider, for simplicity, the basic unpreconditioned versions of the methods, the extension to the preconditioned ones is straight. Throughout the paper matrices and vectors are real. For the complex case the results can be formulated accordingly. A remark to the notation. $X = [x_1, \ldots, x_j]$ denotes the matrix X with columns $x_1, x_2, \ldots, x_j$. Similarly $[X_1, X_2, \ldots, X_l]$ denotes matrix with column blocks $X_1, X_2, \ldots, X_l$. (x, y) is the Euclidean innerproduct of vectors x and y, $(x, y)_Z$ denotes the innerproduct defined by the SPD matrix Z, $(x, y)_Z = x^TZy$.

2 Variants of the GMRES method

After a brief summarization of the classical approach due to Saad a Schultz, we recall the variant proposed by Walker and Lu Zhou and propose a new variant based on the A^TA-orthonormal basis of the Krylov space $K_n(A, r^0)$. Implementation details and optimal formulation of the algorithms suitable for the development of the computer code are discussed in Section 3.

The classical formulation of GMRES proposed by Saad and Schultz [46] is based on the orthonormal basis $V_{n+1} = [v_1, \ldots, v_{n+1}]$ of the Krylov space $K_{n+1}(A, r^0)$ which is computed via the Arnoldi recurrence

$$v_1 = r^0/\varrho, \;\; \varrho = \|r^0\|, \;\; AV_n = V_{n+1}H_{n+1,n}, \;\; V_{n+1}^T V_{n+1} = I_{n+1}. \tag{2.1}$$

It easy to see (cf. [23]) that the Arnoldi recurrence in steps 1 through n can be viewed as a recursive column-oriented QR decomposition of the matrix $[r^0, AV_n]$,

$$[r^0, AV_n] = V_{n+1}U_{n+1}, \tag{2.2}$$

where the upper triangular matrix U_{n+1} and the upper Hessenberg matrix $H_{n+1,n}$ are related by

$$U_{n+1} = [\,\varrho\, e_1, H_{n+1,n}]. \tag{2.3}$$

Using the substitution

$$x^n = x^0 + V_n y^n \tag{2.4}$$

the vector of coefficients y^n is found as the solution of the transformed least squares problem

$$\|\varrho e_1 - H_{n+1,n}y^n\| = \min_{y \in R^n} \|\varrho e_1 - H_{n+1,n}y\|. \tag{2.5}$$

The upper Hessenberg matrix $H_{n+1,n}$ is then reduced to the upper triangular matrix via the Givens transformations

$$J_n(J_{n-1}\ldots J_2 J_1)\; H_{n+1,n} = \begin{pmatrix} R_n \\ 0 \end{pmatrix}, \tag{2.6}$$

where the elementary Givens matrix J_n acts on the n-th and $(n+1)$-th row to eliminate the nonzero element in the $(n+1)$-th row of $H_{n+1,n}$. Subsequently, the norm of the residual can be easily updated step by step without computing the approximation x_n explicitly,

$$\|r^n\| \;=\; \varrho\, e_{n+1}^T\, J_n(J_{n-1}\ldots J_1)e_1. \tag{2.7}$$

For $\|r^n\|$ small enough, y^n is computed from

$$R_n y^n \;=\; \varrho\, [I_n,\, 0]\; J_n(J_{n-1}\ldots J_1)e_1. \tag{2.8}$$

We give a brief summary. The orthonormal basis of the Krylov subspace $K_n(A, r^0)$ is in the Saad and Schultz (SS) variant of GMRES used for reducing the problem (1.5) into the transformed (upper Hessenberg) least squares problem (2.5). The last one is solved by the QR factorization of the upper Hessenberg matrix $H_{n+1,n}$ using the Givens rotations.

A simpler implementation of GMRES not requiring the QR factorization of the upper Hessenberg matrix containing the Arnoldi coefficients was proposed by Walker and Lu Zhou [62]. This variant, denoted here as WLZ, computes orthonormal basis $Q_n = [q_1, \dots, q_n]$ of the shifted Krylov subspaces $AK_n(A, r^0) \equiv K_n(A, Ar^0)$. This can be done via the recursive column by column QR factorization of the matrix $[Ar^0, AQ_{n-1}]$

$$[Ar^0, AQ_{n-1}] = Q_n S_n, \quad Q_n^T Q_n = I_n, \tag{2.9}$$

where S_n is upper triangular. Then, the substitution (2.4) is replaced by

$$x^n = x^0 + [r^0, Q_{n-1}]\, t^n \tag{2.10}$$

and the optimality condition (1.6), rewritten as $Q_n^T r^n = 0$, gives

$$Q_n^T r^0 - Q_n^T A[r^0, Q_{n-1}]\, t^n = 0. \tag{2.11}$$

Combining (2.11) with (2.9), we receive the upper triangular linear system for determining the unknown vector t^n,

$$S_n t^n = Q_n^T r^0. \tag{2.12}$$

In this variant, both the residual vector and its norm can be easily updated step by step. Indeed, using (2.10), (2.9) and (2.12),

$$r^n = r^0 - Q_n Q_n^T r^0 = r^{n-1} - (q_n, r^0)\, q_n, \tag{2.13}$$

$$\|r^n\|^2 = \|r^{n-1}\|^2 - (q_n, r^0)^2. \tag{2.14}$$

From orthogonality, $Q_n^T r^0$ resp. (q_n, r^0) can be replaced in (2.12) - (2.14) by $[(q_1, r^0), (q_2, r^1), \dots, (q_n, r^{n-1})]^T$ resp. (q_n, r^{n-1}).

Summary: By generating the orthonormal basis of $AK_n(A, r^0)$, the WLZ variant provides the step by step update of the residual vector and simpler update of the norm of the residual, as well as simpler computation of the approximate solution than the SS variant. The basis used for determining the approximate solution is, however, not orthogonal, which may affect the numerical stability of this variant.

To determine the approximate solution, the WLZ variant still requires, however, solving an upper triangular system. One can look for a variant computing a simple step by step update of the approximate solution. This can be done by generating the $A^T A$-orthonormal basis $W_n = [w_1, \dots, w_n]$ of $K_n(A, r^0)$. Indeed, let W_n be a result of the recursive column by column QR factorization of the matrix $[r^0, AW_{n-1}]$, assuming the innerproduct $(g, h)_{A^T A} = g^T A^T A h$,

$$[r^0, AW_{n-1}] = W_n G_n, \quad W_n^T A^T A W_n = I_n \tag{2.15}$$

where G_n is upper triangular. Then, using

$$x^n = x^0 + W_n f^n, \tag{2.16}$$

the optimality condition (1.6) implies

$$f^n = (AW_n)^T r^0, \tag{2.17}$$

and the approximate solution, as well as the residual vector and the norm of the residual can be easily updated step by step

$$x^n = x^{n-1} + (Aw_n, r^0)\, w_n \tag{2.18}$$

$$r^n = r^{n-1} - (Aw_n, r^0)\, Aw_n \tag{2.19}$$

$$\|r^n\|^2 \;=\; \|r^{n-1}\|^2 - (Aw_n, r^0)^2 \tag{2.20}$$

Analogously to the WLZ variant, $(AW_n)^T r^0$ resp. (Aw_n, r^0) can be replaced in (2.17) - (2.20) by $[(Aw_1, r^0), (Aw_2, r^1), \ldots, (Aw_n, r^{n-1})]^T$ resp. (Aw_n, r^{n-1}). Throughout the paper, we will denote this formulation of GMRES as the $A^T A$ variant.

In the next section, the WLZ and $A^T A$ variants described above are further developed into detailed algorithms.

3 Implementation details

Detailed schemes for the SS variant of GMRES based on the modified Gram-Schmidt algorithm can be found e.g. in [46], [4] and [59]. The implementation of the SS variant using Householder transformations for computing the Arnoldi basis is described in [60], [61]. In [23], the numerical stability of GMRES is studied. It is shown, that even for the most stable implementations based on the Householder transformations or iterated Gram-Schmidt algorithm, GMRES may fail to produce for an ill-conditioned system results on the same level of accuracy as direct methods based on the QR decomposition. Though [23] is restricted to the SS variant, the development can be extended (with some modifications) to the WLZ and $A^T A$ variants as well. We are not going to present these extensions here (the forthcoming paper [40] is devoted to this subject). Instead, we will discuss the optimal implementations of the WLZ and $A^T A$ variants and describe detailed schemes suitable for the development of code. For simplicity we will consider the basic (i.e. unrestarted and unpreconditioned) forms of the algorithm. Restarting and preconditioning can be incorporated in the well known way, see e.g. [59], [4]. A modification to the truncated schemes (see Section 4.2) is also straight. We will also not present the details of the orthogonalization in computing the basis vectors; for those we refer in [23].

3.1 Walker and Lu Zhou variants

The simplest implementation of the WLZ variant, denoted as WLZ1, is given by the next scheme (following the argument from [62] we use the normalized initial residual $\hat{r}^0 = r^0/\varrho)$:

```
ALGORITHM WLZ1.
$x^0$ is an initial guess;
set $r^0 = b - Ax^0$ and $\varrho = \|r^0\|$; if small enough the accept $x^0$ and exit;
$\hat{r}^0 = r^0/\varrho$; $\varrho_0 = 1$;
$q_1 = A\hat{r}^0/\|A\hat{r}^0\|$; $S_1 = \varrho\,\|A\hat{r}^0\|$;
for $j = 1, 2, \ldots, n$
   update the relative norm of the residual $\varrho_j$:
   $\xi_j = (q_j, \hat{r}^0)$;
   $\varrho_j^2 = \varrho_{j-1}^2(1 - \xi_j^2/\varrho_{j-1}^2)$;
   if $\varrho_j$ small enough then SOLVE($x^j$); quit;
   extend the basis to $Q_{j+1}$ and the triangular matrix to $S_{j+1}$ :
   $w = Aq_j$;
   orthogonalize $w$ against $q_1, \ldots, q_j$ and normalize the result to obtain $q_{j+1}$;
   extend $S_j$ to $S_{j+1}$ by adding the column vector of orthogonalization and
   normalization coefficients;
end;
form the final update:
$\xi_{n+1} = (q_{n+1}, \hat{r}^0)$;
SOLVE($x^{n+1}$);
quit;
```

In this scheme SOLVE(x^j) stands for the following computations:

```
compute $t^j$ as the solution of the upper triangular system $S_j t^j = (\xi_1, \ldots, \xi_j)^T$;
form $x^j = x^0 + \varrho\,[r^0, q_1, \ldots, q_{j-1}]\,t^j$;
quit;
```

The initial residual is normalized for the reasons mentioned in [62]. Note that WLZ1 does not require updating the residual vector r^j which keeps the work and storage at minimum. Also the part SOLVE(x^j) is considerably simpler than that in the original scheme described in [62]. The simplification of updating of the solution in SOLVE(x^j) does not cause, to our opinion, any additional numerical troubles in comparison to original scheme by Walker and we recommend its use. Unfortunately, there is a numerical drawback in the main part of the scheme.

Iterative updating the norm of the residual by the formula

$$\varrho_j^2 = \varrho_{j-1}^2\{1 - (q_j, \hat{r}^0)^2/\varrho_{j-1}^2\} \tag{3.1}$$

is unreliable and may lead to an early termination of the computation caused by underestimating of the norm of the residual due to accumulation of round-off. Roughly speaking, the accumulated error in computing ϱ_j^2 by (3.1) can be bounded by $\gamma_1 jN\varepsilon$, where ε denotes the machine precision unit and γ_1 is constant independent on j, N and ε. Once ϱ_j^2 becomes of the order of $jN\varepsilon$, one cannot trust the result of (3.1) anymore. One may replace (3.1)

$$\varrho_j = \varrho_{j-1} sin(cos^{-1}(\xi_j/\varrho_{j-1})) \tag{3.2}$$

as proposed in [62], but to our opinion, and for the same reason, this does not behave any better. For $\varrho_j^2 \gg jN\varepsilon$, however, the results produced by WLZ1 are satisfactory accurate. Therefore WLZ1 can be considered as the most effective implementation of the WLZ variant if the method is restarted after just a few steps. These considerations are illustrated by our examples below.

If the square of the relative norm of the residual (within possible restarts) drops significantly close to the level $jN\varepsilon$, then the algorithm WLZ1 must be modified by adding the recursive computation of the residual vector at every step. This can be done by using (2.13). Having the normalized residual vector $\hat{r}^{j-1}$, one can further replace, with no additional costs, $(q_j, \hat{r}^0)$ by $(q_j, \hat{r}^{j-1})$, which improves numerical behavior of the algorithm. To see that, note that the actually computed basis vectors $q_1, q_2, \ldots, q_j$ are not orthogonal (for simplicity of exposition we assume that they are exactly normalized). From (2.13) it is clear that the approximate solution x^n computed from (2.12) and (2.10) corresponds to computing r^n by orthogonalization of r^0 against $q_1, q_2, \ldots, q_n$ via the classical Gram-Schmidt algorithm. We may therefore expect that the process is strongly affected by rounding errors and the optimality condition (1.6) is not satisfactorily enforced. One may therefore look for a formulation corresponding to computing r^n via the modified Gram-Schmidt, namely,

$$r^n = (I - q_n q_n^T)(I - q_{n-1}q_{n-1}^T)\ldots(I - q_1 q_1^T)r^0, \tag{3.3}$$

which would guarantee that the optimality condition (1.6) is satisfied to much higher accuracy. Following [7], [5] we denote

$$\tilde{q}_j = (I - q_1 q_1^T)\ldots(I - q_{j-1}q_{j-1}^T)q_j, \;\; j = 2, \ldots, n, \;\; \tilde{q}_1 = q_1 \tag{3.4}$$

and $\tilde{Q}_n = [\tilde{q}_1, \ldots, \tilde{q}_n]$. Then

$$(I - q_n q_n^T)(I - q_{n-1}q_{n-1}^T)\ldots(I - q_1 q_1^T) = I - Q_n\tilde{Q}_n^T. \tag{3.5}$$

Using (2.10),(2.9),

$$r^n = b - Ax^n = r^0 - A[r^0, Q_{n-1}]t^n = r^0 - Q_n S_n t^n. \tag{3.6}$$

Comparing (3.3) and (3.6),

$$(I - Q_n\tilde{Q}_n^T)r^0 = r^0 - Q_nS_nt^n,$$

from which we receive, after a simple manipulation,

$$Q_n(\tilde{Q}_n^T r^0 - S_nt^n) = 0. \tag{3.7}$$

Using the linear independence of the basis vectors $q_1, q_2, \ldots, q_n$, (3.7) implies

$$S_nt^n = \tilde{Q}_n^T r^0. \tag{3.8}$$

The i-th element of the vector $\tilde{Q}_n^T r^0$ is expressed as

$$(\tilde{q}_i, r^0) = ((I - q_1q_1^T)\ldots(I - q_{i-1}q_{i-1}^T)q_i, r^0) = (q_i, r^{i-1}),$$

which finally gives the equation for determining the vector of coordinates t^n

$$S_nt^n = [(q_1, r^0), (q_2, r^1), \ldots, (q_n, r^{n-1})]^T. \tag{3.9}$$

From (3.3),

$$r^n = (I - q_nq_n^T)r^{n-1} = r^{n-1} - (q_n, r^{n-1})q_n. \tag{3.10}$$

Consequently, we have shown that replacing $Q_n^T r^0$ resp. (q_n, r^0) in (2.12)-(2.14) by $[(q_1, r^0), (q_2, r^1), \ldots, (q_n, r^{n-1})]^T$ resp. (q_n, r^{n-1}), one may hope for much better numerical behavior of the algorithm. The iterative updating (3.1) of the residual norm ϱ_j will, however, face essentially the same numerical difficulty as described above, even if $(q_j, \hat{r}^0)$ is replaced by $(q_j, \hat{r}^{j-1})$. For a stable implementation we therefore suggest to base the stopping criterion on the norm of the residual computed by (3.10).

Finally, the implementation of the WLZ variant taking into consideration the above comments on the stability is given by the scheme:

Algorithm WLZ2.

```
$x^0$ is an initial guess;
set $r^0 = b - Ax^0$ and $\varrho = \|r^0\|$; if small enough the accept $x^0$ and quit;
$\hat{r}^0 = r^0/\varrho$;
$q_1 = A\hat{r}^0/\|A\hat{r}^0\|$; $S_1 = \varrho\,\|A\hat{r}^0\|$;
for $j = 1, 2, \ldots, n$
   $\xi_j = (q_j, \hat{r}^{j-1})$;
   $\hat{r}^j = \hat{r}^{j-1} - \xi_j q_j$;
   $\varrho_j = \|\hat{r}^j\|$, if $\varrho_j$ small enough then SOLVE($x^j$); quit;
   $w = Aq_j$;
   orthogonalize $w$ against $q_1, \ldots, q_j$ and normalize the result to obtain $q_{j+1}$;
   extend $S_j$ to $S_{j+1}$ by adding the column vector of orthogonalization and
```

```
    normalization coefficients;
end;
$\xi_{n+1} = (q_{n+1}, \hat{r}^n)$;
SOLVE($x^{n+1}$);
quit;
```

where SOLVE(x^j) has the same meaning as above. WLZ2 costs one more SAXPY and one more DOT product per iteration than WLZ1, but, as illustrated on our examples in Section 5, it behaves much better numerically. It should be therefore preferred over the other WLZ implementations. If the increase in the operation count per iteration is not negligible and is to be taken into account (e.g. in some restarted or truncated versions of the algorithm), we suggest to consider the combined implementation starting with WLZ1 and then switching to WLZ2 if ϱ_j^2 drops below $\alpha j N \varepsilon$, where α is a proper constant (e.g. $\alpha = 10$ or 10^2).

Both WLZ1 and WLZ2 implementations of the WLZ variant of GMRES are considerably simpler in comparison to the SS variant. On the other hand, as noticed in [62], the WLZ variant has certain numerical disadvantage in using nonorthogonal basis $\hat{r}^0, q_1, \ldots, q_{n-1}$ for computing the approximate solution x^n. Assuming exact arithmetic, Walker and Lu Zhou showed that the condition number of the matrix $[\hat{r}^0, Q_{n-1}]$ is large only if the relative residual is small. Consequently, this drawback seems not to be serious unless one is interested in very accurate approximations to the solution (more precisely, very small residuals). A finite precision analysis of this situation still needs further work. As mentioned in the Introduction, a detailed theoretical comparison of the numerical stability of the WLZ and SS variants is out of the range of this paper and will be published elsewhere.

3.2 A^TA variants

The A^TA variant requires computing the A^TA-orthonormal basis $w_1, w_2, \ldots, w_n$ of $K_n(A, r^0)$ which represents a serious complication. We will concentrate on the modified Gram-Schmidt orthogonalization here and show that this difficulty can be eliminated to some extent. The usual MGS implementation

```
(3.11)
$w_1 = r^0 / \|r^0\|_{A^TA}$;
for $j = 1, 2, \ldots, n$
    $t = Aw_j$;
    for $i = 1, 2, \ldots, j$
        $t = t - (At, Aw_i) w_i$;
    end;
    $w_{j+1} = t / \|t\|_{A^TA}$;
end;
```

is rather inefficient due to the kernel operation (At, Aw_i) which requires an extra matrix-vector multiplication in every substep of the MGS orthogonalization. If we consider an additional computing of the orthonormal vectors $u_1, u_2, \ldots, u_n$, where $u_j = Aw_j$, by the MGS process, then (3.11) can be replaced in the price of doubling the recurrences by

```
(3.12)
$w_1 = r^0/\|r^0\|_{A^TA}$; $u_1 = Aw_1$;
for $j = 1, 2, \ldots, n$
   $u = Au_j$; $t = u_j$;
   for $i = 1, 2, \ldots, j$
      $t = t - (u, u_i)w_i$;
      $u = u - (u, u_i)u_i$;
   end;
   $u_{j+1} = u/\|u\|$; $w_{j+1} = t/\|u\|$;
end;
```

It should be noted that in the finite precision arithmetic, the resulting basis vectors w_j may differ slightly from that computed by (3.11) because the recurrences for t from (3.11) and (3.12) are not equivalent in the presence of rounding errors.

For the discussion of other details we refer to Section 3.1. Using (3.12), a simple implementation of the A^TA variant is given analogously to Algorithm WLZ1 by the next scheme:

ALGORITHM A^TA-1.

```
$x^0$ is an initial guess;
set $r^0 = b - Ax^0$ and $\varrho_0 = \|r^0\|$; if $\varrho_0$ small enough then accept $x^0$ and quit;
$\alpha = \|Ar^0\|$; $w_1 = r^0/\alpha$; $u_1 = Aw_1$;
for $j = 1, 2, \ldots, n$
   $\xi_j = (u_j, r^0)$;
   $\varrho_j^2 = \varrho_{j-1}^2(1 - \xi_j^2/\varrho_{j-1}^2)$;
   if $\varrho_j$ small enough then FORM($x^j$); quit;
   $u = Au_j$; $t = u_j$;
   for $i = 1, 2, \ldots, j$
      $\eta_i = (u, u_i)$;
      $u = u - \eta_i u_i$;
      $t = t - \eta_i w_i$;
   end;
   $u_{j+1} = u/\|u\|$; $w_{j+1} = t/\|u\|$;
end;
$\xi_{n+1} = (u_{n+1}, r^0)$;
FORM($x^{n+1}$);
quit;
```

The procedure FORM(x^{m+1}) computes a simple updating of the solution:

```
$x^j = x^0 + \sum_{i=1}^{j} \xi_j w_j$;
quit;
```

It can be done step by step, but computing the sum $\sum_{i=1}^{j} \xi_j w_j$ and the adding to x^0 should be preferred for numerical reasons [32].

Analogously to Section 3.1, a more stable implementation of the $A^T A$ variant requires the recursive updating of the residuals, which results in the Algorithm $A^T A$-2. In practice, similarly to the WLZ variant, both implementations can be combined.

Algorithm $A^T A$-2.

```
$x^0$ is an initial guess;
set $r^0 = b - Ax^0$; if $\|r^0\|$ small enough then accept $x^0$ and quit;
$\alpha = \|Ar^0\|$; $w_1 = r^0/\alpha$; $u_1 = Aw_1$;
for $j = 1, 2, \ldots, n$
  $\xi_j = (u_j, r^{j-1})$;
  $r^j = r^{j-1} - \xi_j u_j$;
  if $\|r^j\|$ small enough then FORM($x^j$); quit;
  $u = Au_j$; $t = u_j$;
  for $i = 1, 2, \ldots, j$
    $\eta_i = (u, u_i)$;
    $u = u - \eta_i u_i$;
    $t = t - \eta_i w_i$;
  end;
  $u_{j+1} = u/\|u\|$; $w_{j+1} = t/\|u\|$;
end;
$\xi_{n+1} = (u_{n+1}, r^n)$;
FORM($x^{n+1}$);
quit;
```

Computing the $A^T A$-orthonormal basis of $K_n(A, r^0)$ via (3.12) requires one more SAXPY operation per every substep of the MGS orthogonalization process and about twice as storage as the MGS used in the SS or WLZ variants. On the other hand, the $A^T A$- implementations are considerably simpler than the implementations of the other variants. They offer a simple step by step updates of both the approximation to the solution and the residual vector. Moreover, the extra work can be easily done in parallel to the other arithmetic and algorithms can be performed in less sequential steps than the implementations of the SS and WLZ variants. Therefore, for parallel computer architectures, the code based on the $A^T A$ variant may become competitive to codes based on the SS or WLZ variants. The $A^T A$ variant of GMRES is closely related to several other methods used for years (e.g. the method ORTHODIR). As it will be explained in Section 4 and then numerically demonstrated in Section 5, the newly proposed $A^T A$ variant of GMRES is clearly superior to these well known relatives with respect to the numerical stability and the accuracy of the computed solution.

4 Relation to other methods

This section gives a very compact and short review of the related methods.

4.1 Symmetric Case

The conjugate gradient method (CG), an algorithm for solving systems of linear equations with symmetric positive definite (SPD) matrices, was introduced by Hestenes and Stiefel [27]. Its iterates minimize the energy norm of the error in the variety (1.1). The conjugate residual method (CR) due to Stiefel [49] is, in fact, the CG method applied to the original SPD system $Ax = b$ assuming the innerproduct $(u, v)_A = u^T A \; v$; its iterates satisfy the minimal residual property (1.5). The CG residual vectors form the orthogonal basis and the CG search vectors the A-orthogonal basis of (1.2). For CR, the bases are A-orthogonal and A^2-orthogonal, respectively. For any basis, the basis vectors are not normalized. The orthogonalization processes are equivalent to the unnormalized 3-term recurrences.

Fridman proposed in [16] the orthogonal direction (OD) method for solving symmetric indefinite systems which is characterized by the minimal error property

$$\|x - x^n\| = \min_{u \in x^0 + K_n(A, Ar^0)} \|x - u\|, \tag{4.1}$$

but his implementation was unstable. The first numerically stable algorithms for symmetric indefinite systems were proposed by Paige and Saunders in [37]. They considered the symmetric Lanczos method [34], [35] to generate an explicit *orthonormal basis* of the Krylov subspaces (1.2). It enables them to identify clearly the trouble with applying CG to indefinite problems. A CG iterate x^n_{CG} can be viewed as a solution of the orthogonal projection of the system $A(x - x^0) = r^0$ onto the subspace (1.2). For A indefinite, such solution simply may not exist due to possible singularity of the projected system matrix (which is the n by n tridiagonal matrix having the coefficients of the Lanczos recurrence as its columns). Even if it exists, its computing by the CG algorithm may lead to numerical instabilities. From the nonsingularity of the projected matrix at the final Lanczos step it follows that (at least) every second CG approximation must exist; Paige and Saunders have proposed a stable algorithm, named SYMMLQ, for computing those. SYMMLQ produces at every step also an approximation x^n_L satisfying (4.1). Using the same Lanczos basis, they derived the MINRES method satisfying the minimal residual property (1.5). This method can be viewed as a numerically stable generalization of the CR method to the symmetric indefinite case. The same Lanczos basis of (1.2) but different ways of computing the approximations to the solution were used by Simon and Chandra [47], [10]. Their methods are mathematically (i.e. in the exact arithmetic) equivalent to CG resp. SYMMLQ or MINRES.

Another approach was taken by Fletcher [15] who described the Bi-CG algorithm for nonsymmetric systems (which was essentially introduced by Lanczos [35]), and then modified its symmetric variant to obtain the OD method by Fridman. This algorithm computes the orthogonal, but not normalized, basis of $K_n(A, Ar^0)$; it faces some numerical difficulties. A stable modification, named STOD, was proposed by Freund and Stoer [50], [51]. In a way similar to OD, the A^2- orthogonal basis of the original Krylov subspace (1.2) can be computed and used to form a variant of the CR method, see [15], [51], [10]. While SYMMLQ and MINRES are based on the computation of the *orthonormal* basis of (1.2), methods mentioned in this paragraph compute orthogonal, but not normalized, vectors generating $K_n(A, Ar^0)$, resp. the A^2- orthogonal vectors generating (1.2). A variant of the residual minimizing method based on (1.5) which generates the A^2- *orthonormal* basis of (1.2) was described and used recently by Rusten [41]. Theoretical equivalence of the methods based on the Lanczos basis and those developed from the Bi-CG algorithm was already known by Fletcher, but there are some inaccuracies in proofs presented in [15].

It should be mentioned, that there are several variants - mathematically equivalent - of implementing the Lanczos recurrence for computing the orthonormal basis of (1.2). In the presence of rounding errors, however, their behavior may be different. The optimal variant, pointed out by Paige [36], can be considered as a symmetric analogy of the Modified Gram-Schmidt process. For A SPD, results of finite precision computation do not seem to depend on the variant of the algorithm used, unless the matrix A is very ill-conditioned. As an example may serve the CG approximation computed via the original Hestenes and Stiefel or SYMMLQ or the Simon's implementation - unless the condition number of the system matrix is extremely high (of order ε^{-1}, where ε is the machine precision), the obtained convergence behavior (measured, e.g., by the norm of the residual or the A-norm of the error) will be almost identical for all the three variants [52], [24]. On the other hand, the choice of the variant may be very important in the symmetric indefinite or nonsymmetric case. In general, finite precision behavior of the mathematically equivalent Krylov space methods may differ remarkably one from another.

4.2 Nonsymmetric case

A natural approach to the nonsymmetric case is to apply the CG method to the related systems with the symmetric positive definite matrix A^TA or AA^T. Solving the system $A^TAx = A^Tb$ by the CG method was suggested already by

Hestenes and Stiefel [27]; it is referred to as the CGNR method. Applying the CG method to the system $AA^Ty = b$, $x = A^Ty$, we get the CGNE method proposed by Craig [9]. From the minimization property of CG it follows that CGNR is characterized by the minimal residual property on $x^0 + K_n(A^TA, A^Tr^0)$ while CGNE by the minimal error property on $x^0 + K_n(A^TA, A^Tr^0)$. More stable implementations of these methods, based on the explicit computation of the orthonormal bases for $K_n(A^TA, r^0)$ and $K_n(A^TA, A^Tr^0)$ event. $K_n(A^TA, A^TAr^0)$ using the bidiagonalization algorithm of Golub and Kahan [21], was proposed by Paige and Saunders [36], [38]. The best known method from this family, referred to as LSQR, produces in the exact arithmetic the same approximation as CGNR. As it was pointed out by Björck in the unpublished work [6], finite precision results of the correct CGNR implementation differ remarkably from the LSQR results only for the very ill-conditioned systems, which opposes to the popular belief about the strong instability of CGNR. This observation is in a good agreement with our remark in paragraph 4.1 - both algorithms build up the basis for the subspaces determined by the SPD matrix A^TA (though this matrix is present in the algorithm only implicitly) and, unless A^TA is very ill-conditioned, we can expect a good agreement of their finite precision results. The convergence of all the methods mentioned in this paragraph is determined by the squares of the singular values of the matrix A, and may therefore be rather slow.

Although there are special situations where the approaches based on A^TA or AA^T are optimal (see, e.g., [18]), in most cases the methods applied directly to the original system $Ax = b$ are preferred. Unfortunately, for the iterates characterized by the minimal residual or the minimal error property on a Krylov subspace generated by a general nonsymmetric matrix, the work and storage per iteration increase linearly with the iteration step.

Numerous algorithms compute the iterates with the minimal residual property (1.5) in the general nonsymmetric case. As pointed out in [18], the idea of minimizing the residual norm has appeared already in [33]. The implementation proposed there is unstable and the discussion of convergence is not correct.

Young and Jea in 1980 [65] described two methods with iterates satisfying the condition (1.6): ORTHOMIN, proposed first by Vinsome in [54], and ORTHODIR. For a general survey we refer also to [28]. For the ORTHOMIN method, also called GCR (see e.g. [14], [13]), the convergence to the solution is assured for positive real systems (i.e. systems with the matrices having positive definite symmetric part $(A + A^T)/2$). ORTHODIR method converges for general nonsymmetric (nonsingular) matrix A. In both methods, the direction vectors build up implicitly the A^TA-orthogonal basis of the space $K_n(A, r^0)$. The basis vectors are unnormalized, and orthogonalized against each other in a way equivalent to the classical Gram-Schmidt orthogonalization. This can lead to the numerical instabilities of these methods. The algorithm of Axelsson [2], [3] is closely related to ORTHODIR and ORTHOMIN.

In all the variants described in the previous section, GMRES differs from these methods by computing an orthonormal basis, used for the construction of the solution, in a far more stable way. We emphasize that both the normalization, and using the modified Gram-Schmidt or even more stable orthogonalization scheme [23], instead of classical Gram-Schmidt, make the variants of GMRES from the previous section numerically superior to ORTHODIR, ORTHOMIN and their modifications.

Several other nonoptimal and related methods were proposed. The Arnoldi (FOM) method [43] is also based on the computation of an orthonormal Arnoldi basis [1]; for the connection to GMRES see [8]. A deep theoretical survey of the relations between various methods can be found in [26]. As an interesting alternative can be considered the methods based on the rank-one updates. Eirola nad Nevanlinna [12] proposed two methods of this type, one of them is mathematically equivalent to GMRES, see also [11], [48]. These methods were also studied and some GMRES generalizations were proposed by Vuik and van der Vorst (see [56], [57], [58]).

There are two distinct approaches how to reduce the amount of computational work and storage in the GMRES and related methods. The first is simply to restart the algorithm every m steps (see e.g., the methods GCR(m) and GMRES(m) presented in [14], [46]). The second approach is to truncate the orthogonalization process so that the new vector in the computed basis is orthogonalized only against the k previous vectors (not necessarily to the last ones), where k is some integer parameter. For examples of these methods, we refer to ORTHODIR(k), ORTHOMIN(k), and GCG-LS(k) (see [65], [2], [28], [14], [3]). A drawback of these techniques is that both the restarted and truncated methods are not optimal in the sense of minimization of the residual norm over the Krylov subspace $K_n(A, r^0)$. Consequently, this can lead to slower convergence. Even worse, in some cases, the restarted or truncated methods may not converge to the solution of the system. Moreover, the truncated orthogonalization may lead to the early loss of the linear independence among the computed vectors due to rounding errors.

5 Experiments

For our experiments we considered different implementations of GMRES described above and compared their behavior in the finite precision arithmetic. Test problems were taken from [32] and [61]. The matrix A and solution vector x were of the form

$$\text{TP1}(n,\alpha): \qquad A = \begin{bmatrix} 1 & 0 & 0 & \dots & \alpha \\ 0 & 2 & 0 & \dots & 0 \\ 0 & 0 & 3 & \dots & 0 \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & n \end{bmatrix}, \quad x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \cdot \\ 1 \end{bmatrix};$$

$$\text{TP2}(n,\alpha,k): \quad A = [\alpha_{ij}] \quad \alpha_{ij} = \begin{cases} \alpha^{j-i} & 0 \le j-i \le k \\ & \\ 0 & \text{otherwise} \end{cases}, \quad x = \begin{bmatrix} 1 \\ 1 \\ 1 \\ \cdot \\ 1 \end{bmatrix};$$

the right hand side vector was set to $b = Ax$. For TP1(n,α), the parameters n, α were set to $n = 100$, $\alpha = 20000$. For TP2(n,α) were used $n = 100$, $\alpha = 1.1$, $k = 25$. Computations were performed in double precision arithmetic using Matlab 4.0 on the SGI Crimson workstation with R4000 processor (machine precision $\varepsilon = 2.22 \times 10^{-16}$).

On Figures 1 and 2, we compared the relative norm of the true residual $\|b - Ax^n\|/\|r^0\|$. We considered the SS variant implemented via the classical Gram-Schmidt orthogonalization (dash-dotted line), and SS variant implemented via the modified Gram-Schmidt (solid line); the modified Gram-Schmidt implementation of the Algorithm WLZ2 (dashed line), the modified Gram-Schmidt implementation of the Algorithm A^TA-2 (o-signs), and the original implementation of ORTHODIR [14] (dotted line). Both figures illustrate that the modified Gram-Schmidt implementations of SS, WLZ2 and A^TA-2 are comparable and are superior to the classical Gram-Schmidt implementation of SS or the ORTHODIR method, that performed poorly, especially for the problem TP2.

On Figures 3 and 4, we compared the modified Gram-Schmidt implementations of the Algorithms WLZ1, WLZ2, and A^TA-1, A^TA-2. For WLZ1, we plotted the recursive computed relative norms of the residual ϱ_j (x-marks), for the WLZ2 the norms of the updated residual vectors $\hat{r}^j$ (solid line). Similarly, for the Algorithm A^TA-1, we have compared the quantities ϱ_j (o-signs) and the norm of the vectors $\hat{r}^j$ in the Algorithm A^TA-2 (dotted line). Figures 3 and 4 illustrate our conclusion from Section 3 stating that the iterative updating of the norm of the residual becomes, after some initial norm reduction, unreliable and leads to an early termination of the Algorithms WLZ1 and A^TA-1 (TP1), or to stagnation (TP2). On Fig. 3 the dotted line coincides with the solid one.

On Figures 5 and 6, the relative norms of the true residual $\|b - Ax^n\|/\|r^0\|$ are displayed for two different ways of computing the approximate solution x^j in the Algorithm WLZ2. We considered the scheme presented in [62] (solid line) and the scheme SOLVE(x^j) discussed in the Algorithm WLZ2 (o-marks). We did not find any significant difference between these two schemes and therefore we recommend the use of the considerably simpler scheme SOLVE(x^j). We have also plotted the relative norm of the true residual for the WLZ2 algorithm, where the solution of the upper triangular system (3.9) was replaced by the solution of the system (2.12) (dotted line). We observed very poor behavior, which is in a good agreement with our discussion in Section 3.1.

Experimental results presented here were chosen from a large set of experiments which had been performed. All of them were in a good agreement with our theoretical considerations described in Section 3 and with remarks on the related methods from Section 5.

6 Conclusions

The methods ORTHODIR, ORTHOMIN, and their modifications, which are based on generating the unnormalized basis vectors and on the orthogonalization equivalent to the classical Gram-Schmidt are clearly outperformed by the corresponding variants of the GMRES method. It is clear that the normalization of the basis vectors together with using modified Gram-Schmidt, or even more stable orthogonalization schemes, affect significantly the numerical stability of the implementations discussed in the previous sections. The truncation and restarting of the presented variants can be considered accordingly.

Measured by the numerical behavior, there is no clear winner between the variants of GMRES described in this paper. The choice of the particular variant depends on the computer architecture. Although A^TA-variant requires somewhat more work and storage, its implementation is simpler than of the other variants. For parallel architectures, the A^TA-variant may be worth to consider, because the SS and WLZ variants need more sequential steps and are relatively more difficult to implement.

In the [23] the numerical stability of the optimal implementation of the SS variant is studied. For a similar analysis of the optimal implementations of the WLZ and A^TA variants we refer to the forthcoming paper [40]. Another interesting question is the further development of the truncated schemes and their comparison to the other short-term approaches. We will return to this point elsewhere.

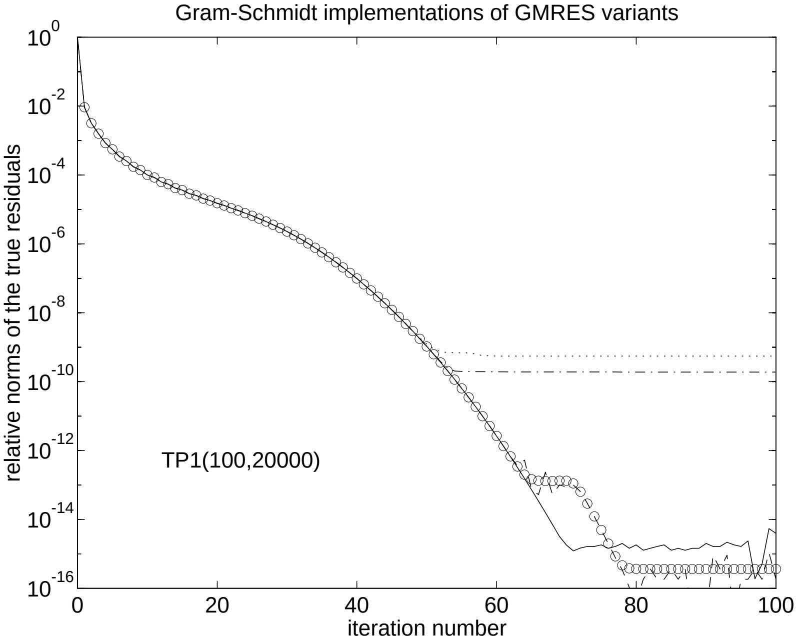
Gram-Schmidt implementations of GMRES variants
relative norms of the true residuals
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{-16}
TP1(100,20000)
0
20
40
60
80
100
iteration number

FIG. 1

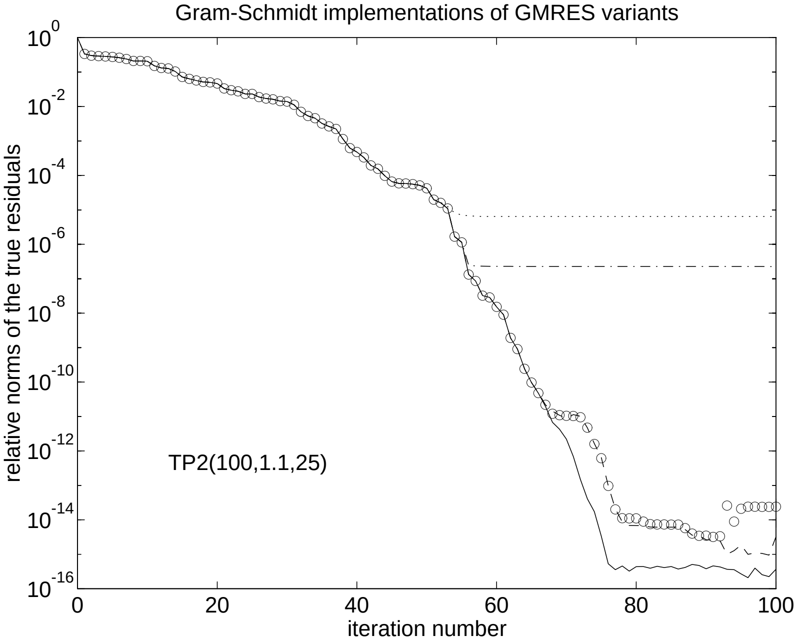
Gram-Schmidt implementations of GMRES variants
relative norms of the true residuals
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{-16}
TP2(100,1.1,25)
0
20
40
60
80
100
iteration number

FIG. 2

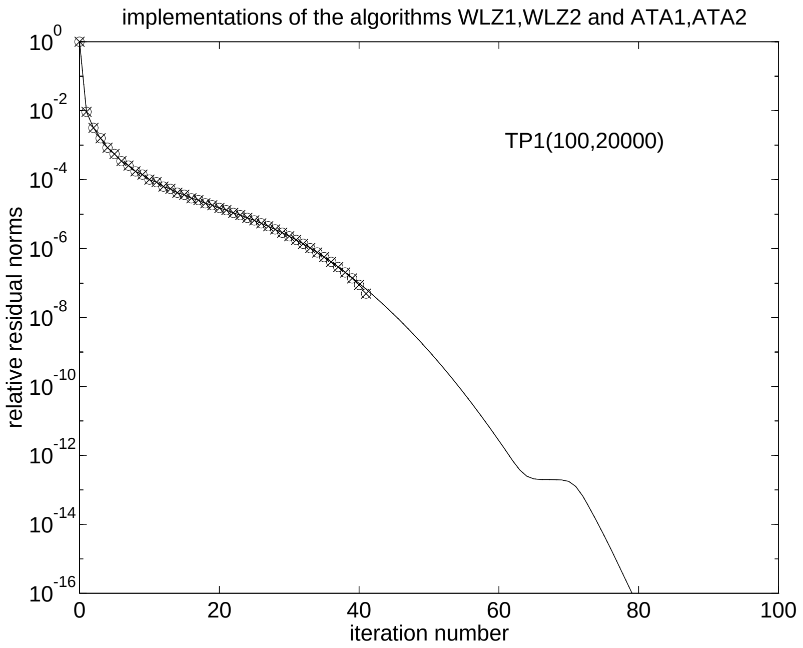
implementations of the algorithms WLZ1,WLZ2 and ATA1,ATA2
TP1(100,20000)
relative residual norms
10^0
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-14
10^-16
0
20
40
60
80
100
iteration number

FIG. 3

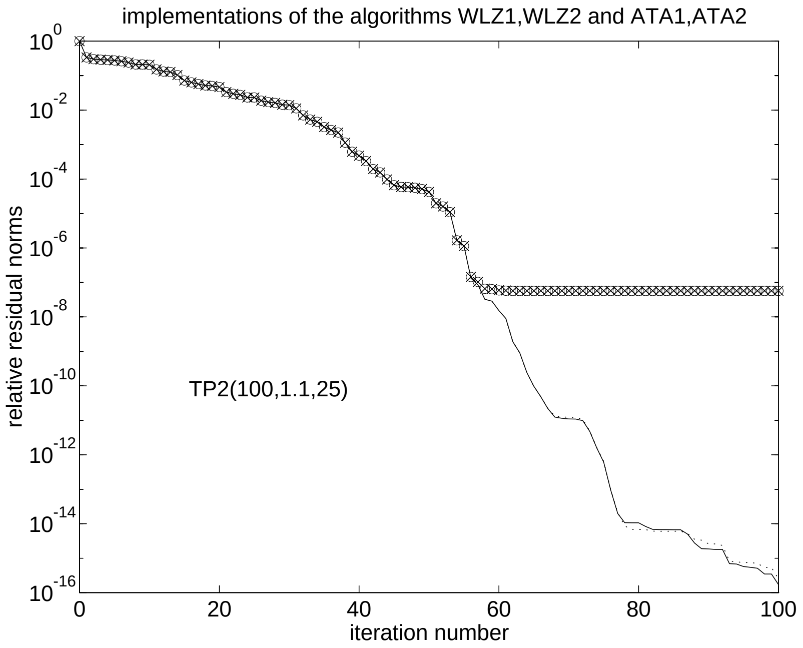
implementations of the algorithms WLZ1,WLZ2 and ATA1,ATA2
TP2(100,1.1,25)
relative residual norms
10^0
10^-2
10^-4
10^-6
10^-8
10^-10
10^-12
10^-14
10^-16
0
20
40
60
80
100
iteration number

FIG. 4

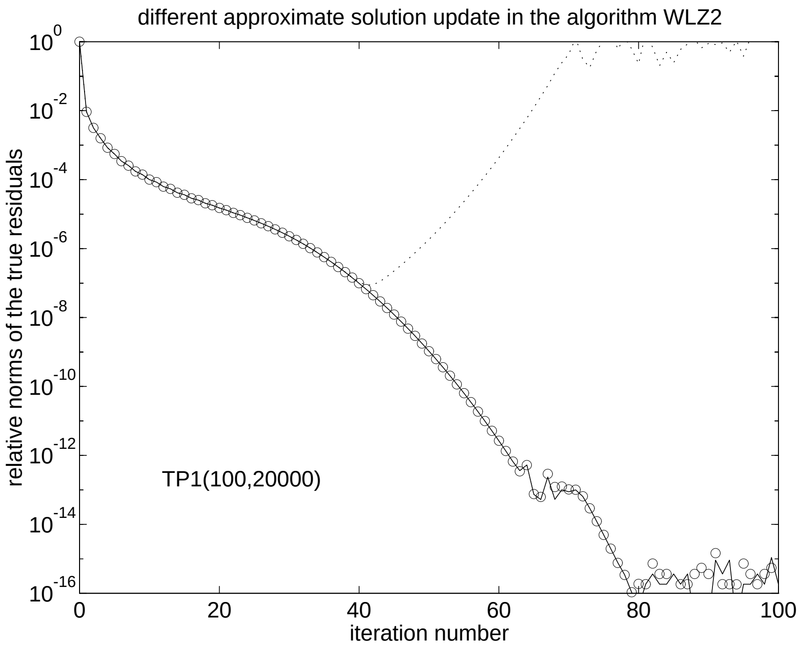
different approximate solution update in the algorithm WLZ2
relative norms of the true residuals
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{-16}
TP1(100,20000)
0
20
40
60
80
100
iteration number

FIG. 5

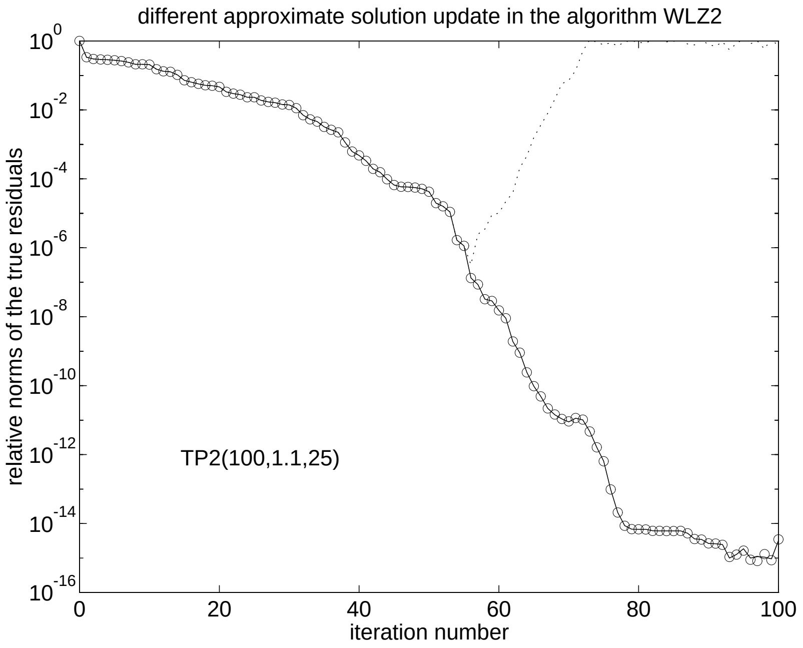
different approximate solution update in the algorithm WLZ2
relative norms of the true residuals
10^{0}
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
10^{-12}
10^{-14}
10^{-16}
TP2(100,1.1,25)
0
20
40
60
80
100
iteration number

FIG. 6

REFERENCES

1. W.E. Arnoldi, *The Principle of Minimized Iterations in the Solution of the Matrix Eigenvalue Problem*, Quart. Appl. Math. 9, pp. 17-29, 1951

2. O. Axelsson, *Conjugate Gradient Type Methods for Unsymmetric and Inconsistent Systems of Linear Equations*, Linear Algebra Appl. 29, pp. 1-16, 1980

3. O. Axelsson, *A Generalized Conjugate Gradient, Least Square Method*, Num. Math. 51, pp. 209-227, 1987

4. R. Barrett et al., *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, Philadelphia, 1994

5. A. Björck, *Solving Linear Least Squares Problems by Gram-Schmidt Orthogonalization*, BIT 7, pp. 1-21, 1967

6. A. Björck, *Conjugate Gradient Algoritms for Solving Sparse Linear Least Squares Problem*, (unpublished notes)

7. A. Björck, C.C. Paige, *Loss and Recapture of Orthogonality in the Modified Gram-Schmidt Algorithm*, SIAM J. Matrix Anal. Appl., Vol. 13, No. 1, pp. 176-190, 1992

8. P.N. Brown, *A Theoretical Comparison of the Arnoldi and GMRES Algorithms*, SIAM Jour. Sci. Stat. Comput. 12, pp. 58-78, 1991

9. E.J. Craig, *The N-step Iteration Procedures*, J. Math. Phys. 34, pp. 64-73, 1955

10. R. Chandra, *Conjugate Gradient Methods for Partial Differential Equations*, Ph.D. Thesis, Yale University, New Haven, 1978

11. J.E. Dennis, K. Turner, *Generalized Conjugate Directions*, Linear Algebra and Its Applications 88/89, pp. 187-209, 1987

12. T. Eirola, O. Nevanlinna, *Accelerating with Rank-one Updates*, Lin. Alg. and Its Appl. 121, pp. 511-520, 1989

13. S.C. Eisenstat, H.C. Elman, M.H. Schultz, *Variational Iterative Method for Nonsymmetric Systems of Linear Equations*, SIAM J. Numer. Anal., Vol. 20, No. 2, pp. 345-357, April 1983

14. H.C. Elman, *Iterative Methods for Large Sparse Nonsymmetric Systems of Linear Equations*, Research Report #229, Dept. of Computer Science, Yale University, New Haven, April 1982

15. R. Fletcher, *Conjugate Gradient Methods for Indefinite Systems*, in Numerical Analysis Dundee 1975 (G.A. Watson, ed.), Lecture Notes in Mathematics 506, Springer, Berlin, pp. 73-89, 1976

16. V.M. Fridman, *The Method of Minimum Iterations with Minimum Errors for a System of Linear Algebraic Equations with a Symmetrical Matrix*, USSR Comput. Math. and Math. Phys. 2, pp. 362-363, 1963

17. R.W. Freund, *On Conjugate Gradient Type Methods and Polynomial Preconditioners for a Class of Complex Non-Hermitian Matrices*, Numer. Math. 57, pp. 285-312, 1990

18. R.W. Freund, G.H. Golub, N.M. Nachtigal, *Iterative Solution of Linear Systems*, Acta Numerica 1, pp. 1-44, 1992

19. V. Faber, T. Manteuffel, *Necessary and Sufficient Conditions for the Existence of a Conjugate Gradient Method*, SIAM J. Numer. Anal., Vol. 21, No. 2, pp. 352-362, April 1984

20. V. Faber, T. Manteuffel, *Orthogonal Error Methods*, SIAM J. Numer. Anal. 24, pp. 170-187, 1987

21. G.H. Golub, W. Kahan, *Calculating the Singular Values and Pseudo-inverse of a Matrix* , SIAM J.Numer.Anal. 2, pp. 205-224, 1965

22. G.H. Golub, J.M. Ortega, *Scientific Computing - an Introduction with Parallel Computing*, Academic Press, New York, 1993.

23. A. Greenbaum, J. Drkošová, M. Rozložník, Z. Strakoš, *Numerical Stability of the GMRES Method*, submitted to BIT, 1994

24. A. Greenbaum, Z. Strakoš, *Predicting the Behavior of Finite Precision Lanczos and Conjugate Gradient Computations*, SIAM J. Matrix Anal. Appl. 13, No. 1, pp. 121-137, 1992

25. A. Greenbaum, Z. Strakoš, *Matrices That Generate the Same Krylov Residual Spaces*, in: Recent Advances in Iterative Methods, G.H. Golub et al. eds., IMA Volumes in Math. and its Applications, Vol. 60, Springer, pp. 95-119, 1993

26. M.H. Gutknecht, *Changing the Norm in Conjugate Gradient Type Algorithms*, SIAM J. Numer. Anal., Vol. 30, No. 1, pp. 40-56, February 1993

27. M.R. Hestenes, E. Stiefel, *Methods of Conjugate Gradients for Solving Linear Systems*, J. Res. Nat. Bur. Stand. 49, pp. 409-436, 1952

28. L.A. Hageman, D.M. Young, *Applied Iterative Methods*, Academic Press, New York, 1981

29. Y. Huang, H. van der Vorst, *Some Observations on the Convergence Behavior of GMRES*, Res.Report 89-09, TU Delft, 1989

30. W.D. Joubert, *Iterative Methods for the Solution of Nonsymmetric Systems of Linear Equations*, the University of Texas at Austin, Center for Numerical Analysis, Report CNA-242, February 1990

31. W.D. Joubert, D.M. Young, *Necessary and Sufficient Conditions for the Simplification of Generalized Conjugate Gradient Algorithms*, Linear Algebra Appl. 88/89, pp. 449-485, 1987

32. R. Karlsson, *A Study of Some Roundoff Effects of the GMRES method*, Li-TH-MAT-R-1990,11, University of Linkoping, Sweden, 1991

33. I.M. Khabaza, *An Iterative Least-Square Method Suitable for Solving Large Sparse Matrices*, Comput. J. 6, pp. 202-206, 1963

34. C. Lanczos, *An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators*, J. Res. Natl. Bur. Stand. 45, pp. 255-282, 1950

35. C. Lanczos, *Solution of Systems of Linear Equations by Minimized Iterations*, J. Res. Natl. Bur. Stand. 49, pp. 33-53, 1952

36. C.C. Paige, *Bidiagonalization of Matrices and Solution of Linear Equations*, SIAM J. Numer. Anal. 11, pp. 197-209, 1974

37. C.C. Paige, M.A. Saunders, *Solution of Sparse Indefinite Systems of Linear Equations*, SIAM J. Num. Anal. 12, No. 4, pp. 617-629, 1975

38. C.C. Paige, M.A. Saunders, *LSQR: An Algorithm for Sparse Linear Equations and Sparse Least Squares*, ACM Trans. Math. Softw. 8, pp. 43-71, 1982

39. M. Rozložník, *Variants of the Generalized Minimal Error Method*, in preparation

40. M. Rozložník, *On the Numerical Stability of GMRES Variants*, in preparation

41. T. Rusten, *Iterative Methods for Mixed Finite Element Systems*, Ph.D. Dissertation, Department of Informatics, University of Oslo, July 1991

42. Y. Saad, *Krylov Subspace Methods for Solving Large Unsymmetric Linear Systems*, Mathematics of Computation, Vol. 37, No. 155, pp. 105-126, July 1981

43. Y. Saad, *Practical Use of Some Krylov Subspace Methods for Solving Indefinite and Unsymmetric Linear Systems*, SIAM J. Sci. Statist. Comput., Vol. 5, pp. 203-228, 1984

44. Y. Saad, *Krylov Subspace Methods on Supercomputers*, SIAM J. Sci. Stat. Comput., Vol. 10, No. 6, pp. 1200-1232, November 1989

45. Y. Saad, M.H. Schultz, *Conjugate Gradient-Like Algorithms for Solving Nonsymmetric Linear Systems*, Mathematics of Computation, Vol. 44, No. 170, pp. 417-424, April 1985

46. Y. Saad, M.H. Schultz, *GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems*, SIAM J. Sci. Stat. Comput. 7, pp. 856-869, 1986

47. H. Simon, *The Lanczos Algorithm for Solving Symmetric Linear Systems*, Ph.D. Thesis, University of California, Berkeley 1982

48. E. Spedicato, Z. Gao, H. Yu, *The Scaled ABS Versions of Some Conjugate Gradient Methods for Non symmetric Linear Systems*, Research Rep. 19, Dept. of Mathematics, University of Bergamo, 1990

49. E. Stiefel, *Relaxations methoden bester Strategie zur Losung linearer Gleichungssysteme*, Comm. Math. Helv. 29, pp. 157-179, 1955

50. J. Stoer, *Solution of Large Linear Systems of Equations by Conjugate Gradient Type Methods*, in Mathematical Programming - The State of the Art (A. Bachem, M. Grotschel and B. Korte eds.), Springer, Berlin, pp. 540-565,1983

51. J. Stoer, R. Freund, *On the Solution of Large Indefinite Systems of Linear Equations by Conjugate Gradients Algorithm*, in Computing Methods in Applied Sciences and Engineering V, (R.Glowinski, J.L.Lions eds.), North Holland - INRIA, pp. 35-53, 1982

52. Z. Strakoš, *On the Real Convergence Rate of the Conjugate Gradient Method*, Lin. Alg. and Its. Appl. 154-156, pp. 535-549, 1991

53. K. Turner, H.F. Walker, *Efficient High Accuracy Solutions with GMRES(m)*, Res. rep. 90/50 (revised Jan. 91), Utah State Univ., 1991

54. P. Vinsome, *Orthomin, an Iterative Method for Solving Sparse Sets of Simultaneous Linear Equations*, in Proceedings of the Fourth Symposium on Reservoir Simulation, Society of Petroleum Engineers of AIME, Los Angeles, 1976

55. V.V. Voevodin, *The Problem of a Non-Selfadjoint Generalization of the Conjugate Gradient Method Has Been Closed*, USSR Comput. Math. and Math. Phys. 23, pp. 143-144, 1983

56. H.A. van der Vorst, C. Vuik, *The Superlinear Convergence Behaviour of GMRES*, Preprint, Utrecht University, 1992

57. C. Vuik, H.A. van der Vorst, *A Comparison of Some GMRES-like Methods*, Linear Alg. Appl. 160, pp.131-162, 1992

58. H.A. van der Vorst, C. Vuik, *GMRESR: A Family of Nested GMRES Methods*, Preprint, Utrecht University, 1992.

59. H.A. van der Vorst, *Lecture Notes on Iterative Methods*, University of Utrecht, 1993.

60. H.F. Walker, *Implementation of the GMRES Method Using Householder Transformations*, SIAM J. Sci. Stat. Comput., Vol. 9, No. 1, pp. 152-163, 1988

61. H.F. Walker, *Implementations of the GMRES Method*, Computer Physics Communications 53, North-Holland, Amsterdam, pp. 311-320, 1989

62. H.F. Walker, Lu Zhou, *A Simpler GMRES*, Research Report 10/92/54, Dept. of Mathematics and Statistics, Utah State University, Logan, 1992

63. R. Weiss, *Error-Minimizing Krylov Subspace Methods*, SIAM J. Scientific Comput. 15(3), 1994

64. R. Weiss, *Do Not Trust the Residual of Iterative Linear Solvers*, in Proceedings of the Third Conference on Nonlinear Analysis, Numerical Analysis and Applied Mathematics, Constanta, Romania, May 1993

65. D.M. Young and K.C. Jea, *Generalized Conjugate Gradient Acceleration of Nonsymmetrizable Iterative Methods*, Linear Algebra Appl. 34, pp. 159-194, 1980