



On the Performance of Various Adaptive Preconditioned GMRES Strategies

Kevin Burrage, Jocelyne Erhel

► To cite this version:

Kevin Burrage, Jocelyne Erhel. On the Performance of Various Adaptive Preconditioned GMRES Strategies. [Research Report] RR-3098, INRIA. 1997. inria-00073593

HAL Id: inria-00073593

<https://inria.hal.science/inria-00073593>

Submitted on 24 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

***On the performance of various adaptive
preconditioned GMRES strategies***

Kevin Burrage , Jocelyne Erhel

N° 3098

Février 1997

_____ THÈME 4 _____



***rapport
de recherche***

On the performance of various adaptive preconditioned GMRES strategies

Kevin Burrage ^{*}, Jocelyne Erhel [†]

Thème 4 — Simulation et optimisation
de systèmes complexes
Projet ALADIN

Rapport de recherche n° 3098 — Février 1997 — 20 pages

Abstract: This paper compares the performance on linear systems of equations of three similar adaptive accelerating strategies for restarted GMRES. The underlying idea is to adaptively use spectral information gathered from the Arnoldi process. The first strategy retains approximations to some eigenvectors from the previous restart and adds them to the Krylov subspace. The second strategy uses also approximated eigenvectors to define a preconditioner at each restart. This paper designs a third new strategy which combines elements of both previous approaches. Numerical results show that this new method is both more efficient and more robust.

Key-words: GMRES, preconditioning, invariant subspace, deflation.

(Résumé : tsvp)

This work was partially funded under Project 95/DIST001G from the Department of Industry Science and Technology in Australia.

^{*} Department of Mathematics, the University of Queensland, Queensland 4072, Australia, kb@maths.uq.oz.au, <http://www.maths.uq.oz.au/~kb>

[†] INRIA, UR Rennes, erhel@irisa.fr, <http://www.irisa.fr/aladin/perso/erhel.html>

Performance de quelques stratégies de préconditionnement de GMRES

Résumé : Ce rapport compare les performances pour résoudre des systèmes linéaires d'équations de trois stratégies adaptatives qui accélèrent de façon similaire l'algorithme de GMRES avec redémarrage. L'idée sous-jacente est d'utiliser dynamiquement l'information spectrale obtenue par le procédé d'Arnoldi. La première stratégie calcule des approximations de vecteurs propres à l'aide du redémarrage précédent et les ajoute à l'espace de Krylov. La seconde stratégie utilise aussi des approximations de vecteurs propres pour définir un preconditionnement à chaque redémarrage. Ce rapport présente une nouvelle et troisième stratégie qui combine des éléments des deux premières. Des résultats numériques montrent que cette nouvelle méthode est à la fois plus efficace et plus robuste.

Mots-clé : GMRES, preconditionnement, sous-espace invariant, déflation.

1 Introduction

The GMRES algorithm [15] is now a standard iterative method for solving large sparse nonsymmetric linear systems of equations

$$Ax = b, \quad x \in \mathbb{R}^n. \quad (1)$$

It uses an Arnoldi algorithm to build an orthonormal basis for Krylov subspace given by

$$K_m(A, r) = \text{Span}\{r, Ar, \dots, A^{m-1}r\}$$

in which an approximation to the solution of (1) lies. Clearly if exact arithmetic is used then the GMRES algorithm will converge in at most n iterations. Since the GMRES algorithm is expensive both computationally and in its memory requirements a restarted version (GMRES(m)) is often used in which the Krylov subspace is restricted to be of fixed dimension m and the Arnoldi process is restarted using the last iterate x_m as a new initial approximation for the restart. Unfortunately, it can be very difficult to know how to choose m apriori and if too small a value is chosen, convergence may stall.

It is known that small eigenvalues of A can slow down convergence. More recent work on the convergence behaviour of GMRES [1] relates the superlinear convergence to the convergence of Ritz values. Basically convergence occurs as if at each iteration of GMRES the next smallest eigenvalue in magnitude is removed from the system. This work has some relation to that of Schropp and Keller [18] and Burrage et al. [13], who show that standard stationary iteration schemes such as Jacobi and SOR can be rapidly accelerated by a deflation process in which the largest eigenvalues in magnitude of the amplification matrix (estimated by the power method) are periodically removed into a coupled stiff subspace (where the small stiff component is solved for by a direct method). This leads to a coupled iteration between the non-stiff and stiff subspace – the effect of which is that the fixed-point iterative scheme is rapidly accelerated.

Unfortunately, if a restarted GMRES procedure is used the information about the smallest eigenvalues and corresponding eigenvectors is lost at each restart and so the superlinear convergence may be lost. For this reason, recently, researchers have examined different ways of reducing the negative effects of a restart.

There are essentially two ways that this can be done: an augmented subspace approach [2, 17] and a preconditioning approach [3, 7]. The first solution retains vectors from the previous restart and adds them to the new subspace. In particular, it investigates saving k approximate eigenvectors of A corresponding to the k smallest eigenvalues in magnitude. These eigenvectors are estimated using a Rayleigh-Ritz method. It should be noted here that after each restart the k eigenvectors become progressively more accurate.

The second approach and the one tested in [3] and [7] is to adaptively build a preconditioner for GMRES based on spectral information gathered from the Arnoldi process during iteration by restarted GMRES. The nice feature of this approach is that the advantages of preconditioning are retained without the difficulty of knowing what type of preconditioner to use. In the case of [3], after each restart the preconditioner is updated by extracting

new eigenvalues corresponding to the remaining smallest eigenvalues which are smallest in magnitude. This is different from the flexible GMRES method [16] because it executes a true GMRES cycle with a constant preconditioner inside the cycle. The preconditioner is equal to the projected matrix onto the approximated invariant subspace (up to a scaling factor) and is taken as the identity on the orthogonal subspace.

Note that the preconditioner can be used on the left or the right with equal facility. The approach in [7] has a similar thrust to that in [3] and the same convergence results hold. Here they precondition on the left and the implementation uses the recurrence formulas of the Implicitly Restarted Arnoldi method [4] and [5] which implies that the application of a preconditioner does not require additional evaluation of any matrix-vector products. These recurrence formulas are truncated versions of the recurrence formulas for the QR algorithm with explicit shifts. This preconditioner is built at each restart and is applied to all the other preconditioners to give a composition effect.

The numerical results in these papers show the efficacy of these methods compared with standard restarted GMRES. However, there are still difficulties in knowing how to choose k and how many eigenvalues should be deflated at each restart. This paper addresses these issues and attempts to develop a method which combines elements of the augmented subspace solution and the automatic preconditioning approach. Thus in section 2 both methods are described while in section 3 a new method combining elements of both is developed. Finally in section 4 numerical results are presented on four test problems: two of which come from specific real-world applications while the other two problems test the efficacy of deflation in terms of eigenvalue distribution and the conditioning of the eigenvectors. The paper concludes with a summary of the work in this paper and future directions.

2 Deflated GMRES

2.1 Restarted GMRES(m)

Let x_0 an initial guess and $r_0 = b - A * x_0$ with $\beta = \|r_0\|$. Let V_m be an orthonormal basis of the Krylov subspace $K_m(A, r_0)$ such that

$$AV_m = V_{m+1}\overline{H}_m \quad (2)$$

with \overline{H}_m a Hessenberg matrix of order $(m+1) \times m$. Algorithm GMRES(m) computes $x_m = x_0 + V_m y_m$ such that

$$\|b - Ax_m\| = \min_{y \in \mathbb{R}^m} \|\beta e_1 - \overline{H}_m y\|. \quad (3)$$

It is known that the convergence of GMRES is similar to conjugate gradients for nearly normal systems in that small eigenvalues of A can slow convergence, but for highly non-normal systems the convergence behaviour is more complicated.

In the sequel, we assume that all eigenvalues of A are non defective, or in other words that A is diagonalizable in \mathbb{C} . Let $|\lambda_1| \leq |\lambda_2| \leq \dots \leq |\lambda_n|$ be the eigenvalues of A . The

following result is well-known but we give it here along with a proof because it will help to prove the new results about our deflation methods.

Proposition 2.1 *If*

$$A = Z\Omega Z^{-1}, \quad (4)$$

with $\Omega = \text{diag}(\lambda_1, \dots, \lambda_n)$ then after a cycle of GMRES(m)

$$\|r_m\| \leq \|r_0\| \text{cond}(Z) \min_{q \in \Pi_m^0} \max_{1 \leq i \leq n} |q(\lambda_i)|, \quad r_m = b - Ax_m, \quad (5)$$

where Π_m^0 is the set of polynomials q of degree at most m with $q(0) = 1$ and $\text{cond}(Z) = \|Z\| \|Z^{-1}\|$.

Proof. Let $r_0 = \sum_{i=1}^n \beta_i z_i = Z\beta$. Then $r_m = q(A)r_0$ with $q \in \Pi_m^0$ so that

$$r_m = \sum_{i=1}^n \beta_i q(\lambda_i) z_i. \quad (6)$$

It follows $\|r_m\| \leq \|\beta\| \|Z\| \max_{1 \leq i \leq n} |q(\lambda_i)|$, with $\beta = (\beta_i), i = 1, \dots, n$. But $r_0 = Z\beta$ implies $\beta = Z^{-1}r_0$ and $\|\beta\| \leq \|Z^{-1}\| \|r_0\|$.

This result, as well as various experiments, shows that the smallest eigenvalues slow down the convergence. As noted in the introduction, superlinear convergence is lost when restarting GMRES(m) because the Ritz values are lost. On the other hand, the Arnoldi process can be used to compute Ritz values and Ritz vectors and to estimate eigenvalues and eigenvectors of A .

The idea underlying deflation methods is to annihilate the components of the residual r_m on the eigenvectors z_i corresponding to the smallest eigenvalues. The method to achieve this goal is to estimate these eigenvectors at each restart and to use them at the next restart. The solution developed in [2] is to augment the Krylov subspace with these estimated eigenvectors whereas the solution chosen in [3] and [7] is to define a preconditioner. In the first solution, the components of the residual are killed at the end of the cycle while the preconditioner will kill them at each internal iteration.

2.2 Adding eigenvectors

We now describe the augmented subspace method due to [2] and experimented in [17]. The method will be denoted MORGAN(m, k) where m will be the dimension of the Krylov subspace and k the maximal dimension of the invariant subspace so that the solution will be approximated in an augmented subspace of dimension $m + l$ with $l \leq k$. Let V_m be an orthonormal basis of $K_m(A, r_0)$, U a basis of the approximated invariant subspace and

$W = (V_m, U)$ the basis of the augmented subspace. GMRES requires an orthogonalisation of AW , which is obtained through a modified-Gram-Schmidt algorithm, so that

$$AW = V\overline{H}, \quad (7)$$

where $V = (V_{m+1}, V_l)$ is orthogonal of size $n \times (m+1+l)$ and H is a Hessenberg matrix of size $(m+1+l) \times (m+l)$.

Algorithm MORGAN(m,k) computes $x_m = x_0 + Wy_m$ such that

$$\|b - Ax_m\| = \min_{y \in \mathbb{R}^{m+k}} \|\beta e_1 - \overline{H}y\|. \quad (8)$$

ALGORITHM : MORGAN(m,k)

ϵ is the tolerance for the residual norm ;

convergence:= false ;

choose x_0 ;

$U := \{\}$;

$l := 0$;

until convergence **do**

$r_0 = b - Ax_0$;

 Arnoldi process applied to A to compute V_m ;

$W = (V_m, U)$;

 compute AU ;

 orthogonalize AW to get V ;

$y_m = \operatorname{argmin}_{y \in \mathbb{R}^{m+l}} \|\beta e_1 - \overline{H}y\|$;

$x_m := x_0 + Wy_m$;

if $\|b - Ax_m\| < \epsilon$ convergence := true ;

else

$x_0 = x_m$;

 choose $l \leq k$;

 estimate l eigenvectors U of A ;

endif

enddo

The choice of l and the computation of U are described later on.

2.3 Preconditioning

We now describe the preconditioning approach due to [3] which is also developed in a slightly different way in [7]. Now M will denote a preconditioning matrix which will be updated at each restart and V_m is an orthonormal basis of the Krylov subspace $K_m(r_0, AM^{-1})$. The algorithm, denoted DEFLGMRES(m,k), applies GMRES(m) to the matrix AM^{-1} . We thus have

$$AM^{-1}V_m = V_{m+1}\overline{H}_m \quad (9)$$

where \overline{H}_m is a Hessenberg matrix of size $(m+1) \times m$.

Algorithm DEFLGMRES(m,k) computes $x_m = x_0 + M^{-1}V_m y_m$ such that

$$\|b - Ax_m\| = \min_{y \in \mathbb{R}^m} \|\beta e_1 - \overline{H}_m y\|. \quad (10)$$

The preconditioner M^{-1} is defined by

$$M^{-1} := I_n + U(|\lambda_n|T^{-1} - I_l)U^T \quad (11)$$

where U is a set of $l \leq k$ orthonormal vectors and $T = U^T A U$. In practice, U will approximate the basis of an invariant subspace.

ALGORITHM : DEFLGMRES(m,k)

ϵ is the tolerance for the residual norm ;

convergence := false ;

choose x_0 ;

$M := I_n$;

$U := \{\}$;

$l := 0$;

until convergence **do**

$r_0 = b - Ax_0$;

Arnoldi process applied to AM^{-1} to compute V_m ;

$y_m = \operatorname{argmin}_{y \in \mathbb{R}^m} \|\beta e_1 - \overline{H}_m y\|$;

$x_m := x_0 + M^{-1}V_m y_m$;

if $\|b - Ax_m\| < \epsilon$ convergence := true ;

else

$x_0 = x_m$;

choose $l \leq k$;

estimate l eigenvectors U of A ;

compute $T := U^T A U$;

$M^{-1} := I_n + U(|\lambda_n|T^{-1} - I_l)U^T$;

endif

enddo

2.4 Estimating eigenvectors

Now we deal with the estimation of U , assuming l has been chosen. A first solution to estimate eigenvectors, used in [3], is to solve the oblique problem

$$V_m^*(AM^{-1} - \theta I)V_m u = 0$$

which reduces to the ordinary eigenvalue problem

$$H_m u = \theta u, \quad U = V_m u, \quad (12)$$

where H_m is the Hessenberg matrix obtained from \overline{H}_m by removing the last row. The classical QR algorithm for Hessenberg matrices, implemented for example in the Lapack library, computes all the eigenvalues and eigenvectors which are then sorted to keep the l smallest. Some precaution is required to keep together complex conjugate eigenvalues.

A second solution, which is used in the augmented subspace approach [2, 17], is to solve the harmonic problem

$$(AW)^*(A - \theta I)Wu = 0$$

which reduces to the generalized eigenvalue problem

$$Ru = \theta[QV^*W]u, \quad U = Wu, \quad (13)$$

with $Q\overline{H} = \begin{pmatrix} R \\ 0 \end{pmatrix}$ where Q is a unitary matrix of dimension $(m+1+l) \times (m+1)$ and R is an upper triangular matrix of order $m+l$ and $[QV^*W]$ is the matrix of order $m+l$ composed of the first $m+l$ rows of QV^*W . Here too, the generalized eigenproblem is solved by a classical procedure, as implemented in the library Lapack, and the eigenvectors are sorted with attention paid to complex conjugate eigenvalues.

In [17], it is found that the harmonic projection is more accurate than the oblique or the orthogonal projections. This harmonic projection can be used also for the preconditioning approach as will be seen in section 3.

A third solution, as used in [7], is to estimate eigenvectors by the Implicitly Restarted Arnoldi method, but this will not be studied here.

Now various strategies have been implemented to choose the number l of estimated eigenvectors. In all cases, m and k are chosen in advance. In [2] and [17], $l = k$ at each restart. In [3], l increases from 0 to k with *freq* vectors added at each restart. The same strategy occurs in [7] but only when the vectors are accurate enough.

2.5 Theoretical convergence

Both deflated GMRES aim at annihilating the components of the residual r_m in the eigenvectors corresponding to the smallest eigenvalues. In this part, we assume that U is the set of exact eigenvectors z_1, \dots, z_k .

The augmented subspace approach leads to the following result, which generalizes the result stated in [2].

Proposition 2.2 *Let $U = Z_1 = (z_1, z_2, \dots, z_k)$ and $Z_2 = (z_{k+1}, z_{k+2}, \dots, z_n)$ the eigenvectors of A corresponding to the eigenvalues $\lambda_1, \dots, \lambda_n$ ordered by $|\lambda_1| \leq \dots \leq |\lambda_n|$. The residual r_m computed with $MORGAN(m, k)$ (and $l=k$) satisfies*

$$\|r_m\| \leq \|r_{0,2}\| \min_{q \in \Pi_m^0} \max_{k+1 \leq i \leq n} |q(\lambda_i)| \text{cond}(Z_2) \quad (14)$$

where $r_0 = \sum_{i=1}^n \beta_i z_i$, $r_{0,2} = \sum_{i=k+1}^n \beta_i z_i$ and $\text{cond}(Z_2) = \|Z_2\| \|(Z_2^* Z_2)^{-1} Z_2^*\|$.

Proof. For MORGAN(m,k) we get, as shown in [2],

$$r_m = \sum_{i=k+1}^n \beta_i q(\lambda_i) z_i$$

so that

$$\|r_m\| \leq \|\beta_2\| \|Z_2\| \max_{k+1 \leq i \leq n} |q(\lambda_i)|$$

with $\beta_2 = (\beta_i)_{i=k+1, \dots, n}$. Now since $r_{0,2} = Z_2 \beta_2$, it follows $\beta_2 = (Z_2^* Z_2)^{-1} Z_2^* r_{0,2}$ and $\|\beta_2\| \leq \|(Z_2^* Z_2)^{-1} Z_2^*\| \|r_{0,2}\|$ and the proof is complete.

The preconditioning approach leads to the following result, which generalizes the result stated in [7].

Proposition 2.3 *Let $U = Z_1 = (z_1, z_2, \dots, z_k)$ be the eigenvectors of A and AM^{-1} corresponding respectively to the eigenvalues $\lambda_1, \dots, \lambda_k$ and $|\lambda_n|, \dots, |\lambda_n|$. Let $Y_2 = (y_{k+1}, y_{k+2}, \dots, y_n)$ be the eigenvectors of AM^{-1} corresponding to the eigenvalues $\lambda_{k+1}, \dots, \lambda_n$ where the eigenvalues are ordered by $|\lambda_1| \leq \dots \leq |\lambda_n|$. The residual r_m computed with DEFLGMRES(m,k) (and $l=k$) satisfies*

$$\|r_m\| \leq \min_{q \in \Pi_m^0} (q(|\lambda_n|) \|r_{0,1}\| + \max_{k+1 \leq i \leq n} |q(\lambda_i)| \|r_{0,2}\| \text{cond}(Y_2)) \quad (15)$$

where $r_0 = r_{0,1} + r_{0,2}$ with $r_{0,1} = \sum_{i=1}^k \beta_i z_i$, $r_{0,2} = \sum_{i=k+1}^n \beta_i y_i$ and $\text{cond}(Y_2) = \|Y_2\| \|(Y_2^* Y_2)^{-1} Y_2^*\|$.

Proof. For DEFLGMRES(m,k) we get

$$r_m = q(AM^{-1}) r_0.$$

The eigenvalues of AM^{-1} are $|\lambda_n|$ with multiplicity k and $\lambda_{k+1}, \dots, \lambda_n$. Hence we get

$$r_m = q(|\lambda_n|) \sum_{i=1}^k \beta_i z_i + \sum_{i=k+1}^n \beta_i q(\lambda_i) y_i$$

so that

$$\|r_m\| \leq q(|\lambda_n|) \|r_{0,1}\| + \|\beta_2\| \|Y_2\| \max_{k+1 \leq i \leq n} |q(\lambda_i)|$$

The result follows using the same proof as before.

Theoretical convergence results are thus similar. These results assume that $U = Z_1$ is exact. However, an approximate set of eigenvectors can still have a beneficial effect on convergence. The formula (14) can be modified for approximate eigenvectors by the addition of a second term to the righthandside [2]. Experimental convergence results will depend on the way the eigenvectors are estimated and less strongly on the strategy to choose $l \leq k$.

As in [7], this analysis ignores the effect of pseudospectra on the convergence [14]. Clearly this preconditioner will be particularly effective if there is a cluster of eigenvalues that have a large deleterious influence on convergence.

2.6 Memory requirements and complexity

Now let us examine memory requirements and arithmetic complexity. Both methods have the same memory requirements which are at least $(m + 1 + k)$ vectors and preferably $(m + 1 + 2k)$ vectors to compare with $(m + 1)$ vectors for restarted GMRES(m). Indeed MORGAN(m, k) requires the storage of W and preferably V_l while DEFLGMRES(m, k) requires to store V_{m+1}, U and preferably AU .

Both methods have also roughly the same arithmetic complexity. Indeed, the bulk is a Arnoldi process, completed either by an orthogonalisation of l vectors against m other vectors or by a preconditioning which amounts to m orthogonalisations of a vector against l vectors. Also both methods require to estimate l eigenvectors U and to compute AU .

3 New deflation method

In [17], it appears to be more accurate to estimate eigenvectors by an harmonic projection than by an oblique projection. Therefore we implemented this method in our preconditioning deflation. The resulting code is called HARMONIC(m, k). We thus solve the problem

$$(AM^{-1}V_m)^*(A - \theta I)M^{-1}V_mu = 0$$

which reduces to a similar generalized eigenvalue problem

$$R_mu = \theta[Q_m V_{m+1}^* M^{-1} V_m]u, \quad U = M^{-1} V_mu, \quad (16)$$

with $Q_m \overline{H} = \begin{pmatrix} R_m \\ 0 \end{pmatrix}$ where Q_m is a unitary matrix of dimension $(m + 1) \times (m + 1)$ and R_m is an upper triangular matrix of order m .

Numerical experiments show a significant improvement of HARMONIC(m, k) over DEFLGMRES(m, k). However, for some test cases, MORGAN(m, k) performs much better. This is due to the fact that it updates the k vectors U at each restart so that they become more and more accurate approximations of eigenvectors. On the other hand, DEFLGMRES(m, k) stops estimating eigenvectors when U is large enough so that it retains sometimes poor approximations.

Therefore we designed a new preconditioning deflation which continuously updates the vectors in U . The new code is called DEFLATION(m, k). The strategy can be decomposed into two steps: the first step is as in DEFLGMRES, indeed our algorithm still computes up to k vectors at a frequency *freq*; the second step updates U at each restart using the strategy described below.

Let U_0 be k approximate eigenvectors used in the cycle. At the end of the cycle, the algorithm will compute *freq* new vectors using the harmonic projection (16). After orthogonalisation, we obtain a new set U_1 of $k + \text{freq}$ eigenvectors. The new deflation method updates them by solving a new generalized eigenproblem and by sorting k approximate eigenvectors. More precisely, we solve the problem

$$(AU_1)^*(A - \theta I)U_1u = 0$$

which is rewritten as the generalized eigenvalue problem

$$(AU_1)^* AU_1 u = \theta (AU_1)^* U_1 u, \quad U_2 = U_1 u. \quad (17)$$

After solving this problem, we sort the k smallest eigenvalues out of $k + freq$ and orthogonalize the corresponding k vectors to get u_2 and $U_2 = U_1 u_2$. This set of orthonormal vectors U_2 is the new set of k approximate eigenvectors which are used to build the new preconditioner for the next cycle. We thus get the following algorithm :

```

ALGORITHM : DEFLGMRES(m,k)
 $\epsilon$  is the tolerance for the residual norm ;
convergence:= false ;
choose  $x_0$  ;
 $M := I_n$  ;
 $U := \{\}$  ;
 $l := 0$  ;
until convergence do
   $r_0 = b - Ax_0$  ;
  Arnoldi process applied to  $AM^{-1}$  to compute  $V_m$  ;
   $y_m = \operatorname{argmin}_{y \in \mathbb{R}^m} \|\beta e_1 - \overline{H_m} y\|$  ;
   $x_m := x_0 + M^{-1} V_m y_m$  ;
  if  $\|b - Ax_m\| < \epsilon$  convergence := true ;
  else
     $x_0 = x_m$  ;
    estimate  $freq$  new eigenvectors  $U_1$  of  $A$  using (16) ;
     $U := \operatorname{orthog}(U, U_1)$  ;
     $l = l + freq$  ;
    if  $l > k$  do
      estimate  $k$  eigenvectors  $U$  of  $A$  using (17) ;
    endif ;
    compute  $T := U^T A U$  ;
     $M^{-1} := I_n + U(|\lambda_n| T^{-1} - I_l) U^T$  ;
  endif
enddo

```

4 Numerical Results

In this section numerical results will be presented that compare the methods described in the previous sections on a variety of test problems and in both a workstation and parallel computing environment. The first two classes of problems can be considered to be somewhat artificial but have been used, in modified form, by various authors ([2],[3] and [7]) in order to investigate convergence in terms of both the spectrum and conditioning of the eigenvectors.

The first class of problems allows for various sets of real eigenvalues which are spaced in different arithmetic progressions in the different sets, while the second set of problems give rise to a mixture of real eigenvalues and eigenvalues arranged on the boundary of some circle in the complex plane.

The third linear system arises when solving a biharmonic problem on a square when a B-spline (cubic basis) is used for an orthogonal collocation discretization. Some pre-pivoting is used for several of the last rows which corresponds to the boundary points ([8]). Biharmonic equations arise in a number of areas such as the modelling of thin plate mechanics and in Stokes flow. Hermite bicubic orthogonal collocation discretizations produce fourth-order accuracy for both uniform and non-uniform meshes. Previous techniques for solving these problems include FFTs and a direct approach ([8]) - which uses the fact that the problem can be represented as a coupled Poisson equation, that is a 2 block process.

The fourth problem comes from an Australian application which deals with the fitting of surfaces to various climate data gathered from the Australian continent (see [6] for more details). In particular, rainfall data from up to 6,000 irregularly scattered weather stations is processed every week in order to produce rainfall surfaces for the whole of Australia with the view to evaluating the effects of drought on agricultural production [10]. A Generalised Cross Validation process takes place in which thin plate splines are fitted to the rainfall data. This requires the solution of a sequence of linear systems of the form

$$(A + \lambda I)y = b,$$

where $\lambda > 0$ is the surface fitting parameter which is minimised in the GCV process [12]. While this system of equations is always positive definite as λ approaches its minimum value many of the eigenvalues are close to zero. The data set that is chosen represents 1080 data points chosen across the state of Queensland.

Problem 1:

Let $S(n, n_1, \beta) = (1, \beta)$ be a bidiagonal matrix of order n with 1 on the diagonal and 0 on the upper diagonal from elements 1 to n_1 and β from elements $n_1 + 1$ on, and also let $D(n, p, r)$ be a diagonal matrix of order n with values ranged from p to r in arithmetic progression. Let the system matrix of dimension N then be given by

$$A = S(N, n_1, \beta) \begin{pmatrix} D(n, p, r) & 0 \\ 0 & D(N - n, p_1, r_1) \end{pmatrix} S(N, n_1, \beta)^{-1}$$

and the right hand side be the unit vector. By choosing $N, n, n_1, p, p_1, r, r_1$ in different ways we are able to investigate the effect of various types of clustering on our adaptive algorithms. Furthermore, by allowing β to vary we are also able to investigate the effect of conditioning on the convergence of the algorithms.

The results on Problem 1 for deflated GMRES, Morgan and our new deflation method are given in Figures 1-5. In what follows the notation DEFLATION(16,4), for example, will denote that a restart of 16 is used with at most 4 eigenvalues being deflated in total. In

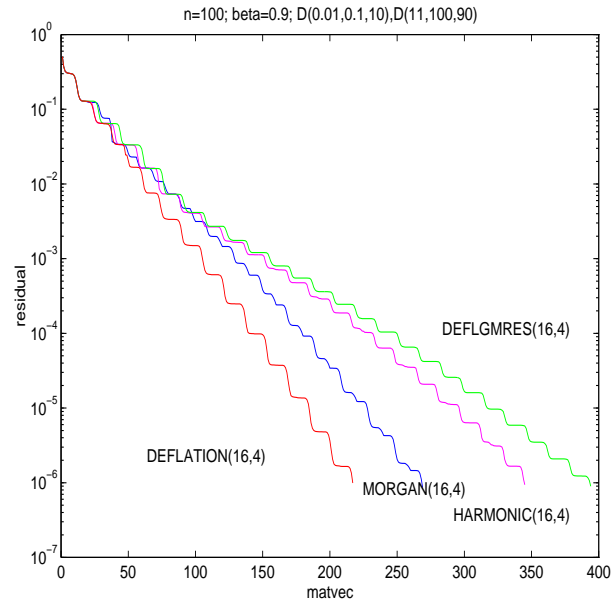


Figure 1: Comparison of implementation strategies

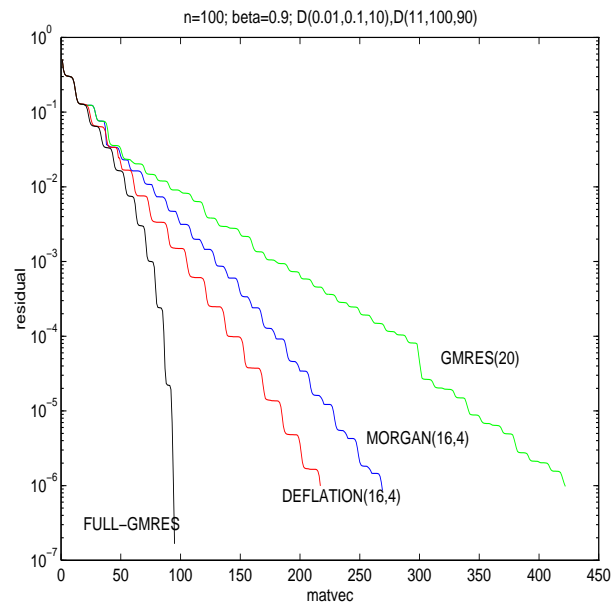


Figure 2: Comparison of Full and restarted GMRES with deflation

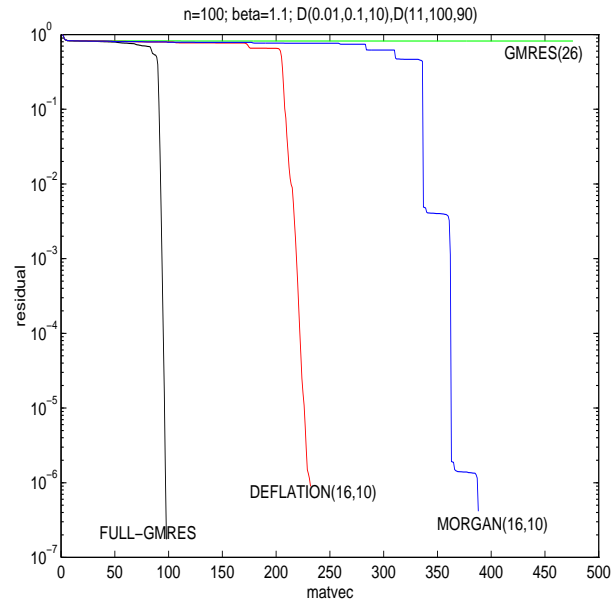


Figure 3: Convergence rates for ill-conditioned systems

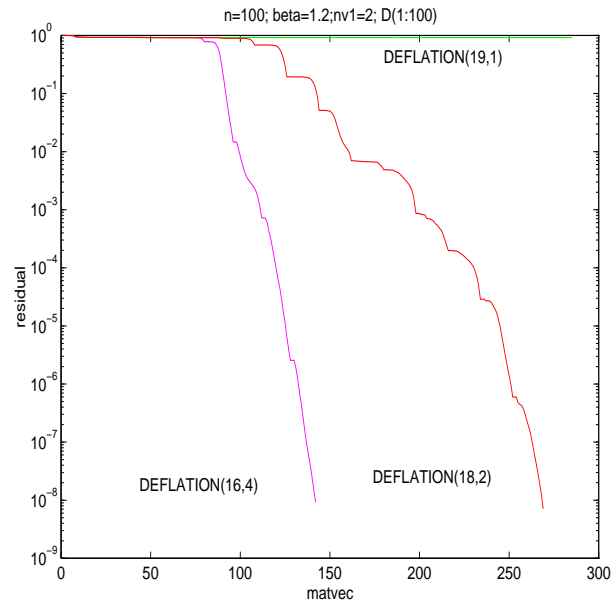


Figure 4: Convergence rates for severely ill-conditioned systems

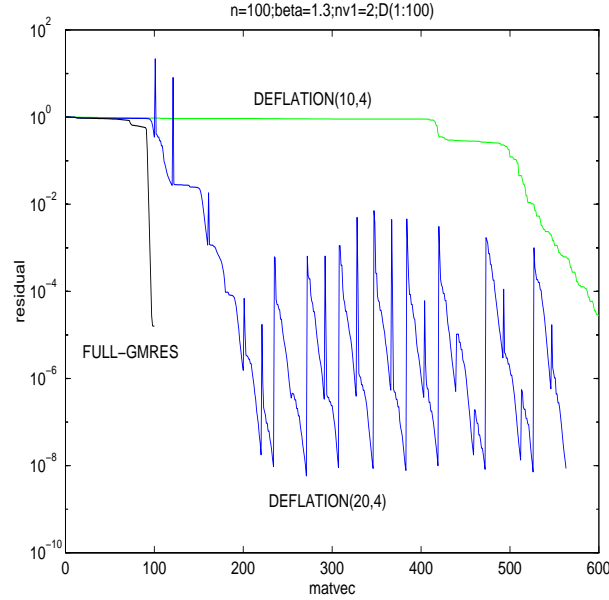


Figure 5: Convergence rates for variably coupled severely ill-conditioned systems

most cases the number of eigenvalues which can be deflated at any given time is 2 unless stated otherwise. It is important to recall that in all of these implementations eigenvalues are only deflated after each restart.

Figure 1 essentially compares the performance of the original deflation implementation (DEFLGMRES) with the same implementation in which a harmonic estimation of the eigenvalues is used rather than the Power method (HARMONIC), and with Morgan's method (MORGAN) and with the new deflation method (DEFLATION). For a problem of moderate size (Problem 1) in which there is a small cluster of small eigenvalues between 0.01 and 0.1 and then a uniform distribution of larger eigenvalues it can be seen that DEFLATION is superior to all the other methods with an improvement in efficiency of about a factor of 2 over the original implementation described in Erhel et al. (1996). These results are placed in context with Full GMRES and with restarted GMRES(20) which has approximately the same memory requirements as DEFLATION(16,4) in Figure 2.

In Figure 3 the conditioning of the problem is made considerably worse by changing the value of β from 0.9 to 1.1, although the eigenvalue distribution remains the same. Figure 3 shows the effect of this ill-conditioning in that restarted GMRES with a bigger restart now fails to converge. In fact DEFLATION(16,4) and MORGAN(16,4) also failed to converge and more eigenvalues (10) had to be deflated in order to obtain convergence. In all cases, including Full GMRES, convergence was initially very slow, with DEFLATION being

smoother in its convergence than MORGAN. This property of initial stalling was noticed for all methods when the system matrix was severely ill-conditioned.

Finally for Problem 1, Figure 4 shows the effects of having two subsystems in which there is a strong coupling between the two subsystems and for which the overall matrix is very ill-conditioned ($\beta = 1.2$). These results indicate that it is important to deflate at least all of the eigenvalues in the first block otherwise the ill-conditioning will cause the convergence to stall. These effects are emphasised in Figure 5, where the same eigenvalue distribution is maintained but β is increased to 1.3. Some very interesting effects are observed here in that even for Full GMRES it is not possible to gain a stable residue of less than approximately 10^{-5} . Another interesting effect is that as the restart is increased beyond 20, convergence either stalls or the iterations grow in an unbounded fashion.

The problem here, as with Morgan's method (in which only the same eigenvalues are refined after each restart) is that only a few eigenvalues are deflated at each restart and if the restart is too large but not large enough (in this case it has to be close to the value N) then by the time a small set of eigenvalues is deflated it is too late to be of any use, and if too large a set is deflated then they are too inaccurate. This difficulty could perhaps be ameliorated by allowing eigenvalues to be deflated at any time rather than just at a restart.

Problem 2

The second class of problems is one chosen from [7], with a slight modification. In this case A is given by

$$A = S(N, n_1, \beta) \begin{pmatrix} A_n & 0 \\ 0 & B_{N-n} \end{pmatrix} S(N, n_1, \beta)^{-1},$$

where A_n is a circulant matrix of dimension n with first row $(c, 0, \dots, 0, d)$ while B_n is a diagonal matrix of size n with uniformly distributed random numbers in $[a, b]$. The eigenvalues of A_n lie on a circle of centre c and radius d .

Convergence results for Full GMRES, restarted GMRES, Morgan and Deflation are given for Problem 2 in Figures 6 and 7. For this problem the smallest eigenvalues in magnitude are distributed evenly around a circle (in this case 30 eigenvalues in a circle of radius 2 and centre $-3/2$). For $N = 100$ and $\beta = 0.9$, Figure 6 shows that all methods stall at a residue of about 10^{-5} with Full GMRES only converging on the last iteration. Thus in this case restarted GMRES will prove completely inadequate while DEFLATION is again superior to MORGAN. Other numerical tests not presented here show that in this case deflating about half of the eigenvalues scattered around the circle gave close to optimal performance while deflating 4 eigenvalues at a time at each restart was superior to deflating 2 at a time.

Figure 7 shows that for the same eigenvalue distribution but with much greater ill-conditioning, MORGAN fails for almost all possible choices of parameters for the restart and the eigenvalue deflation, while DEFLATION is still robust. The difficulties associated with severe ill-conditioning are clearly illustrated in Figure 7 when comparing DEFLATION(25,12,2) with the β values of 0.9 and 1.1. Here the notation DEFLATION(25,12,2,1.1) means that deflation is used with a restart of 25, with 12 eigenvalues being deflated, at two at a time, on a problem with $\beta = 1.1$.

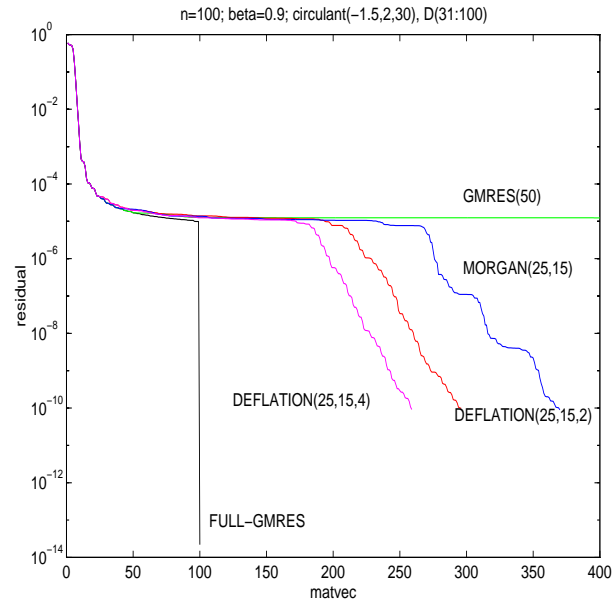


Figure 6: Convergence rates for Example 2

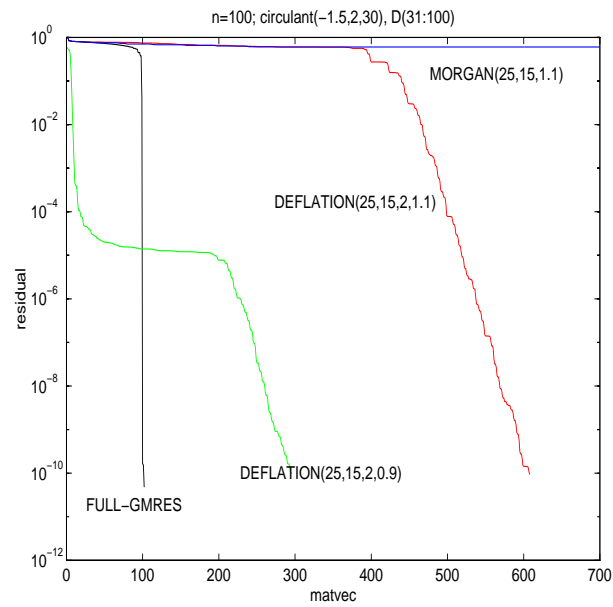


Figure 7: Convergence rates with severe ill-conditioning

Problem 3

Problem 3 is included to show the performance of the deflation approach on an important application coming from the numerical solution of partial differential equations using collocation techniques. In this case the size of the problem is 2048 and the system of equations is dense. An initial diagonal row scaling was applied as a preconditioner. The tests were performed on Morgan's method running on 1, 2 and 4 processors of an SGI Power Challenge at the University of Queensland and the times are given in seconds. In each case the residue in the 2-norm was about 10^{-4} .

performance	FULL GMRES	MORGAN(26,4)	GMRES(30)
iterations	57	77	1487
1 processor	17.33	21.73	45.111
2 processors	9.02	11.41	
4 processors)	6.06	7.33	

Table 1: Results comparing MORGAN, Full and Restarted GMRES

These results show a number of important points.

- In terms of time MORGAN(26,4) is at least twice as fast as GMRES(30) for the same memory requirements.
- Full GMRES is not much faster than MORGAN.
- There are good parallel efficiencies between 1 and 4 processors (approximately 75%) on a shared memory parallel computer. This is due to the fact that the problem is dense and reasonably large. For sparse problems the efficiencies would be considerably worse.

Problem 4

Finally in this section some negative results are reported for the surface fitting problem when λ is small. In this case the system matrix has a large number of eigenvalues clustered near zero and only a few relatively large eigenvalues. Thus the deflating of a small number of eigenvalues gives little advantage over restarted GMRES. However, in this GCV application a number of systems of equations have to be solved which are just updated by a constant diagonal term λ . By preserving the eigenvectors across the systems [9] and [11] have shown that deflation can work very impressively across systems of equations if special structures can be exploited (in this case the eigenvectors remain the same from system to system).

5 Conclusions

This paper has built on the previous work of [3] and [2] to develop an automatic preconditioning approach for restarted GMRES in which new eigenvalues are deflated and progressively

refined in a preconditioner which is updated at each restart. This modification leads to considerable improvements over both the augmented subspace method and the original deflating preconditioner. Furthermore, this approach seems to be more robust than either of these two approaches in the face of extreme ill-conditioning of the eigenvectors. Difficulties still remain in knowing how to choose apriori the size of the restart, the total number of eigenvalues to be deflated and how many are deflated at each restart. However, the numerical results presented here show that some general strategies can be applied if apriori information is known about the spectrum and the conditioning of the eigenvectors. In any case the deflation approach appears to be robust against possible poor choices of these parameters.

It is intended, however, to automate this choice further and to consider how this approach can be used in software where sequences of linear systems have to be solved from step to step (as in ordinary differential codes) in which much of the system information (eigenvectors, for example) is preserved from step to step.

References

- [1] H.A. Van der Vorst and C. Vuik. The superlinear convergence behaviour of gmres. *Journal of Computational and Applied Mathematics*, 48:327–341, 1993.
- [2] R. B. Morgan. A restarted GMRES method augmented with eigenvectors *SIAM J. matrix Anal. App.*, 16:1154–1171, 1995.
- [3] J. Erhel, K. Burrage, B. Pohl. Restarted GMRES preconditioned by deflation. *J. Comput. App. Math.*, 69 :303-318,1996.
- [4] D.C. Sorenson. Implicit application of polynomial filters in a k -step Arnoldi method. *SIAM J. Matrix Anal. Appl.*,13:357–385, 1992
- [5] R. Lehoucq Analysis and implementation of an implicitly restarted Arnoldi iteration. *Ph.D. Thesis, Rice University, Houston,USA*, 1995.
- [6] K. Burrage, A. Williams, J. Erhel and B. Pohl. The implementation of a Generalized Cross Validation algorithm using deflation techniques for linear systems. *J. Applied Num. Math.*, 19: 17–31, 1995.
- [7] J. Baglami, D. Calvetti, G.H. Golub and L. Reichel. Adaptively preconditioned GMRES algorithms. *Research report*
- [8] W. Sun. Orthogonal Collocation Solution of Biharmonic Equations. *Research report, Simon Fraser University, Canada*, 1996.
- [9] A. Williams Efficient iterative methods for large systems of linear equations. *Ph.D. Thesis, The University of Queensland, Brisbane, Australia*, 1996.

- [10] L. Lau, M. Rezný, J. Belward, K. Burrage and B. Pohl. ADVISE-Agricultural Development Visualization Interactive Software Environment. *Proceedings of CONPAR 1994, Lectures Notes in Computer Science, Springer-Verlag*, 854:184-195, 1994.
- [11] R.B. Sidje, A. Williams. Fast Generalized Cross Validation. *Research Report, University of Queensland, Brisbane, Australia*, 1996.
- [12] G. Wahba, J. Wendelberger. Some new mathematical methods for variational objective analysis using splines and cross validation. *Monthly Weather Rev.*, 108 :1122 :1145, 1980.
- [13] K. Burrage, J. Erhel and B. Pohl A deflation technique for linear systems of equations. *SISC*, To appear.
- [14] Lloyd N. Trefethen Noel N. Nachtigal, Lothar Reichel. A hybrid gmres algorithm for nonsymmetric linear systems. *SIAM J. Matrix Anal. Appl.*, 13(3):796–825, July 1992.
- [15] Y. Saad and M.H. Schultz. Gmres: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.
- [16] Youcef Saad. A flexible inner-outer preconditioned gmres algorithm. *SIAM J. Sci. Stat. Comput.*, 14(2):461–469, March 1993.
- [17] A. Chapman and Y. Saad. Deflated and augmented Krylov subspace techniques. *Research report*, University of Minnesota, UMSI 95/181, 1995.
- [18] Gautam M. Shroff and Herbert B. Keller. Stabilization of unstable procedures: the recursive projection method. *SIAM J. Numer. Anal.*, 30(4):1099–1120, August 1993.



Unité de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unité de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unité de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unité de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unité de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399