Science & Technology Facilities Council

Technical Report

Using FGMRES to obtain backward stability in mixed precision

Mario Arioli and Iain S. Duff

February 12, 2008

RAL-TR-2008-006

Library and Information Services
SFTC Rutherford Appleton Laboratory
Harwell Science and Innovation Campus
Didcot
OX11 0QX
UK
Tel: +44 (0)1235 445384
Fax: +44(0)1235 446403
Email: library@rl.ac.uk

ISSN 1358-6254

RAL-TR-2008-006

Using FGMRES to obtain backward stability in mixed precision[1]

Mario Arioli and Iain S. Duff

ABSTRACT

We consider the triangular factorization of matrices in single precision arithmetic and show how these factors can be used to obtain a backward stable solution. Our aim is to obtain double precision accuracy even when the system is ill-conditioned. We examine the use of iterative refinement and show by example that it may not converge. We then show both theoretically and practically that the use of FGMRES will give us the result that we desire with fairly mild conditions on the matrix and the direct factorization. We perform extensive experiments on dense matrices using MATLAB and indicate how our work extends to sparse matrix factorization and solution.

Keywords: FGMRES, backward stability, iterative refinement, Gaussian elimination, LAPACK, HSL, multifrontal method.

AMS(MOS) subject classifications: 65F05, 65F10, 65F50, 65G50.

---

[1]Current reports available by anonymous ftp to `ftp.numerical.rl.ac.uk` in directory pub/reports. This report is in file arduRAL20070XX.pdf. The report also available through the URL `www.numerical.rl.ac.uk/reports/reports.html`. This work was supported by the EPSRC Grants EP/E053351/1 and EP/F006535/1.

Computational Science and Engineering Department
Atlas Centre
Rutherford Appleton Laboratory
Oxon OX11 0QX

Contents

1 Introduction

We are concerned with the solution of

$$Ax = b, \tag{1.1}$$

when A is an $n \times n$ matrix and x and b are vectors of length n. For most of our discussion the matrix A is dense and unsymmetric although we will consider the case of sparse symmetric A in Section 6. We will solve these systems using a direct method where the matrix A is first factorized as

$$A \rightarrow LU,$$

where L and U are triangular matrices. The solution is then obtained through forward elimination

$$Ly = b$$

followed by back substitution

$$Ux = y,$$

where we have omitted permutations required for numerical stability and sparsity preservation for the sake of clarity. When A is symmetric we use an LDL^T factorization where the matrix D is block diagonal with blocks of order 1 and 2, so that we can stably factorize indefinite systems.

On many emerging computer architectures, the use of single precision arithmetic (by which we mean working with 32-bit floating-point numbers) is faster than using double precision. In fact on the Cell processor, single precision working can be more than ten times as fast as using double precision (Buttari, Dongarra, Langou, Langou, Luszczek and Kurzak 2007). In addition, half the storage is required when using single precision and the movement of data between memory hierarchies and cache and processing units is much reduced. However, in many applications, a higher accuracy is required than single precision (with a value of machine precision around 10^{-7}) or the matrix can be so ill-conditioned that single precision working is unable to obtain accuracy to even one significant figure .... that is the results are meaningless.

In this paper, we show how the selective use of double precision post-processing can enable solutions with a backward error (scaled residual) of double precision accuracy (around 10^{-16}) even when the factorization is computed in single precision. We show that the use of iterative refinement in double precision may fail when the matrix is ill-conditioned and then show that, even for such badly behaved matrices, the use of FGMRES (Saad 1993) can produce answers to the desired level of accuracy, namely that the solution process using FGMRES is backward stable at the level of double precision. We prove that, under realistic assumptions on the matrix and the factorization, the computation in mixed precision, where the LU factorization is computed in single precision and the FGMRES iteration in double precision, gives a backward stable algorithm.

We briefly discuss iterative refinement and FGMRES in Section 2 and prove the convergence of the mixed precision FGMRES algorithm in Section 3. We then describe

our construction of dense test matrices in Section 4 and illustrate the performance of our algorithms in Section 5. We then show how this can be extended to sparse matrices in Section 6 where we perform the single precision factorization using the HSL code `MA57` (Duff 2004). We conclude in Section 7 by illustrating how our ideas are important for high performance computing.

2 Iterative refinement and FGMRES

The standard technique for improving a solution to (1.1) is to use iterative refinement. This consists of computing the residual

$$r^{(k)} = b - Ax^{(k)} \tag{2.1}$$

to the current estimate of the solution $x^{(k)}$ and then calculating a change to this estimate by solving

$$A\delta x^{(k)} = r^{(k)} \tag{2.2}$$

and obtaining the new estimate as

$$x^{(k+1)} = x^{(k)} + \delta x^{(k)}.$$

The solution of the correction equation (2.2) will of course use the original factorization of A and so can be performed relatively quickly. It is easy to see that the condition for the convergence of iterative refinement is that the spectral radius of $I - MA$ is less than one where M is the approximation to A^{-1} obtained using the factorization of A.

Originally it was customary (Wilkinson 1965) to perform the computation of the residual in higher precision but more recently (Skeel 1980) established that in order to reduce the scaled residual (backward error) to machine precision, it was only necessary to compute the residual and correction in the same precision as the original computation. However, since we wish to obtain residuals with double precision accuracy when using a single precision factorization we will follow the original recommendation and compute the residuals in double precision.

One potentially major restriction on using iterative refinement is the condition on the spectral radius of $I - MA$. If M is not a very accurate factorization for A then this condition may not be met.

We (Arioli, Duff, Gratton and Pralet 2007) have discussed the case at length when A is sparse and the factorization is computed using static pivoting, for example using the HSL code `MA57` (Duff 2004). There we have shown that in cases when iterative refinement fails, FGMRES (Saad 1993) will normally work and is far more robust than either iterative refinement or GMRES (Saad and Schultz 1986).

In this current work, we are also potentially computing an M which is far from A^{-1} because we compute it in single precision. This will be particularly the case when the condition number of the matrix is large, say around the inverse of single precision rounding

(10^7). We will this also study the use of FGMRES in this context both experimentally (Section 5) and theoretically (Section 3).

The FGMRES algorithm is an Arnoldi method based on Krylov sequences and we present it in detail as Algorithm 2.1. The main reason why Arioli et al. (2007) found that FGMRES was superior to GMRES was that the second set of vectors Z_k corresponding to the preconditioned problem are computed and held in addition to the normal orthonormal sequence v_k.

Algorithm 2.1

procedure [x] = FGMRES(A,M_i,b,maxit)
$x_0 = M_0^{-1}b$, $r_0 = b - Ax_0$ *and* $\beta = ||r_0||$
$v_1 = r_0/\beta$; $k = 0$; $r = r_0$
while $||r|| > \varepsilon\,(||b|| + ||A||\;||x_k||)$ *and* $k < maxit$
$k = k + 1$;
$z_k = M_k^{-1}v_k$; $w = Az_k$;
for $i = 1, \ldots, k$ *do*
$h_{i,k} = v_i^T w$;
$w = w - h_{i,k}v_i$;
end for;
$h_{k+1,k} = ||w||$;
$v_{k+1} = w/h_{k+1,k}$;
$Z_k = [z_1, \ldots, z_k]$; $V_k = [v_1, \ldots, v_k]$;
$H_k = \{h_{i,j}\}_{1 \le i \le j+1; 1 \le j \le k}$;
$y_k = \arg\min_y ||\beta e_1 - H_k y||$;
if $||\beta e_1 - H_k y_k|| \le \varepsilon\,(||b|| + ||A||\;||x_k||)$ *do*
$x_k = x_0 + Z_k y_k$ *and* $r = b - Ax_k$;
end if
end while;
end procedure.

3 Theoretical proof of convergence for FGMRES

The first part of our analysis is independent of the choice of the matrices M_i in FGMRES (Algorithm 2.1). The only thing that we assume is that the computed version of the Z_k matrix:

$$Z_k = [z_1, \ldots, z_k]$$

is of rank k for all values of $k \le n$. This will guarantee convergence of the algorithm. We will later show how mixed precision influences the rank and the convergence of

the algorithm. Furthermore, the roundoff error analysis of FGMRES in theorem 3.1 is independent of the specific choice of $\bar{z}_k$ at each step if the resulting $\bar{Z}_k$ is full rank.

Under this assumption, we decompose Algorithm 2.1 into three main subalgorithms;

- Computation of the matrices $C^{(k)}$, V_k, and R_k by the Modified Gram-Schmidt algorithm (MGS) such that

$$C^{(k)} = [r_0, AZ_k] = V_{k+1}R_k; \qquad V_j^T V_j = I_j \quad \forall j, \tag{3.1}$$

  where

$$R_k = \left[\begin{array}{cc} \beta e_1 & H_k \end{array} \right], \tag{3.2}$$

$$AZ_k = V_{k+1}H_k. \tag{3.3}$$

  and H_k is upper Hessenberg. Column $k+1$ of $C^{(k+1)}$ is computed after the k-th step of MGS in (3.1) and (3.2) computing or choosing a new $\bar{z}_{k+1}$. We then continue by generating the next column of V_{k+2} and R_{k+1}.

- Computation of the vector y_k by solving the least-squares problem

$$\min_y ||\beta e_1 - H_k y|| \tag{3.4}$$

  using a QR algorithm based on Givens rotations and the upper Hessenberg structure of H_k

- Computation of $x_k = x_0 + Z_k y_k$ when the residual $||\beta e_1 - H_k y_k||$ is less than or equal to the prescribed threshold.

In the following, we will denote by $c_p(n, j)$ functions that depend only on the dimension n and the integer j. If the second index is omitted then the function will depend only on n. We will avoid a precise formulation of these dependences, but we assume that each $c_p(n, j)$ grows moderately with n and j. Finally, if $B \in \mathbb{R}^{p\times q}, p \geq q$ is a full rank matrix, we denote by $\kappa(B) = ||B||\ ||B^+||$ its spectral condition number where $B^+ = (B^T B)^{-1} B$. For all matrices and vectors we denote by $|B|$ the matrix or vector of the absolute values. Furthermore, we will denote the computed quantities of R_k, V_k, H_k, y_k, r_k, and x_k by the same symbol with a bar above it, i.e. the matrix $\bar{H}_k$ will be the computed value of the matrix H_k.

Theorem 3.1 *If we apply Algorithm 2.1 to solve (1.1), using finite-precision arithmetic conforming to IEEE standard with relative precision ε and under the following hypotheses:*

$$2.12(n+1)\varepsilon < 0.01 \qquad \textit{and} \qquad c_0(n)\varepsilon\,\kappa(C^{(k)}) < 0.1\ \forall k \tag{3.5}$$

where

$$c_0(n) = 18.53 n^{\frac{3}{2}}$$

and

$$|\bar{s}_k| < 1 - \varepsilon\,,\ \forall k, \tag{3.6}$$

where $\bar{s}_k$ *are the sines computed during the Givens algorithm applied to* $\bar{H}_k$ *in order to compute* $\bar{y}_k$*, then there exists* $\hat{k}$*,* $\hat{k} \leq n$ *such that,* $\forall k \geq \hat{k}$*, we have*

$$\begin{aligned} ||b - A\bar{x}_k|| \ \leq\ & c_1(n,k)\varepsilon \Big(||b|| + ||A||\,||\bar{x}_0|| + \\ & ||A||\,||\,|\bar{Z}_k|\,|\bar{y}_k|\,|| + ||A\bar{Z}_k||\,||\bar{y}_k|| \Big) + \mathcal{O}(\varepsilon^{\,2}). \end{aligned} \tag{3.7}$$

Proof. The proof is based on the following two lemmata.
Note that we will use some componentwise bounds to obtain the factor $||\,|\bar{Z}_k|\,|\bar{y}_k|\,||$ in bound (3.7). In our earlier analysis we had replaced this with the majorizing quantity $||\bar{Z}_k||\,||\bar{y}_k||$ but found that this was too loose a bound and that the quantity $||\bar{Z}_k||\,||\bar{y}_k||$ could be very large in our numerical experiments.

Lemma 3.1 *If we apply MGS to factorize* $C^{(k)}$ *in (3.1), using finite-precision arithmetic conforming to IEEE standard with relative precision* ε *and under the hypotheses (3.5), then there exist orthonormal matrices* $\hat{V}_k$ *such that*

$$\bar{C}^{(k)} = [r_0 + f, A\bar{Z}_k + E_k] = \hat{V}_{k+1}\overline{R}_k \qquad \forall k \leq n. \tag{3.8}$$

with

$$\begin{aligned} &||f|| \leq c_2(n,1)\varepsilon\,(||b|| + ||A||\,||\bar{x}_0||) + \mathcal{O}(\varepsilon^{\,2}) \qquad \textit{and} \\ &|E_k| \leq c_3(n,k)\varepsilon\,(||A\bar{Z}_k||\mathbf{u}_n\mathbf{u}_k^T + |A|\,|\bar{Z}_k|) + \mathcal{O}(\varepsilon^{\,2}), \end{aligned} \tag{3.9}$$

where we denote by $\mathbf{u}_j$ *the vectors of order* j *with all entries equal to* 1*. Moreover, the computed value of* $\bar{V}_k$ *satisfies the relation*

$$||\bar{V}_k^+|| \leq 1.3. \tag{3.10}$$

Proof. By standard techniques (Higham 2002), the computed matrix by vector products satisfy the relations

$$\bar{r}_0 \ = \ r_0 + f_1 \qquad ||f_1|| \leq c_4(n,1)\varepsilon\,(||b|| + ||A||\,||\bar{x}_0||) + \mathcal{O}(\varepsilon^{\,2}), \tag{3.11}$$

$$fl(A\bar{Z}_k) \ = \ A\bar{Z}_k + F_k^{(1)}, \qquad |F_k^{(1)}| \leq c_5(n,k)\varepsilon\,|A|\,|\bar{Z}_k| + \mathcal{O}(\varepsilon^{\,2}). \tag{3.12}$$

Following Björck and Paige (1992) and Giraud and Langou (2002), the Gram-Schmidt orthogonalization process applied to $fl(C^{(k)})$ computes an upper triangular matrix $\overline{R}_k$ for which there exists an orthonormal matrix $\hat{V}_{k+1}$ that satisfies the relations:

$$\begin{cases} [\bar{r}_0; fl(A\bar{Z}_k)] + [f_2; F_k^{(2)}] = \hat{V}_{k+1}\overline{R}_k, \qquad \hat{V}_{k+1}^T\hat{V}_{k+1} = I_{k+1} \\ ||f_2|| \leq c_6(n,1)\varepsilon\,||r_0|| + \mathcal{O}(\varepsilon^{\,2}) \qquad ||F_k^{(2)}|| \leq c_7(n,k)\varepsilon\,||A\bar{Z}_k|| + \mathcal{O}(\varepsilon^{\,2}) \end{cases} \tag{3.13}$$

under the hypothesis (3.5).
By combining (3.11), (3.12), and (3.13), we have

$$\begin{cases} [r_0; A\bar{Z}_k] + [f_1 + f_2; F_k^{(1)} + F_k^{(2)}] = \hat{V}_{k+1}\overline{R}_k, \\ ||f_1 + f_2|| = ||f|| \leq c_2(n,1)\varepsilon\,(||b|| + ||A||\,||\bar{x}_0||) + \mathcal{O}(\varepsilon^{\,2}) \quad \text{and} \\ |F_k^{(1)} + F_k^{(2)}| = |E_k| \leq c_3(n,k)\varepsilon\,(||A\bar{Z}_k||\mathbf{u}_n\mathbf{u}_k^T + |A|\,|\bar{Z}_k|) + \mathcal{O}(\varepsilon^{\,2}). \end{cases} \tag{3.14}$$

Lemma 3.2 *Applying the QR factorization with Givens rotations to solve (3.4), using finite-precision arithmetic conforming to IEEE standard with relative precision ε and under the condition*

$$0.1 > c_0(n)\varepsilon\,\kappa(\bar{H}_k) + \mathcal{O}(\varepsilon^{\,2}) \quad \forall k, \tag{3.15}$$

there exist an orthonormal matrix $\hat{G}^{[k]}$, a vector $g^{[k]}$, and an upper Hessenberg matrix δH such that the computed value $\bar{y}_k$ satisfies the following relations

$$\begin{cases} \bar{y}_k = \arg\min_y ||\hat{G}^{[k]}(\bar{\beta}e_1 + g^{[k]} - (\bar{H}_k + \Delta H_k)y)||, \\ ||\Delta H_k|| \le c_8(k,1)\varepsilon\,||\bar{H}_k|| + \mathcal{O}(\varepsilon^{\,2}) \text{ and } ||g^{[k]}|| \le c_9(k,1)\varepsilon\,\bar{\beta} + \mathcal{O}(\varepsilon^{\,2}). \end{cases} \tag{3.16}$$

Moreover, the residuals

$$\alpha_k = ||\hat{G}^{[k]}(\bar{\beta}e_1 + g^{[k]} - (\bar{H}_k + \Delta H_k)\bar{y}_k)||,$$

satisfy the equations

$$\begin{cases} \alpha_k = \bar{\beta}\Big(\prod_{j=0}^{k} |\bar{s}_j|\Big)\Big(\prod_{j=0}^{k}(1+\zeta_j)\Big) \\ |\zeta_j| \le \varepsilon \quad \forall\, j. \end{cases} \tag{3.17}$$

Under Hypothesis 3.6, we have that α_k is strictly decreasing to zero and $\alpha_{\hat{k}} = 0$ for some value of $\hat{k} \le n$.

Proof. See (Arioli 2008).

We point out that Hypothesis 3.5 implies Hypothesis 3.15.

From the orthogonality of

$$\widetilde{V}_{k+1} = \hat{V}_{k+1}\hat{G}^{[k]T},$$

we have (using (3.2) and (3.8)) that

$$\begin{aligned} \alpha_k &= ||\widetilde{V}_{k+1}\hat{G}^{[k]}(\bar{\beta}e_1 + g^{[k]} - (\bar{H}_k + \Delta\bar{H}_k)\bar{y}_k)|| \\ &= ||r_0 + f + \hat{V}_{k+1}g^{[k]} - A(\bar{Z}_k + A^{-1}(E_k + \hat{V}_{k+1}\Delta\bar{H}_k))\bar{y}_k|| \\ &= ||r_0 + \delta r_0 - A(\bar{Z}_k + \hat{Z}_k)\bar{y}_k||, \end{aligned} \tag{3.18}$$

where

$$\delta r_0 = f + \hat{V}_{k+1}g^{[k]} \tag{3.19}$$

$$\hat{Z}_k = A^{-1}(E_k + \hat{V}_k\Delta\bar{H}_k) \tag{3.20}$$

Thus, under the hypotheses (3.5) and (3.6) we have that

$$\bar{H}_k + \Delta\bar{H}_k$$

is full rank for all k and therefore the matrices $(\bar{Z}_k + \hat{Z}_k)$ have full rank for all k.

From (3.17), we have that

$$\alpha_k = \alpha_{k-1}|\bar{s}_k|(1+\zeta_k), \quad |\zeta_k| \le \varepsilon\,. \tag{3.21}$$

Then the values of α_k converge monotonically to zero for a finite value of $k = \hat{k}$. In the worst case this will happen for $\hat{k} = n$.

The last part of FGMRES is the computation of $\bar{x}_k$. The value $\bar{x}_k$ satisfies the relations

$$\begin{cases} \bar{x}_k = \bar{x}_0 + \bar{Z}_k \bar{y}_k + \delta x_k, \\ ||\delta x_k|| \leq c_{10}(k,1)\varepsilon \, || \, |\bar{Z}_k| \, |\bar{y}_k| \, || + \varepsilon \, ||\bar{x}_0|| + \mathcal{O}(\varepsilon^{\,2}). \end{cases} \tag{3.22}$$

Therefore we have from (3.18)

$$\alpha_k = ||r_0 + \delta r_0 + A\delta x_k - A\delta x_k - A\bar{Z}_k \bar{y}_k - A\hat{Z}_k \bar{y}_k||. \tag{3.23}$$

From (3.22), we have

$$\begin{cases} \alpha_k = ||b - A\bar{x}_k + w|| \geq ||b - A\bar{x}_k|| - ||w|| \\ w = \delta r_0 + A\delta x_k - A\hat{Z}_k \bar{y}_k \end{cases} \tag{3.24}$$

i.e.

$$||b - A\bar{x}_k|| \leq ||w|| + \alpha_k. \tag{3.25}$$

From (3.14), (3.16), and (3.22) we have

$$\begin{cases} ||\delta r_0|| \leq c_{11}(n,1)\varepsilon \, (||b|| + ||A|| \, ||\bar{x}_0||) + \mathcal{O}(\varepsilon^{\,2}) \\ ||A\delta x_k|| \leq c_{12}(n,1)\varepsilon \, ||A|| \left[|| \, |\bar{Z}_k| \, |\bar{y}_k| \, || + ||\bar{x}_0|| \right] + \mathcal{O}(\varepsilon^{\,2}) \end{cases} \tag{3.26}$$

and finally from (3.9), (3.12), (3.20), (3.24), and (3.26)

$$\begin{aligned} ||w|| \;\; \leq \;\; & c_{13}(n,k)\varepsilon \Big(||b|| + ||A|| \, ||\bar{x}_0|| + ||\bar{H}_k|| \, ||\bar{y}_k|| + \\ & ||A \, \bar{Z}_k|| \, ||\bar{y}_k|| + ||A|| \, || \, |\bar{Z}_k| \, |\bar{y}_k| \, || \Big) + \mathcal{O}(\varepsilon^{\,2}). \end{aligned} \tag{3.27}$$

Therefore under the hypothesis (3.5) and from (3.25), we have

$$\begin{aligned} ||b - A\bar{x}_k|| \;\; \leq \;\; & \alpha_k + c_1(n,k)\varepsilon \Big(||b|| + ||A|| \, ||\bar{x}_0|| + ||\bar{H}_k|| \, ||\bar{y}_k|| + \\ & ||A \, \bar{Z}_k|| \, ||\bar{y}_k|| + ||A|| \, || \, |\bar{Z}_k| \, |\bar{y}_k| \, || \Big) + \mathcal{O}(\varepsilon^{\,2}) \end{aligned} \tag{3.28}$$

and, then, taking into account that, from the first relation in (3.14) and (3.2),

$$||\bar{H}_k|| \leq ||A\bar{Z}_k|| + \mathcal{O}(\varepsilon \,) \; \forall k < \hat{k}, \tag{3.29}$$

we have (3.7).

3.1 Computing $\bar{Z}_k$ using single precision arithmetic

In this subsection, we choose $M_j = A$ for all j in Algorithm 2.1, and we justify the satisfactory convergence behaviour of FGMRES when $\bar{z}_j$ the j-th column of $\bar{Z}_k$ is computed by solving the system

$$Az_j = v_j, \tag{3.30}$$

using an LU factorization of A based on IEEE single precision arithmetic (we assume that the single precision unit roundoff is $\approx \sqrt{\varepsilon}$).

The computed solution $\bar{z}_j$ (Higham 2002) satisfies the relations

$$A\bar{z}_j = \bar{v}_j + w_j \qquad ||w_j|| \leq \sqrt{\varepsilon}\, c_{14}(n)||A||\, ||\bar{z}_j||\, \Gamma, \tag{3.31}$$

where, given the computed $\bar{L}$ and $\bar{U}$,

$$\Gamma = \frac{|| \, |\bar{L}| \, |\bar{U}| \, ||}{||A||}.$$

Thus, we have the following relation

$$\begin{array}{l} A\bar{Z}_k = \bar{V}_k + W_k \\ W_k = [w_1, \ldots, w_k] \\ ||W_k|| \leq \sqrt{\varepsilon}\, c_{15}(n)\sqrt{k}\, \Gamma\, ||A||\, ||\bar{Z}_k|| \end{array} \tag{3.32}$$

Multipling the first equation in (3.22) by A, we have

$$A(\bar{x}_k - \bar{x}_0 - \delta x_k) = A\bar{Z}_k\bar{y}_k. \tag{3.33}$$

and then from (3.32) it follows that

$$A(\bar{x}_k - \bar{x}_0 - \delta x_k) = \bar{V}_k\bar{y}_k + W_k\bar{y}_k. \tag{3.34}$$

Under the hypotheses (3.5) of Lemma 3.1, $\bar{V}_k^T\bar{V}_k$ is invertible and thus we obtain

$$\bar{V}_k^+\Big[A(\bar{x}_k - \bar{x}_0 - \delta x_k) - W_k\bar{y}_k\Big] = \bar{y}_k. \tag{3.35}$$

Finally, combining (3.35) with (3.10), (3.22), and (3.32), we have

$$\begin{array}{rl} ||\bar{y}_k|| \leq & 1.3\Big[||A(\bar{x}_k - \bar{x}_0)|| + \\ & ||A||\Big(c_{10}(k,1)\varepsilon\, ||\bar{Z}_k||\, ||\bar{y}_k|| + \varepsilon\, ||\bar{x}_0||\Big) + \\ & \sqrt{\varepsilon}\, c_{15}(n)\sqrt{k}\, \Gamma\, ||A||\, ||\bar{Z}_k||\, ||\bar{y}_k||\Big] + \mathcal{O}(\varepsilon^{\,2}) \leq \\ & 1.3\Big[||A(\bar{x}_k - \bar{x}_0)|| + \varepsilon\, ||A||\, ||\bar{x}_0|| + \\ & c_{16}(n)\, \sqrt{\varepsilon}\, \Gamma\, ||A||\, ||\bar{Z}_k||\, ||\bar{y}_k||\Big] + \mathcal{O}(\varepsilon^{\,2}) \end{array} \tag{3.36}$$

Taking into account that $\bar{x}_0$ is computed in single precision, we have

$$||A\bar{x}_0 - b|| \leq \sqrt{\varepsilon}\, c_{15}(n)\, ||A||\, ||\bar{x}_0||\, \Gamma\,.$$

Then we can further simplify the right-hand side of (3.36) so that we have

$$\begin{array}{rl} ||\bar{y}_k|| \leq & 1.3\Big[||A\bar{x}_k - b|| + (\sqrt{\varepsilon} + \varepsilon\,)\, c_{15}(n)\, \Gamma\, ||A||\, ||\bar{x}_0|| + \\ & c_{16}(n)\, \sqrt{\varepsilon}\, \Gamma\, ||A||\, ||\bar{Z}_k||\, ||\bar{y}_k||\Big] + \mathcal{O}(\varepsilon^{\,2}) \end{array} \tag{3.37}$$

Moreover, if

$$\rho = c_{16}(n)\,\sqrt{\varepsilon}\,\Gamma\,||A||\,||\bar{Z}_k|| < 1, \tag{3.38}$$

then we have

$$||\bar{y}_k|| \leq \frac{1.3}{1-\rho}\Big[||A\bar{x}_k - b|| + (\sqrt{\varepsilon} + \varepsilon)\; c_{15}(n)\,\Gamma\,||A||\,||\bar{x}_0||\Big] + \mathcal{O}(\varepsilon^{\,2}). \tag{3.39}$$

Substituting the upper bound of (3.39) for $||\bar{y}_k||$ in (3.7) and assuming that

$$\chi = \frac{1.3\,c_1(n,k)}{1-\rho}\,\varepsilon\;\Gamma\,||A||\,||\bar{Z}_k|| < 1, \tag{3.40}$$

we have the final bound

$$||b - A\bar{x}_k|| \leq \frac{\varepsilon\;c_{17}(n,k)}{1-\chi}\Big(||b|| + \Gamma\,||A||\,||\bar{x}_0||\Big) + \mathcal{O}(\varepsilon^{\,2}). \tag{3.41}$$

Therefore, if Γ is not too big then we have normwise backward stability.

4 Construction of test matrices

We use a standard technique to generate test matrices with specified condition number and eigenvalue distribution. We do this by generating a diagonal matrix with the required properties and then pre and post multiplying it by random orthogonal matrices to obtain our test matrices. Thus, if we choose the matrix D as $\mathrm{diag}\{d_i\}$ where

$$d_i = 10^{-c\left(\frac{i-1}{n-1}\right)^{\gamma}} \tag{4.1}$$

then the singular values lie between 1 and 10^{-c}, the condition number is 10^c, and the distribution can be skewed by altering γ. γ equal to 1 gives a log-linear uniform distribution, values of γ greater than 1 skew towards 1 and values of γ less than 1 towards 10^{-c}.

We then use MATLAB in a standard fashion to generate random orthogonal matrices H and V and run our factorization and solution algorithms on the matrix

$$A \;=\; HDV. \tag{4.2}$$

5 Experimental results

In this section we report on our experiments on dense unsymmetric matrices generated as described in Section 4. We conduct these experiments using MATLAB. We perform the single precision factorization using SGETRF from LAPACK. More precisely, given the randomly generated matrix A (4.2), we use the MATLAB command

[L,U] = lu(single(A))

in order to generate the single precision factors L and U. As mentioned in Section 1, we will use this single-precision factorization as a preconditioner for Richardson's method (that is iterative refinement) or FGMRES.

We present two variants of the preconditioning:

1. the vector $\bar{z}_k$ is computed using the forward and backward substitution algorithm in single precision on the single precision conversion of vector $\bar{v}_k$,

2. the vector $\bar{z}_k$ is computed using the forward and backward substitution algorithm in double precision on $\bar{v}_k$ after we converted the factors L and U to double precision.

Note that all other computations are in double precision. The second case has the disadvantage of using more memory but makes the algorithm more robust. Moreover, even if the number of restarts increases, the total number of iterations decreases significantly in some examples. Note that our problems are essentially singular in single precision (we take c in equation (4.1) to be 8.2 or 8.0) so we should not be surprised if sometimes many iterations are required for full convergence to a scaled residual (backward error) at double-precision accuracy.

In all the tables, the first column reports the total number of iterations and the second the number of iterations after the last restart. These numbers are of course the same if no restarting is required. Our restart algorithm is automatic. We stop when the Arnoldi residual is at machine precision and restart only if the actual residual is not.

In Tables 5.1, 5.2, and 5.3 we show the numerical results for A of dimension 200. We point out that for values of γ in (4.1) less than 1 both variants of FGMRES (see Table 5.1) converge rapidly even for a condition number greater than 10^8. Note that the bounding quantities of equation (3.7) that are shown in columns 4,5 and 9,10 are very reasonable. For this case ($\gamma = 0.5$), the iterative refinement algorithm also converges usually in slightly more iterations. For $\gamma = 1$, however, iterative refinement either does not converge or converges very slowly. We see, in Table 5.2, that both FGMRES variants converge and, although the bounding quantities are larger than in Table 5.1, they are still reasonable, except for the fourth case. Here we need to restart to get convergence to full precision.

In the last case when $\gamma = 2$, the behaviour of all our algorithms deteriorates, both FGMRES variants restart after we detect a small residual for the least-squares internal problem but the computed residual $||b - A\bar{x}_k|| > \varepsilon$ ($\bar{x}_k$ the computed solution). However, both variants converge after few restarts (see Table 5.3). Again, on convergence, the bounding quantities are reasonable.

In Tables 5.4, 5.5, and 5.6, we present the results for A of dimension 400. The increased dimensionality of the matrix and the log-linear uniform distribution of the eigenvalues that causes a greater clustering near 10^{-c} exacerbates some of the behaviour observed for the lower dimensional case. Although the $\gamma = 0.5$ distribution still works well (also for iterative refinement), the algorithms require more iterations (and restarts) as γ increases although we eventually converge to double-precision machine precision.

Finally, in all our experiments the ratio between $||\bar{H}_k||$ and $||A\bar{Z}_k||$ is very close to 1.The mathematical relationship in equation (3.3) suggests that this ratio should be one and the relationship for computed values is given in equation (3.13) where the difference is of order $\varepsilon\,||A\bar{Z}_k||$. This result is a good vindication of this part of our theory.

Finally, when we compare our two variants of the preconditioning, we note that it is really quite advantageous to use double precision for all the computations (noting of course

that the matrix was always factorized in single precision). We would clearly recommend that this is done but we have included experiments where single precision is used since there might be architectures where the performance of these computations could be very sensitive to the precision used.

6 Extension to sparse systems

We cannot extend the experimental results to sparse systems totally within MATLAB since our version of MATLAB computes a sparse factorization only in double precision. We thus compute the factors separately in a Fortran program using a single precision version of `MA57` and then convert these to data structures that we can feed directly to MATLAB that performs the rest of the computation in double precision. Note that this means that we actually solve equation (2.2) using double precision arithmetic but our analysis is still valid.

We show the results of runs on some rather ill-conditioned sparse matrices in Table 6.1. In this case we use restarted FGMRES since the cost of keeping too many vectors can be high for these larger dimensioned systems. We note that although iterative refinement essentially converges on the first three examples, it is still not as accurate as FGMRES and requires many more iterations than FGMRES. On the last example, the convergence of iterative refinement was so slow that we stopped after 53 iterations. Note again that good convergence is associated with a small value for $|| \, |\bar{Z}_{\hat{k}}| \, |\bar{y}_{\hat{k}}| \, ||$ and it is noticeable how it often decreases markedly after a restart.

As we explained it is quite difficult to do these experiments using MATLAB, but we are looking into this further in the context of an HPC software grant to solve sparse equations efficiently in a multicore environment.

7 Conclusions

We have established both by theory and by experiment that solutions with a backward error at double-precision level can be obtained when using a single-precision factorization that is used as a preconditioner for FGMRES. We have also found that iterative refinement often does not work in such cases. This implies that we can take advantage of the faster speed of single-precision working on machines where speed or storage considerations give advantages for this mode of working. We have illustrated that this applies to sparse matrix factorizations as well as in the dense case.

Furthermore, all our analysis would be equally valid if an extended precision accuracy was required from a double precision factorization. In this case, the penalties of using extended precision, normally implemented in software, are very significant.

Acknowledgements

We would like to thank .....

References

Arioli, M., Duff, I. S., Gratton, S. and Pralet, S. (2007), 'A note on GMRES preconditioned by a perturbed LDL^T decomposition with static pivoting', *SIAM J. Scientific Computing* **29**(5), 2024–2044.

Arioli, M. (2008), 'Roundoff error analysis of orthogonal factorizations of upper Hessenberg rectangular matrices', RAL-TR-2008-004.

Björck, Å. and Paige, C. C. (1992), 'Loss and recapture of orthogonality in the modified Gram–Schmidt algorithm', *SIAM J. Matrix Anal. Appl.* **13**(1), 176–190.

Buttari, A., Dongarra, J., Langou, J., Langou, J., Luszczek, P. and Kurzak, J. (2007), 'Mixed precision iterative refinement techniques for the solution of dense linear systems', *Int. J. of High Performance Computing Applications* **21**(4), 457–466.

Duff, I. S. (2004), 'MA57 – A code for the solution of sparse symmetric indefinite systems', *ACM Trans. Math. Softw.* **30**(2), 118–144.

Giraud, L. and Langou, J. (2002), 'When modified Gram-Schmidt generates a well-conditioned set of vectors', *IMA J. Numer. Anal.* **22**, 521–528.

Higham, N. J. (2002), *Accuracy and Stability of Numerical Algorithms, Second Edition*, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA.

Paige, C., Rozložník, M. and Strakoš, Z. (2006), 'Modified Gram-Schmidt (MGS), least squares, and backward stability of MGS-GMRES', *SIAM Journal on Matrix Analysis and Applications* **28**(1), 264–284.

Saad, Y. (1993), 'A flexible inner-outer preconditioned GMRES algorithm', *SIAM J. Scientific and Statistical Computing* **14**, 461–469.

Saad, Y. and Schultz, M. H. (1986), 'GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems.', *SIAM J. Scientific and Statistical Computing* **7**, 856–869.

Skeel, R. D. (1980), 'Iterative refinement implies numerical stability for Gaussian elimination', *Mathematics of Computation* **35**(151), 817–832.

Wilkinson, J. H. (1965), *The Algebraic Eigenvalue Problem*, Oxford University Press, Oxford.

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
14	14	9.6e-17	1.5e+00	2.0e+01	14	14	8.3e-17	1.5e+00	2.0e+01
13	13	1.0e-16	1.5e+00	1.8e+01	13	13	1.0e-16	1.5e+00	1.8e+01
14	14	4.8e-17	1.7e+00	2.0e+01	12	12	1.5e-16	1.6e+00	1.9e+01
14	14	1.3e-16	1.5e+00	2.2e+01	13	13	1.2e-16	1.5e+00	2.1e+01
14	14	9.0e-17	1.7e+00	1.9e+01	14	14	7.2e-17	1.5e+00	1.9e+01
14	14	9.7e-17	1.6e+00	2.2e+01	14	14	1.1e-16	1.5e+00	2.0e+01
14	14	6.7e-17	1.6e+00	2.0e+01	14	14	4.8e-17	1.6e+00	2.0e+01
13	13	7.3e-17	1.5e+00	1.9e+01	13	13	8.8e-17	1.5e+00	1.9e+01
13	13	5.7e-17	1.4e+00	1.8e+01	13	13	6.0e-17	1.4e+00	1.8e+01
13	13	1.1e-16	1.4e+00	1.9e+01	13	13	9.1e-17	1.3e+00	1.8e+01

Table 5.1: Random dense matrices. $n = 200$, $c = 8.2$, $\gamma = 0.5$

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
26	26	2.5e-16	7.4e+00	1.9e+02	20	20	1.7e-16	8.0e+00	7.3e+01
27	27	6.6e-16	4.2e+00	4.7e+02	20	20	2.0e-16	3.9e+00	5.9e+01
25	25	1.7e-16	3.3e+00	5.9e+01	20	20	2.7e-16	3.5e+00	4.0e+01
52	52	3.9e-15	4.6e+01	3.0e+03	20	20	1.1e-15	4.5e+01	4.7e+02
88	36	1.1e-16	4.6e+01	6.0e-04	25	5	1.5e-16	4.8e+01	1.5e-05
24	24	1.3e-16	2.0e+00	3.8e+01	20	20	2.6e-16	2.2e+00	2.8e+01
31	31	2.5e-16	8.8e+00	1.7e+02	20	20	1.9e-16	1.1e+01	8.4e+01
24	24	2.0e-16	3.5e+00	1.2e+02	20	20	2.0e-16	3.9e+00	6.9e+01
24	24	1.8e-16	2.7e+00	8.8e+01	20	20	6.2e-16	3.0e+00	5.8e+01
26	26	2.7e-16	3.2e+00	1.5e+02	20	20	3.2e-16	3.5e+00	3.6e+01
44	44	5.7e-16	1.9e+01	5.9e+02	20	20	4.0e-16	2.0e+01	1.6e+02

Table 5.2: Random dense matrices. $n = 200$, $c = 8.2$, $\gamma = 1$

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
200	200	3.6e-09	2.2e+01	3.5e+09	20	20	9.1e-11	6.1e+00	3.1e+02
247	47	2.0e-16	2.3e+01	3.4e+01	40	20	1.8e-15	6.4e+00	1.5e-01
					56	16	2.1e-16	6.1e+00	5.7e-06
200	200	2.0e-09	4.9e+01	2.1e+09	20	20	2.5e-11	1.4e+01	7.9e+02
256	56	2.0e-16	4.9e+01	8.5e+00	40	20	9.5e-16	1.4e+01	9.5e-02
131	131	8.7e-13	8.8e+02	9.4e+05	20	20	5.7e-11	8.2e+02	1.0e+05
253	122	2.0e-16	9.1e+02	9.4e-01	40	20	2.3e-16	7.2e+02	2.1e+01
58	58	7.8e-15	1.1e+01	3.6e+03	20	20	4.7e-12	1.4e+01	1.9e+02
89	31	1.8e-16	1.2e+01	3.2e-05	39	19	2.0e-16	1.2e+01	1.8e-02
108	108	4.2e-14	1.2e+02	3.8e+04	20	20	2.0e-11	1.2e+02	1.3e+03
195	87	1.7e-16	1.1e+02	9.2e-02	40	20	2.2e-16	1.2e+02	3.9e-01
200	200	3.6e-09	2.3e+02	4.3e+09	20	20	4.2e-10	7.0e+01	1.4e+04
299	99	2.0e-16	2.3e+02	7.9e+01	40	20	8.0e-15	7.2e+01	1.6e+01
					56	16	2.2e-16	6.7e+01	1.3e-04
200	200	4.9e-10	3.3e+03	6.9e+08	20	20	1.1e-12	8.8e+02	1.3e+04
338	138	7.2e-16	3.2e+03	6.5e+02	38	18	1.9e-16	9.1e+02	1.7e-01
78	78	1.4e-14	2.8e+01	1.0e+04	20	20	6.1e-12	3.1e+01	1.1e+03
128	50	2.0e-16	3.0e+01	6.5e-04	40	20	2.0e-16	2.9e+01	5.3e-02
79	79	2.7e-15	1.9e+01	2.3e+03	20	20	4.4e-12	1.9e+01	5.5e+02
117	38	2.0e-16	2.0e+01	9.3e-05	39	19	2.0e-16	2.1e+01	5.4e-02
48	48	1.4e-15	7.6e+00	6.9e+02	20	20	5.5e-13	9.6e+00	1.7e+02
75	27	2.0e-16	7.9e+00	5.3e-06	36	16	2.1e-16	1.0e+01	2.6e-03

Table 5.3: Random dense matrices. $n = 200$, $c = 8.2$, $\gamma = 2$

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
13	13	6.3e-17	1.4e+00	2.5e+01	13	13	5.1e-17	1.3e+00	2.5e+01
14	14	6.9e-17	1.4e+00	2.8e+01	14	14	5.3e-17	1.4e+00	2.6e+01
13	13	1.1e-16	1.4e+00	2.7e+01	13	13	8.7e-17	1.3e+00	2.5e+01
13	13	1.1e-16	1.4e+00	2.6e+01	13	13	8.6e-17	1.4e+00	2.5e+01
13	13	8.6e-17	1.4e+00	2.7e+01	13	13	7.9e-17	1.3e+00	2.6e+01
13	13	7.9e-17	1.4e+00	2.8e+01	13	13	5.5e-17	1.3e+00	2.7e+01
13	13	1.1e-16	1.4e+00	2.7e+01	13	13	7.8e-17	1.3e+00	2.6e+01
14	14	4.0e-17	1.6e+00	2.7e+01	14	14	5.9e-17	1.4e+00	2.6e+01
14	14	8.1e-17	1.4e+00	2.7e+01	13	13	9.7e-17	1.4e+00	2.6e+01
14	14	7.5e-17	1.4e+00	2.7e+01	13	13	1.1e-16	1.3e+00	2.6e+01

Table 5.4: Random dense matrices. $n = 400$, $c = 8.0$, $\gamma = 0.5$

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
77	77	7.1e-15	1.4e+01	1.1e+04	20	20	4.1e-11	1.2e+01	2.6e+02
129	52	1.5e-16	1.4e+01	3.1e-04	40	20	2.3e-14	1.1e+01	3.5e-01
					57	17	1.7e-16	1.1e+01	1.7e-04
72	72	9.8e-16	1.9e+01	1.4e+03	20	20	1.5e-11	1.9e+01	2.7e+02
					40	20	1.3e-15	1.9e+01	1.5e-01
					49	9	1.4e-16	1.9e+01	9.1e-06
56	56	1.3e-15	8.6e+00	1.7e+03	20	20	4.0e-12	8.9e+00	1.8e+02
84	28	1.6e-16	8.4e+00	1.4e-05	40	20	3.5e-16	8.5e+00	2.2e-02
97	97	7.3e-14	4.1e+01	1.5e+05	20	20	4.4e-11	3.6e+01	7.5e+02
166	69	1.2e-16	4.2e+01	7.1e-03	40	20	2.2e-14	3.6e+01	7.5e-01
					57	17	1.4e-16	3.3e+01	2.2e-04
60	60	4.9e-16	7.4e+00	6.5e+02	20	20	2.1e-11	7.0e+00	1.4e+02
					40	20	5.6e-15	8.0e+00	7.2e-02
					55	15	2.0e-16	8.3e+00	2.5e-05
56	56	7.7e-15	7.6e+00	8.2e+03	20	20	1.7e-12	6.4e+00	1.0e+02
80	24	1.7e-16	7.5e+00	2.3e-05	40	20	1.7e-16	6.8e+00	5.5e-03
50	50	5.3e-16	6.6e+00	6.6e+02	20	20	4.6e-12	7.1e+00	1.3e+02
					40	20	2.4e-16	7.4e+00	2.7e-02
91	91	7.0e-14	3.8e+01	1.2e+05	20	20	6.6e-11	3.6e+01	1.3e+03
158	67	1.4e-16	3.9e+01	3.7e-03	40	20	4.2e-14	3.6e+01	1.3e+00
					58	18	1.7e-16	3.4e+01	6.2e-04
89	89	1.6e-15	3.8e+01	2.6e+03	20	20	1.2e-11	3.7e+01	5.8e+02
156	67	1.4e-16	3.7e+01	1.2e-03	40	20	7.8e-15	3.7e+01	1.8e-01
					55	15	1.5e-16	4.2e+01	1.3e-04
67	67	7.0e-15	9.6e+00	8.8e+03	20	20	2.8e-10	8.8e+00	1.5e+03
106	39	1.4e-16	8.5e+00	1.3e-04	40	20	6.4e-14	8.5e+00	1.1e+00
					58	18	1.4e-16	7.7e+00	2.7e-04

Table 5.5: Random dense matrices. $n = 400$, $c = 8.0$, $\gamma = 1$

Single Precision					Double Precision				
Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
96	96	6.8e-15	1.5e+01	5.3e+03	20	20	9.9e-09	1.8e+01	6.5e+02
165	69	2.5e-16	1.9e+01	3.0e-04	40	20	5.4e-10	1.7e+01	9.0e+01
					60	20	3.3e-11	1.6e+01	2.3e+00
					80	20	1.2e-12	1.5e+01	1.6e-01
					100	20	6.2e-14	1.6e+01	9.6e-03
					120	20	4.0e-15	1.6e+01	3.8e-04
					133	13	7.1e-16	1.6e+01	2.6e-05
93	93	3.9e-15	8.5e+00	2.6e+03	20	20	1.6e-08	7.4e+00	4.9e+02
148	55	2.9e-16	1.2e+01	5.5e-05	40	20	1.6e-09	7.6e+00	4.6e+01
					60	20	2.8e-10	7.8e+00	9.3e+00
					80	20	3.9e-11	7.7e+00	1.4e+00
					100	20	4.7e-12	7.5e+00	1.4e-01
					120	20	9.3e-13	7.6e+00	2.4e-02
					140	20	1.1e-13	7.4e+00	3.2e-03
					160	20	1.1e-14	7.7e+00	6.3e-04
					180	20	1.7e-15	7.3e+00	3.4e-05
					196	16	6.5e-16	7.6e+00	6.6e-06
204	204	2.5e-14	2.1e+02	2.8e+04	20	20	4.8e-09	1.4e+02	3.1e+03
368	164	2.8e-16	2.5e+02	5.5e-02	40	20	1.8e-10	1.3e+02	2.1e+02
					60	20	8.0e-12	1.3e+02	5.1e+00
					80	20	3.3e-13	1.5e+02	3.5e-01
					100	20	1.9e-14	1.5e+02	1.3e-02
					119	19	8.5e-16	1.4e+02	1.0e-03

Table 5.6: Random dense matrices. $n = 400$, $c = 8.0$, $\gamma = 2.0$

Matrix Id	n	Iterative refinement		FGMRES				
		Total It	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	Total It	Inner it	$\frac{\|\|b - A\bar{x}_{\hat{k}}\|\|}{(\|\|A\|\| \, \|\|\bar{x}_{\hat{k}}\|\| + \|\|b\|\|)}$	$\|\|A\bar{Z}_{\hat{k}}\|\|$	$\|\| \, \|\bar{Z}_{\hat{k}}\| \, \|\bar{y}_{\hat{k}}\| \, \|\|$
bcsstk20	485	30	2.1e-15	2	2	1.4e-11	1.7e+00	4.6e+02
($\kappa(A) \approx 5 \times 10^9$)				4	2	3.4e-14	1.6e+00	3.8e-01
				6	2	7.2e-17	1.6e+00	5.6e-04
bcsstm27	1224	22	1.6e-15	2	2	5.8e-11	1.7e+00	2.7e+01
($\kappa(A) \approx 5 \times 10^9$)				4	2	1.8e-11	6.3e-01	1.3e+00
				6	2	6.0e-13	2.0e+00	7.6e-02
				8	2	1.5e-13	1.7e+00	1.0e-02
				10	2	1.2e-14	1.7e+00	1.9e-03
				12	2	2.6e-15	1.8e+00	1.7e-04
				14	2	1.8e-16	1.6e+00	4.3e-05
s3rmq4m1	5489	16	2.2e-15	2	2	3.5e-11	1.0e+00	8.6e+01
($\kappa(A) \approx 4 \times 10^9$)				4	2	2.1e-13	1.1e+00	3.2e-01
				6	2	4.5e-15	1.7e+00	6.4e-03
				8	2	1.1e-16	1.6e+00	1.3e-04
s3dkq4m2	90449	53	1.1e-10	10	10	6.3e-17	1.2e+00	1.2e+03
($\kappa(A) \approx 7 \times 10^{10}$)								

Table 6.1: Sparse matrices results.