# ESTIMATES OF THE $L_2$ NORM OF THE ERROR IN THE CONJUGATE GRADIENT ALGORITHM

GÉRARD MEURANT

**Abstract.** In this paper we derive a formula relating the norm of the $l_2$ error to the $A$–norm of the error in the conjugate gradient algorithm. Approximating the different terms in this formula, we obtain an estimate of the $l_2$ norm during the conjugate gradient iterations. Numerical experiments are given for several matrices.

**1. Introduction.** In this paper we derive a formula relating the $l_2$ norm of the error to the $A$–norm of the error in the conjugate gradient algorithm for solving linear systems with a symmetric positive definite matrix. The problem of computing estimates for the $A$–norm of the error was considered in [5], [6], [7], [8], [9]. This is summarized in [10]. The computation of estimates in finite precision arithmetic was studied in [11].

Let $A$ be a large and sparse symmetric positive definite matrix of order $n$ and suppose we have an approximate solution $\tilde{x}$ of the linear system

$$Ax = g,$$

where $g$ is a given right hand side vector. The residual $r$ is defined as $r = g - A\tilde{x}$. The error $e$ being $e = x - \tilde{x}$, we obviously have, $e = A^{-1}r$. Therefore, if we consider the $A$–norm of the error,

$$\|e\|_A^2 = (e, Ae) = e^T Ae = r^T A^{-1} A A^{-1} r = r^T A^{-1} r.$$

Here we are interested to use the $l_2$–norm, for which

$$\|e\|^2 = r^T A^{-2} r.$$

To solve the linear system we use the conjugate gradient (CG) algorithm : let $x^0$ be given, $r^0 = g - Ax^0$, $p^0 = r^0$, for $k = 1, \ldots$ until convergence

$$\gamma_{k-1} = \frac{{r^{k-1}}^T r^{k-1}}{{p^{k-1}}^T A p^{k-1}},$$

$$x^k = x^{k-1} + \gamma_{k-1} p^{k-1},$$

$$r^k = r^{k-1} - \gamma_{k-1} A p^{k-1},$$

$$\beta_k = \frac{{r^k}^T r^k}{{r^{k-1}}^T r^{k-1}},$$

$$p^k = r^k + \beta_k p^{k-1}.$$

We would like to cheaply estimate the $l_2$ norm of the error, eventually some iterations before the current one.

The contents of the paper are as follows. In section 2 we derive a formula relating the $l_2$ norm to the $A$–norm of the error. Section 3 shows how to use this formula to compute estimates of the $l_2$ norm by introducing a delay. Section 4 gives some numerical experiments. In Section 5 we comment on what can be done when introducing a preconditioner to speed up convergence. The last section gives some conclusions.

**2. A formula for the norm of the error.** Formulas were given in [5], [6], [7], [8], [9] to compute bounds or estimates for the $A$–norm of the error for the conjugate gradient (CG) method. It is well known that CG is closely related to the Lanczos algorithm. These computations used the formula

$$(A\epsilon^k, \epsilon^k) = (r^0, A^{-1}r^0) - \|r^0\|^2 (T_k^{-1}e^1, e^1)$$

where $T_k$ is the matrix of the Lanczos algorithm coefficients and $e^j$ is the $j$th column of the identity matrix. The relation for the matrix $V_k$ of the Lanczos vectors is the following:

$$AV_k = V_k T_k + \eta_{k+1} v^{k+1} (e^k)^T,$$

$T_k$ is a tridiagonal matrix denoted as

$$T_k = \begin{pmatrix} \alpha_1 & \eta_2 & & & \\ \eta_2 & \alpha_2 & \eta_3 & & \\ & \ddots & \ddots & \ddots & \\ & & \eta_{k-1} & \alpha_{k-1} & \eta_k \\ & & & \eta_k & \alpha_k \end{pmatrix}.$$

This can also be written as

$$AV_k = V_{k+1} \tilde{T}_k,$$

with

$$\tilde{T}_k = \begin{pmatrix} T_k \\ \eta_{k+1}(e^k)^T \end{pmatrix}.$$

We also have

$$V_k^T A V_k = T_k.$$

The entries of $T_k$ are obtained from the CG coefficients by

$$\alpha_k = \frac{1}{\gamma_{k-1}} + \frac{\beta_{k-1}}{\gamma_{k-2}}, \quad \beta_0 = 0, \quad \gamma_{-1} = 1,$$

$$\eta_{k+1} = \frac{\sqrt{\beta_k}}{\gamma_{k-1}}.$$

Estimates of the $l_2$ norm were considered in [8] using techniques developed in [2], but this was needing lower and upper bounds of the smallest and largest eigenvalues of $A$ which cannot be easily available for some problems. In CG the iterates are (implicitly) given by

$$x^k = x^0 + V_k u^k.$$

Enforcing the orthogonality constraint $V_k^T r^k = 0$ we find that $u^k$ is the solution of

$$T_k u^k = \|r^0\| e^1.$$

where $e^j$ is the $j$th column of the identity matrix. The CG relations are obtained from Lanczos by considering the Cholesky decomposition of the tridiagonal matrix $T_k$ (which is positive definite) and it turns out that the Lanczos basis vectors are related to the CG residuals $r^k = b - Ax^k$ by

$$v^{k+1} = (-1^k)\frac{r^k}{\|r^k\|}.$$

Computing the $l_2$ norm of the error we have

$$\|\epsilon^k\|^2 = (b - Ax^k, A^{-2}(b - Ax^k)) = (b, A^{-2}b) - 2(b, A^{-1}x^k) + (x^k, x^k).$$

But,

$$(b, A^{-1}x^k) = (b, A^{-1}x^0) + (b, A^{-1}V_k u^k),$$

$$(x^k, x^k) = (x^0, x^0) + 2(x^0, V_k u^k) + (u^k, u^k).$$

The last term is obtained because of the orthonormality of the basis vectors. Putting all this together, we obtain

$$\|\epsilon^k\|^2 = (r^0, A^{-2}r^0) - 2(A^{-1}b - x^0, V_k u^k) + (u^k, u^k).$$

We are able to compute upper and lower bounds or at least good estimates of the first term on the right hand side using Gaussian quadrature. So, it remains to see what we can do with the two other terms. Let us consider the first one

$$(A^{-1}b - x^0, V_k u^k) = (r^0, A^{-1}V_k u^k).$$

We have

$$V_k T_k^{-1} = A^{-1}V_k + \eta_{k+1}A^{-1}v^{k+1}(e^k)^T T_k^{-1},$$

Then,

$$(r^0, A^{-1}V_k u^k) = (r^0, V_k T_k^{-1}u^k) - \eta_{k+1}(r^0, A^{-1}v^{k+1}(e^k)^T T_k^{-1}u^k).$$

The first term is easy to evaluate since

$$(r^0, V_k T_k^{-1}u^k) = \|r^0\|(V_k^T r^0, T_k^{-2}e^1) = \|r^0\|^2(e^1, T_k^{-2}e^1).$$

For the second term, we remark that $(e^k)^T T_k^{-1}u^k$ is a scalar. Therefore

$$\eta_{k+1}(r^0, A^{-1}v^{k+1}(e^k)^T T_k^{-1}u^k) = \eta_{k+1}\|r^0\|[(e^k)^T T_k^{-2}e^1](r^0, A^{-1}v^{k+1}).$$

A we said before, the basis vectors are proportional to the residuals

$$(r^0, A^{-1}v^{k+1}) = \frac{(-1)^k}{\|r^k\|}(r^0, A^{-1}r^k).$$

But

$$(r^0, A^{-1}r^k) = (r^0, \epsilon^k) = (r^0, e^0) - (r^0, V_k u^k).$$

Therefore,

$$(r^0, A^{-1}r^k) = (r^0, A^{-1}r^0) - \|r^0\|^2(e^1, T_k^{-1}e^1) = \|\epsilon^k\|_A^2.$$

Finally

$$\eta_{k+1}(r^0, A^{-1}v^{k+1}(e^k)^T T_k^{-1}u^k) = (-1)^k \eta_{k+1}\frac{\|r^0\|}{\|r^k\|}(e^k, T_k^{-2}e^1)\|\epsilon^k\|_A^2.$$

To obtain the $l_2$ norm it remains to see that

$$(u^k, u^k) = \|r^0\|^2(e^1, T_k^{-2}e^1).$$

Grouping these results together we have the following result.
THEOREM 2.1.

$$\|\epsilon^k\|^2 = (r^0, A^{-2}r^0) - \|r^0\|^2(e^1, T_k^{-2}e^1) + (-1)^k 2\eta_{k+1}\frac{\|r^0\|}{\|r^k\|}(e^k, T_k^{-2}e^1)\|\epsilon^k\|_A^2.$$

□

The formula for the $l_2$ norm can be written in an alternate way since as we have seen before

$$r^k = -\eta_{k+1}\|r^0\|(e^k, T_k^{-1}e^1)v^{k+1}.$$

Therefore

$$\frac{(-1)^{k+1}\|r^k\|}{\eta_{k+1}} = \|r^0\|(e^k, T_k^{-1}e^1).$$

COROLLARY 2.2.

$$\|\epsilon^k\|^2 = \|r^0\|^2[(e^1, T_n^{-2}e^1) - (e^1, T_k^{-2}e^1)] - 2\frac{(e^k, T_k^{-2}e^1)}{(e^k, T_k^{-1}e^1)}\|\epsilon^k\|_A^2.$$

□

**3. Estimates of the $l_2$ norm of the error.** Our goal is to be able to compute estimates of the $l_2$ norm of the error using the formula of the previous section. Let us start by computing $(e^1, T_k^{-2}e^1)$. This can be done using a QR decomposition of the tridiagonal matrix $T_k$

$$Q_k T_k = R_k,$$

where $Q_k$ is an orthogonal matrix and $R_k$ an upper triangular matrix. We have $T_k^2 = R_k^T R_k$, therefore

$$(e^1, T_k^{-2}e^1) = (R_k^{-T}e^1, R_k^{-T}e^1).$$

Therefore we just have to solve a linear system with matrix $R_k^T$ and right hand side $e^1$. To compute the decomposition of $T_k$ we use the results of B. Fischer [4]. Let us look at the first steps of the reduction. To put a zero in the $(2, 1)$ position of

$$\begin{pmatrix} \alpha_1 \\ \eta_2 \end{pmatrix}$$

we define $\hat{r}_{1,1} = \alpha_1$, $r_{1,1} = \sqrt{\hat{r}_{1,1}^2 + \eta_2^2}$ with

$$c_1 = \frac{\hat{r}_{1,1}}{r_{1,1}}, s_1 = \frac{\eta_2}{r_{1,1}}.$$

When we apply this rotation to

$$\begin{pmatrix} \alpha_1 & \eta_2 \\ \eta_2 & \alpha_2 \end{pmatrix}$$

we obtain

$$\begin{pmatrix} r_{1,1} & r_{2,2} \\ 0 & \hat{r}_{1,2} \end{pmatrix} = \begin{pmatrix} r_{1,1} & c_1\eta_2 + s_1\alpha_2 \\ 0 & -s_1\eta_2 + c_1\alpha_2 \end{pmatrix}.$$

Then we reduce the column

$$\begin{pmatrix} r_{2,2} \\ \hat{r}_{1,2} \\ \eta_3 \end{pmatrix}$$

by a $s_2$ $c_2$ rotation. We obtain

$$\begin{pmatrix} r_{1,1} & r_{2,2} & r_{3,3} \\ 0 & r_{1,2} & r_{2,3} \\ 0 & 0 & \hat{r}_{1,3} \end{pmatrix}.$$

The general formulas are (see Fischer [4])

$$\hat{r}_{1,1} = \alpha_1, \; \hat{r}_{1,2} = c_1\alpha_2 - s_1\eta_2, \; \hat{r}_{1,n} = c_{n-1}\alpha_n - s_{n-1}c_{n-2}\eta_n, \; n \geq 3,$$

$$r_{1,n} = \sqrt{\hat{r}_{1,n}^2 + \eta_{n+1}^2},$$

$$r_{3,n} = s_{n-2}\eta_n, \; n \geq 3,$$

$$r_{2,2} = c_1\eta_2, \; r_{2,n} = c_{n-2}c_{n-1}\eta_n + s_{n-1}\alpha_n, \; n \geq 3,$$

$$c_n = \frac{\hat{r}_{1,n}}{r_{1,n}}, \; s_n = \frac{\eta_{n+1}}{r_{1,n}}.$$

Now, we would like to incrementally compute the solution of the linear systems $R_k^T w^k = e^1$ for $k = 1, 2, \ldots$ $R_k^T$ is a lower triangular matrix but we have to be careful that even though the other elements stay the same during the iterations, the $(k, k)$ element changes when we go from $k$ to $k+1$. Therefore, for $k = 1$ we have

$$w_1^1 = \frac{1}{\hat{r}_{1,1}},$$

and for $k = 2$

$$w_1^2 = \frac{1}{r_{1,1}}, w_2^2 = -\frac{r_{2,2}w_1^2}{\hat{r}_{1,2}}.$$

Hence changing notations, we define

$$\hat{w}_1 = \frac{1}{\hat{r}_{1,1}}, w_1 = \frac{1}{r_{1,1}}$$

$$\hat{w}_2 = -\frac{r_{2,2}w_1^2}{\hat{r}_{1,2}}, w_2 = -\frac{r_{2,2}w_1^2}{r_{1,2}}$$

and more generally for $n \geq 3$

$$\hat{w}_i = -\frac{(r_{3,i}w_{i-2} + r_{2,i}w_{i-1})}{\hat{r}_{1,i}}, w_i = -\frac{(r_{3,i}w_{i-2} + r_{2,i}w_{i-1})}{r_{1,i}}.$$

Therefore, $\hat{w}_k$ is the last component of the solution at iteration $k$ and $w_k$ will be used in the subsequent steps. Then,

$$\|R_k^{-T}e^1\|^2 = \sum_{j=1}^{k-1} w_j^2 + \hat{w}_k^2.$$

Now we proceed as we did in [8] and [9] for the $A$–norm of the error. We introduce an integer delay $d$ and we approximate $(r^0, A^{-2}r^0) - \|r^0\|^2(e^1, T_{k-d}^{-2}e^1)$ at iteration $k$ by the difference of the $k$ and $k-d$ terms computed from the solutions that is

$$\hat{w}_k^2 - \hat{w}_{k-d}^2 + \sum_{j=k-d}^{k-1} w_j^2,\ k > d.$$

To approximate the last term $(-1)^{k-d}2\eta_{k+1-d}\frac{\|r^0\|}{\|r^{k-d}\|}(e^{k-d}, T_{k-d}^{-2}e^1)\|\epsilon^{k-d}\|_A^2$ we use the approximation of $\|\epsilon^{k-d}\|_A$ we can compute from [8] (a lower bound obtained using Gauss quadrature) and the value $(e^{k-d}, T_{k-d}^{-2}e^1)$ which is $\hat{w}_{k-d}/\hat{r}_{1,k-d}$.

We can see that computing an estimate of the $l_2$ norm of the error add only a few operations to each CG iteration.

**4. Numerical experiments.** As test problems, we use some of the examples that were used in [6]. Example 3 arises from the 5–point finite difference approximation of a diffusion equation in a unit square,

$$-\text{div}(a\nabla u)) = f,$$

with Dirichlet boundary conditions. The diffusion coefficient in the $x$ direction is 100 if $x \in [1/4, 3/4]$, 1 otherwise. The coefficient in the $y$ direction is constant and equal to 1. We symmetrically scale the matrix by putting 1's on the diagonal. This corresponds to using a diagonal preconditioner. For this problem, we choose $n = 900$, the right hand side such that the exact solution $x_{ex}$ is $x_{ex} = (1, \dots, 1)^T$ and a random initial guess $x^0$. The results are given in figure 1.

Example 4 is taken from [7]. The matrix $A$ is diagonal. The diagonal elements are defined as

$$\mu_i = a + \frac{i-1}{n-1}(b-a)\rho^{n-i}, \quad i = 2, \dots, n-1 \quad \mu_1 = a,\ \mu_n = b$$

As in [7], we take $n = 48$, $a = 0.1$, $b = 100$ and $\rho = 0.875$. This is a difficult example for CG since we see in the result that we have to do much more than 48

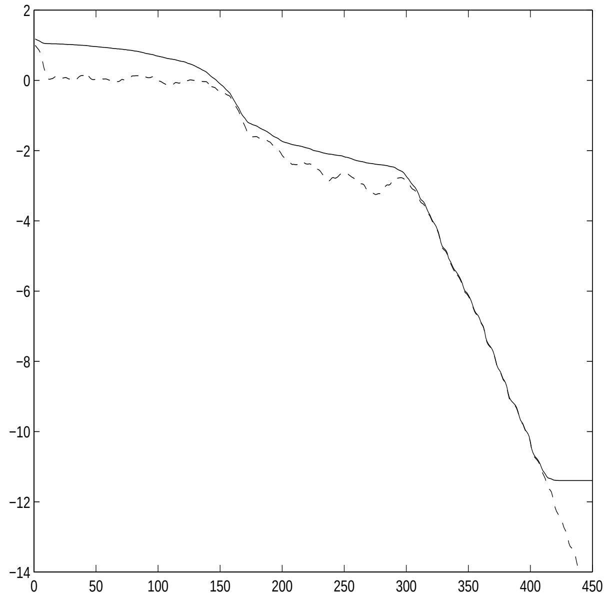


FIG. 1. *Example 3,* $d = 10$

iterations to reduce the error below $10^{-10}$. Since the last term in the error formula is always negative, it is likely that the sum of the first two terms could give (at least asymptotically) an upper bound. This is shown in figure 2. The relative difference between the exact $l_2$ error and the estimate is given in figure 3.

Another example is the matrix $1138 - bus$ from the Matrix Market collection (http://math.nist.gov). This is an impedance matrix of order 1138. We symmetrically scale the matrix. The results are given in figure 4.

**5. Using a preconditioner.** Let $M$ be a symmetric positive definite matrix which is going to be the preconditioner. It is well known that PCG for solving our linear system is obtained by applying CG to the transformed system

$$M^{-1/2}AM^{-1/2}(M^{1/2}x) = M^{-1/2}g,$$

for which the matrix is still symmetric positive definite. Then we obtain recurrences for the approximations to $x$ by going back to the original variables. Let $r^k = g - Ax^k$ and $y^k = M^{1/2}x^k$. For the preconditioned equation the residual is

$$\hat{r}^k = M^{-1/2}g - M^{-1/2}AM^{-1/2}y^k = M^{-1/2}(g - Ax^k) = M^{-1/2}r^k.$$

Let $z^k$ be given by solving $Mz^k = r^k$. Then, the scalar product we need in PCG is

$$(\hat{r}^k)^T \hat{r}^k = (\hat{r}^k, \hat{r}^k) = (M^{-1}r^k, r^k) = (z^k, r^k).$$

Moreover, let $\hat{p}^k = M^{1/2}p^k$. Then

$$(\hat{p}^k, M^{-1/2}AM^{-1/2}\hat{p}^k) = (p^k, Ap^k).$$

By using this change of variable, the PCG algorithm is the following:

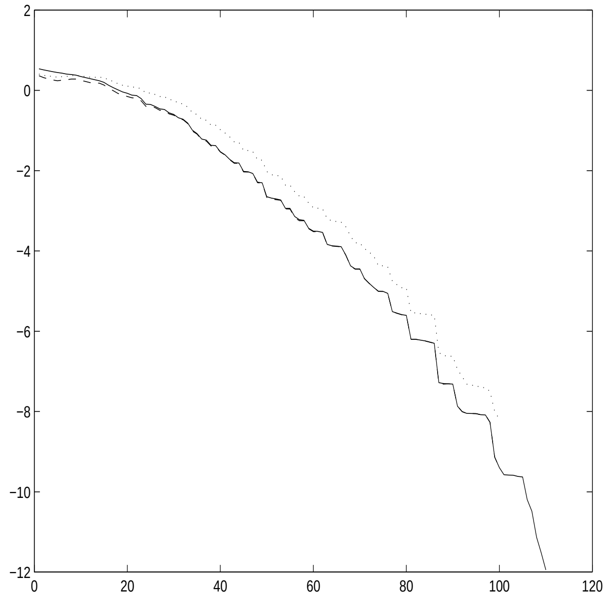


FIG. 2. *Example 4, $d = 10$, dashed: complete formula, dotted: first two terms*

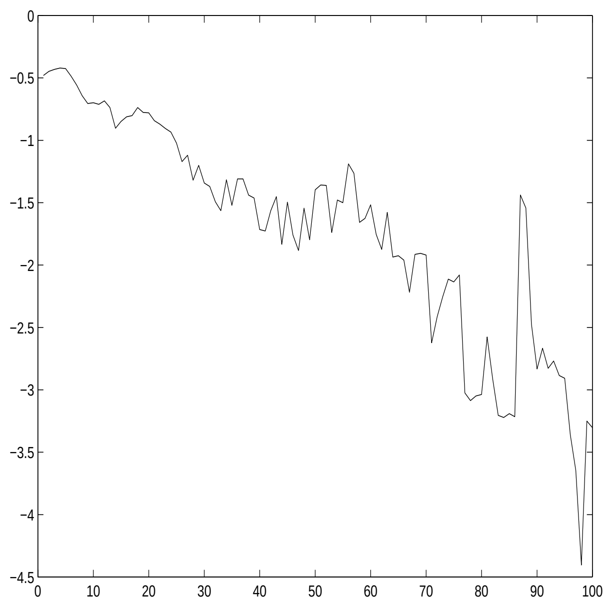


FIG. 3. *Example 4, $d = 10$, relative difference with the exact error*

let $x^0$ be given, $r^0 = g - Ax^0$, $Mz^0 = r^0$, $p^0 = z^0$, for $k = 1, \ldots$ until convergence

$$\alpha_{k-1} = \frac{{z^{k-1}}^T r^{k-1}}{{p^{k-1}}^T A p^{k-1}},$$

$$x^k = x^{k-1} + \alpha_{k-1} p^{k-1},$$

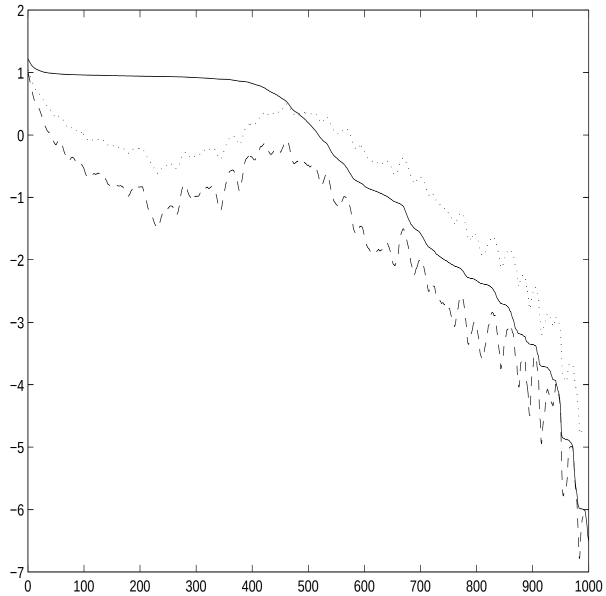


FIG. 4. *1138-bus, $d = 10$, dashed: complete formula, dotted: first two terms*

$$r^k = r^{k-1} - \alpha_{k-1} A p^{k-1},$$

$$M z^k = r^k,$$

$$\beta_k = \frac{{z^k}^T r^k}{{z^{k-1}}^T r^{k-1}},$$

$$p^k = z^k + \beta_k p^{k-1}.$$

Let $\hat{e}^k = y^k - y$ where $y = M^{1/2} x$ and $e^k = x^k - x$. Then,

$$\|\hat{e}^k\|^2_{M^{-1/2}AM^{-1/2}} = (M^{-1/2} A M^{-1/2}(y^k - y), y^k - y) = (A(x^k - x), x^k - x) = \|e^k\|^2_A.$$

This shows that for the $A$–norm we can use the formula

$$\|e^k\|^2_A = (z^0, r^0)((T_n^{-1})_{1,1} - (T_k^{-1})_{1,1}),$$

where the Lanczos matrix $T_k$ is constructed from the PCG coefficients. Unfortunately, things are not so nice for the $l_2$ norm since

$$\|\hat{e}^k\|^2 = (y^y - y, y^k - y) = (M(x^k - x), x^k - x) = \|e^k\|_M.$$

Therefore, directly translating the formula for the $l_2$ norm will only provide us with the $M$–norm of the error. However, let us suppose that $M = LL^T$ where $L$ is a triangular matrix. Then,

$$\|e^k\| \leq \|L^{-1}\| \|\hat{e}^k\|.$$

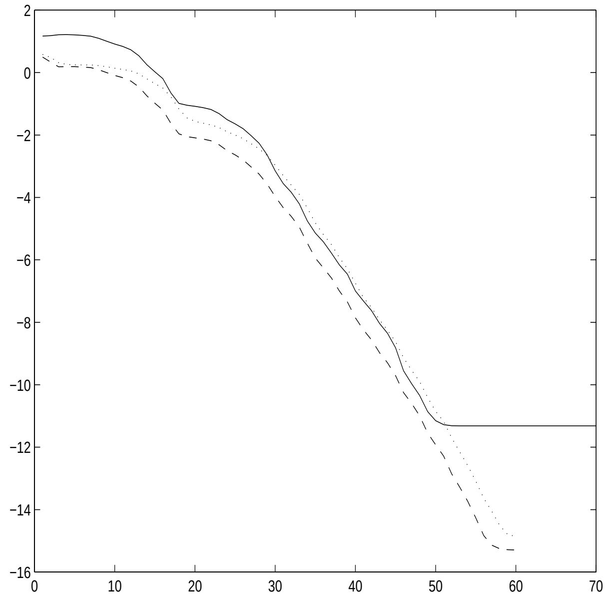


FIG. 5. *Example 3, $d = 10$, IC preconditioner, dashed: complete formula, dotted: first two terms*

It is difficult to compute or estimate the $l_2$ norm of $L^{-1}$. We will replace this by the $l_\infty$ norm of this matrix. We suppose that $M$ is an M–matrix. Then, $L^{-1}$ is a matrix with positive elements. If $w$ is the solution of $Lw = e$ where $e$ is the vector of all ones, then $l = \|L^{-1}\|_\infty = max_i w_i$. Hence,

$$\|e^k\| \simeq l\|\hat{e}^k\|_M.$$

When the matrix $A$ is symmetrically scaled with 1's on the diagonal, it turns out that it is not too bad to use $l = 1$. Results with this choice are given for example 3 and an incomplete Cholesky decomposition with no fill as a preconditioner on figure 5. Figure 6 shows results with an approximate inverse AINV preconditioner, see [1].

**6. Conclusion.** In this paper we have derived a formula relating the $l_2$ norm of the error to the $A$–norm of the error. This allows to compute an approximation of the $l_2$ norm by introducing a delay and using what was done previously for the $A$–norm. We also discussed what can be done when a preconditioner is used. These estimates are obtained by adding only a few floating point operations for each PCG iteration. Numerical results demonstrated that good estimates are obtained using these techniques.

REFERENCES

[1] M. BENZI, C.D. MEYER AND M. TUMA, *A sparse approximate inverse preconditioner for the conjugate gradient method*, SIAM J. Sci. Comput., vol 17, (1996), pp 1135–1149.

[2] Z. BAI AND G.H. GOLUB, *Bounds for the trace of the inverse and the determinant of symmetric positive definite matrices*, Ann. Numer. Math. v4 , (1997), pp 29–38.

[3] B. FISCHER AND G.H. GOLUB, *On the error computation for polynomial based iteration methods*, Report NA 92–21, Stanford University (1992).

[4] B. FISCHER, *Polynomial based iteration methods for symmetric linear systems*, Wiley–Tubner (1996).

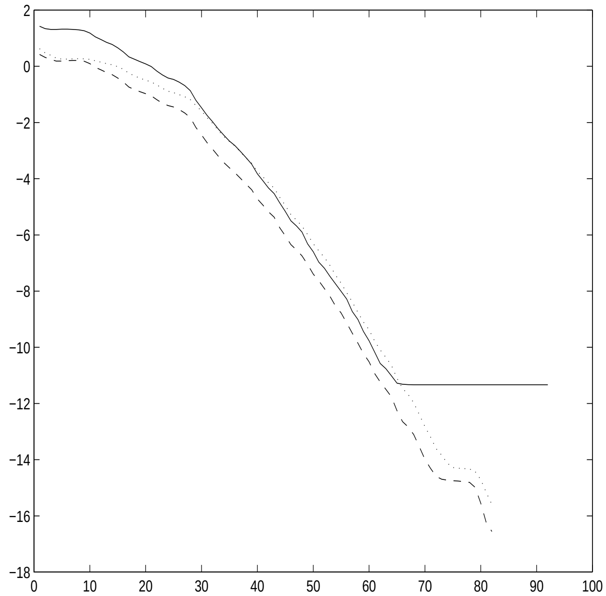


FIG. 6. *Example 3, $d = 10$, AINV preconditioner, dashed: complete formula, dotted: first two terms*

[5] G.H. Golub and G. Meurant, *Matrices, moments and quadrature*, in: *Numerical Analysis 1993*, eds. D.F. Griffiths and G.A. Watson, Pitman Research Notes in Mathematics, v 303, (1994), pp. 105–156.

[6] G.H. Golub and G. Meurant, *Matrices, moments and quadrature II or how to compute the norm of the error in iterative methods*, BIT, v 37 n3, (1997), pp 687–705.

[7] G.H. Golub and Z. Strakŏs, *Estimates in quadratic formulas*, Numerical Algorithms, v 8, no II–IV, (1994).

[8] G. Meurant, *The computation of bounds for the norm of the error in the conjugate gradient algorithm*, Numerical Algorithms v 16, (1997), pp 77–87.

[9] G. Meurant, *Numerical experiments in computing bounds for the norm of the error in the preconditioned conjugate gradient algorithm*, Numerical Algorithms v 22, (1999), pp 353–365.

[10] G. Meurant, *Computer solution of large linear systems*, North–Holland, (1999).

[11] Z. Strakŏs and P. Tichy, *On error estimates in the conjugate gradient method and why it works in finite precision computations*, ETNA, v 13, (2002), pp 56–80.