Journal of Scientific Computing, Vol. 15, No. 4, 2000

Solving $A\underline{x} = \underline{b}$ Using a Modified Conjugate Gradient Method Based on Roots of A

Paul F. Fischer[1] **and Sigal Gottlieb**[2]

Received January 23, 2001; accepted February 14, 2001

We consider the modified conjugate gradient procedure for solving $A\underline{x} = \underline{b}$ in which the approximation space is based upon the Krylov space associated with $A^{1/p}$ and $\underline{b}$, for any integer p. For the square-root MCG ($p = 2$) we establish a sharpened bound for the error at each iteration via Chebyshev polynomials in $\sqrt{A}$. We discuss the implications of the quickly accumulating effect of an error in $\sqrt{A}\,\underline{b}$ in the initial stage, and find an error bound even in the presence of such accumulating errors. Although this accumulation of errors may limit the usefulness of this method when $\sqrt{A}\,\underline{b}$ is unknown, it may still be successfully applied to a variety of small, "almost-SPD" problems, and can be used to jump-start the conjugate gradient method. Finally, we verify these theoretical results with numerical tests.

KEY WORDS: Modified conjugate gradient method; conjugate gradient method; Krylov space; convergence rate; stability.

1. INTRODUCTION

The modified conjugate gradient (MCG) method is based on the standard conjugate gradient (CG) method, which solves $A\underline{x} = \underline{b}$ (where $A \in \mathbb{R}^{n \times n}$ is symmetric positive definite) iteratively. At the kth iteration of CG, the solution $\underline{x}_k$ comes from the Krylov space

$$V^k = \mathscr{K}^k_{A,\underline{b}} := span\{\underline{b}, A\underline{b}, A^2\underline{b}, ..., A^{k-1}\underline{b}\}$$

[1] Math. and CS division, Argonne National Lab, Argonne, Illinois 60439; e-mail: fischer@mcs.anl.gov

[2] Department of Mathematics, UMASS-Dartmouth, North Dartmouth, Massachusetts 02747 and Division of Applied Mathematics, Box F, Brown University, Providence, Rhode Island 02912; e-mail: sg@cfm.brown.edu

0885-7474/00/1200-0441$18.00/0

In exact arithmetic, the CG method finds the best fit solution at each iteration [Lanczos (1952)]. That is, at the kth iteration, $\underline{x}_k \in V^k$ satisfies

$$\|\underline{x} - \underline{x}_k\|_A \leqslant \|\underline{x} - \underline{v}\|_A \qquad \forall \underline{v} \in V^k \tag{1.1}$$

where the A-norm is given by $\|\underline{w}\|_A = (\underline{w}^T A \underline{w})^{1/2}$. As a result of this best fit property and the polynomial form of the Krylov space, the error can be bounded as [Birkhoff and Lynch (1984); Golub and Van Loan (1989)]:

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x} - \underline{x}_0\|_A} \leqslant 2\left(\frac{\sqrt{\kappa_A} - 1}{\sqrt{\kappa_A} + 1}\right)^k \tag{1.2}$$

where κ_A is the condition number (the ratio of the largest eigenvalue to the smallest eigenvalue) of the matrix A. To improve upon the convergence rate, we must either change the condition number of the matrix A by preconditioning, or change the approximation space.

In Gottlieb and Fischer (1998) we presented a MCG method, which uses a finite-term recurrence and one multiplication by the matrix A per iteration to find the best-fit solution in the alternative approximation space, $\underline{x}_k \in \mathscr{K}^k_{\sqrt{A}, \underline{b}} = span\{\underline{b}, \sqrt{A}\,\underline{b}, (\sqrt{A})^2\,\underline{b}, \ldots, (\sqrt{A})^{k-1}\,\underline{b}\}$. This approximation space suggests an (optimal) error bound

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x} - \underline{x}_0\|_A} \leqslant 2\left(\frac{\sqrt{\kappa_M} - 1}{\sqrt{\kappa_M} + 1}\right)^k \tag{1.3}$$

which is based on κ_M, the condition number of $M = \sqrt{A}$. This error bound would be a significant improvement, since $\kappa_M = \sqrt{\kappa_A}$. With the usual initial guess $\underline{x}_0 = 0$, we previously obtained the error bound

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x}\|_A} \leqslant 2(k\kappa_M + 1)\left(\frac{\sqrt{\kappa_M} - 1}{\sqrt{\kappa_M} + 1}\right)^k \tag{1.4}$$

which is the optimal error bound multiplied by a factor which depends linearly on the condition number and the iterate. In this work, we sharpen the previous error bound to

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x}\|_A} \leqslant 2(2k + 1)\left(\frac{\sqrt{\kappa_M} - 1}{\sqrt{\kappa_M} + 1}\right)^k \tag{1.5}$$

which is a significant improvement, as the multiplicative factor now depends linearly only on the iterate, and there is no additional condition number dependence.

The idea is generalizable to the case where a series of vectors $\{\underline{b}, A^{1/p}\underline{b}, A^{2/p}\underline{b}, \ldots, A^{(p-1)/p}\underline{b}\}$ are known initially. In that case, at each iteration the modified method finds the best fit in the Krylov space $V^k = \mathscr{K}^k_{A,\underline{b}} := span\{\underline{b}, A^{1/p}\underline{b}, A^{2/p}\underline{b}, \ldots, A^{(k-1)/p}\underline{b}\}$. This is achieved with a finite term recurrence and only one matrix-vector multiplication per iteration. This approach will also be used to study the non-iterative approximation based on this series of vectors.

In theory, the MCG method could always be used instead of the CG method. In practice, however, the need for $A^{1/p}\underline{b}$ limits the use of this method. However, if $A^{1/p}\underline{b}$ can be approximated well and few iterations are needed, MCG may yield the optimal results. We will show analytically and numerically, (for $p = 2$), that the error in this initial approximation builds up at each iteration and adversely affects the convergence rate. At worst, the method still converges as $1/p$ the rate of CG (i.e., every pth iteration is a CG iteration). This suggests that even where the MCG method is not the optimal choice, it may be useful for a few iterations, to lower the residual before switching to the CG method.

2. THE MODIFIED CONJUGATE GRADIENT METHOD AND THE EFFECT OF ERROR IN $\sqrt{A}$

We begin with a brief review of the construction of the MCG method detailed in Gottlieb and Fischer (1998). We define $\underline{x}_k \in V^k$ to be the solution vector at the kth iteration. Now let $\underline{e}_k = \underline{x} - \underline{x}_k$ be the error at the kth iteration, and let $\underline{r}_k = A(\underline{x} - \underline{x}_k)$ be the residual at the kth iteration. Usually we set $\underline{x}_0 = 0$ and so $\underline{r}_0 = \underline{b}$. Each $\underline{x}_k$ is computed by

$$\underline{x}_k = \underline{x}_{k-1} + \alpha_k \underline{p}_k \tag{2.1}$$

For $\underline{x}_k$ to be the best fit solution in V^k we require

$$\alpha_k = \frac{\underline{p}_k^T \underline{b}}{\underline{p}_k^T A \underline{p}_k} = \frac{\underline{p}_k^T \underline{r}_{k-1}}{\underline{p}_k^T A \underline{p}_k} \tag{2.2}$$

where $\underline{p}_k \in V^k$ and $\{\underline{p}_j\}$ form an A-orthogonal set (i.e., $\underline{p}_j^T A \underline{p}_m = 0$ for $j \neq m$). To find such a set, $\underline{p}_k$ is chosen by picking a seed vector $\underline{v}_k \in V^k$ and using Gram–Schmidt orthogonalization with respect to $\{\underline{p}_j\}_1^{k-1}$.

$$\underline{p}_k = \underline{v}_k + \sum_{j=1}^{k-1} \beta_j \underline{p}_j \tag{2.3}$$

where

$$\beta_j = -\frac{\underline{p}_j^T A \underline{v}_k}{\underline{p}_j^T A \underline{p}_j} \tag{2.4}$$

The proposed square-root MCG (based on $\sqrt{A}$ and b) is obtained by taking the following sequence of seed vectors

$$\underline{v}_1 = \underline{r}_0 \tag{2.5}$$

$$\underline{v}_2 = \sqrt{A}\,\underline{r}_0 \tag{2.6}$$

$$\underline{v}_k = \underline{r}_{k-2}, \qquad \text{for} \quad k > 2 \tag{2.7}$$

which span the Krylov space $\mathscr{K}_{\sqrt{A},\,\underline{b}}$, and reduce (2.3) to the finite term recurrence

$$\underline{p}_k = \underline{v}_k + \beta_{k-3}\underline{p}_{k-3} + \beta_{k-2}\underline{p}_{k-2} + \beta_{k-1}\underline{p}_{k-1} \tag{2.8}$$

What is remarkable about this method is that it has essentially the same complexity as CG (in terms of matrix-vector products), and achieves a superior convergence rate by choosing a candidate search direction that is *not* the steepest descent, that is, by choosing $\underline{r}_{k-2}$ rather than the current residual $\underline{r}_{k-1}$.

If a sequence of vectors $\{A^{n/p}\underline{b}\}_{n=0}^{p-1}$ is known, the choice of seed vectors

$$\underline{v}_j = A^{(j-1)/p}\underline{b}, \qquad \text{for} \quad j = 1,\ldots, p$$

$$\underline{v}_k = \underline{r}_{k-p}, \qquad \text{for} \quad k > p$$

will yield the Krylov space $\mathscr{K}^k_{A^{1/p},\,\underline{b}}$, and a finite term recurrence

$$\underline{p}_k = \underline{v}_k + \sum_{j=1}^{2p-1} \beta_{k-j}\underline{p}_{k-j} \tag{2.9}$$

We expect this method to have an error bound which depends upon

$$\left(\frac{\sqrt{(\kappa_A)^{1/p}} - 1}{\sqrt{(\kappa_A)^{1/p}} + 1}\right)^k \tag{2.10}$$

This approach also suggests an error bound for a non-iterative approximation to the solution of $A\underline{x} = \underline{b}$. Given a sequence of vectors $\{A^{n/p}\underline{b}\}_{n=0}^{p-1}$, we can build the best-fit polynomial using the MCG procedure. This approximation will have an error bound asymptotically equal to (2.10) with $k = p - 1$.

The previous discussion assumed that we can readily find $\sqrt{A}\,\underline{b}$, or that some sequence of roots $\{A^{n/p}\}_{n=1}^{p-1}$ is known. Unfortunately, this is not always the case, and these quantities must usually be approximated. We now turn our attention to the case where $\sqrt{A}\,\underline{b}$ is approximated by $Q\underline{b}$, with some error $E = Q - \sqrt{A}$. The approximation space from which the (best-fit) kth iterate is taken is now $V^k = span\{\{\underline{b}, A\underline{b}, A^2\underline{b}, ..., A^{[(k-1)/2]}\underline{b}\} \cup \{Q\underline{b}, AQ\underline{b}, A^2Q\underline{b}, ..., A^{[(k-2)/2]}Q\underline{b}\}\}$, where $[n]$ denotes the integer part of n. The kth iterate may be written as:

$$\begin{aligned}\underline{x}_k &= \sum_{j=0}^{[(k-1)/2]} c_j A^j \underline{b} + \sum_{j=0}^{[(k-2)/2]} d_j A^j Q\underline{b} \\ &= P_{k-1}(\sqrt{A})\,\underline{b} + \tilde{P}_{[(k-2)/2]}(A)\,E\underline{b}\end{aligned}$$

Clearly, if the error $E = 0$, then the first polynomial $P_{k-1}(\sqrt{A})\,\underline{b}$ is the MCG best fit polynomial. The second polynomial $\tilde{P}_{(k-1)/2-1}(A)$ can be understood as amplifying the error E introduced by approximating $\sqrt{A}\,\underline{b}$, and it is the odd part of the first polynomial. For the class of functions traditionally used to bound CG-type methods, we observe that the odd parts of such polynomials grow quickly with the order of the polynomial (see Fig. 2.1). This implies that the error introduced by approximating $\sqrt{A}\,\underline{b}$ will grow and destroy the convergence rate. However, this error will not grow without bound, since if we consider an even polynomial $P_{k-1}(\sqrt{A})$, the polynomial multiplying E would be zero. This approach is equivalent to the best fit even polynomial in the krylov space of CG so that even with a large error in approximating $\sqrt{A}\,\underline{b}$, the MCG method would converge at half the rate of CG, or equivalent to CG at every second iteration. This is an interesting guarantee on the convergence of MCG regardless of the initial approximation error. This error analysis suggests that MCG is useful where E very small and few iterations are needed, such as in the case of a matrix with few eigenvalues or where few iterations are used to reduce the residual before starting the CG method, or when the CG method stalls.

This approach is applicable to linear systems $B^TB\underline{x} = \underline{b}$ for which B is not symmetric, but is close enough to symmetric so that the simple (and cheap to compute) approximation $\sqrt{B^TB} \approx \frac{1}{2}(B + B^T)$ is highly accurate. For such systems, E is small, and if the number of iterations needed is small, the initial error will not have a chance to grow and diminish the convergence rate. Such cases will be discussed in the numerical section. In Gottlieb and Fischer (1998) we observed instability in the MCG method, even in the absence of an approximation to $\sqrt{A}\,\underline{b}$. This behavior was apparent more quickly in codes run with 32-bit precision, and diminished in 64-bit precision and 128-bit precision. We suggest that this instability

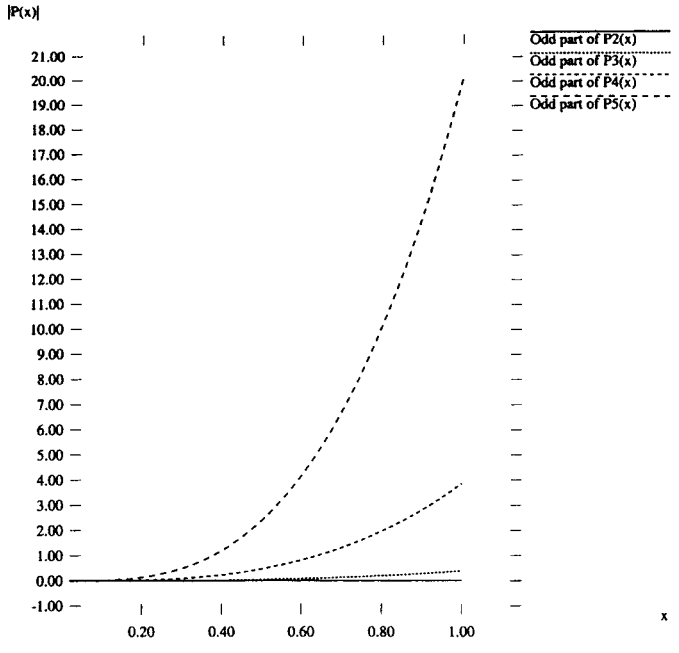

Fig. 2.1. The odd part of the polynomial $Q_{k+1}(x)$ used in the error analysis. This polynomial grows with its order, which implies that the error $E\underline{b}$ in the initial approximation of $\sqrt{A}\,\underline{b}$ will be amplified at each iteration.

can be explained as the result of machine-accuracy error in $\sqrt{A}\,\underline{b}$, and can be resolved only by using higher precision. This idea is validated by the fact that a series of numerical experiments in which E exists and is diminishing exhibits similar behavior. This behavior leads to a new view on preconditioning. A class of preconditioners which reduce the size of the relative error in $\sqrt{A}\,\underline{b}$ would help speed convergence. Traditional preconditioning aims to reduce the condition number of the iteration matrix. This new type of preconditioning would focus on making the approximation to $\sqrt{A}$ more accurate. Specifically, for the case $A = B^T B$ discussed above, a preconditioner would make the matrix B more symmetric in some sense.

3. A SHARPER ERROR BOUND FOR THE MODIFIED METHOD

Given initial vectors $\{\underline{b}, \sqrt{A}\,\underline{b}\}$ the square-root MCG method finds the best fit in a readily computed approximation space, at a fixed cost per iteration. The "best fit" property implies that, in exact arithmetic, strict error bounds can be computed for the kth iterate, independent of the

particular algorithm used to compute $\underline{x}_k$. In this section we derive a sharper bound for the case where the approximation space is taken as the Krylov space $\mathscr{K}^k_{M,\underline{b}}$ associated with the symmetric positive definite (SPD) matrix $M := \sqrt{A}$.

The best-fit property implies that the kth iterate, $\underline{x}_k \in \mathscr{K}^k_{M,\underline{b}}$, is to satisfy (1.1):

$$\begin{aligned} \|\underline{x} - \underline{x}_k\|_A &\leqslant \|\underline{x} - \underline{v}\|_A && \forall \underline{v} \in \mathscr{K}^k_{M,\underline{b}} \\ &\leqslant \|\underline{x} - P_{k-1}(M)\,\underline{b}\|_A && \forall P_{k-1}(x) \in \mathbb{P}_{k-1}(x) \end{aligned} \tag{3.1}$$

where $\mathbb{P}_{k-1}$ is the space of all polynomials of degree $k-1$ or less in the argument. Defining $\underline{r}_k = \underline{b} - A\underline{x}_k = A(\underline{x} - \underline{x}_k)$ as the residual at the kth iteration, we have $\|\underline{x} - \underline{x}_k\|_A = \|\underline{r}_k\|_{A^{-1}}$. The definition of M and (3.1) imply

$$\begin{aligned} &\text{For all } P_{k-1}(M) \in \mathbb{P}_{k-1}(M): \\ &\quad \|\underline{x} - \underline{x}_k\|_A \leqslant \|\underline{b} - M^2 P_{k-1}(M)\,\underline{b}\|_{A^{-1}} \\ &\qquad\qquad \leqslant \|I - M^2 P_{k-1}(M)\|_{A^{-1}}\, \|\underline{b}\|_{A^{-1}} \\ &\qquad\qquad = \|I - M^2 P_{k-1}(M)\|_{A^{-1}}\, \|\underline{x}\|_A \end{aligned} \tag{3.2}$$

where the matrix norm, $\|\cdot\|_{A^{-1}}$, is the natural norm induced by the same vector norm. If M and A are any two SPD matrices which commute, and $Q(M)$ is any polynomial in M, a straightforward calculation reveals that $\|Q(M)\|_{A^{-1}} = \|Q(M)\|_2 = \rho(Q(M))$, where $\rho(Q)$ is spectral radius of Q. Consequently, an upper bound on $\|\underline{x} - \underline{x}_k\|_A$ can be derived by choosing a polynomial $Q(M) := I - M^2 P_{k-1}(M)$ which minimizes $\rho(Q)$. Denoting the eigenvalues of M by μ_i, where $0 < \mu_1 \leqslant \cdots \leqslant \mu_n$, we have

$$\rho = \max_i |Q(\mu_i)| \leqslant \max_{\mu_1 \leqslant \mu \leqslant \mu_n} |Q(\mu)| \tag{3.3}$$

While the choice of P_{k-1} is arbitrary up to the maximal degree, $k-1$, the choice of $Q(x)$ is more restricted. Let $\mathbb{P}^{1,0}_{k+1}$ be the subset of $\mathbb{P}_{k+1}$ defined by

$$\mathbb{P}^{1,0}_{k+1} = \{q : q(0) = 1;\ q'(0) = 0;\ q \in \mathbb{P}_{k+1}\} \tag{3.4}$$

Clearly, $Q \in \mathbb{P}^{1,0}_{k+1}$. Combining (3.2) and (3.3), one obtains

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x}\|_A} \leqslant \min_{Q \in \mathbb{P}^{1,0}_{k+1}} \max_{\mu_1 \leqslant \mu \leqslant \mu_n} |Q(\mu)| \tag{3.5}$$

The class of polynomials defined by (3.4) has been studied extensively by B. Fischer under the heading of Hermite kernel polynomials. Although (3.5) is not addressed directly, it is noted in B. Fischer (1996) that polynomials in $\mathbb{P}^{1,0}_{k+1}$ can be expressed as a linear combination of the same translated and scaled Chebyshev polynomials $\hat{T}_k$ (defined below) that were used to derive (1.2). Although not the unique minimizer, the bound resulting from such a combination will be very close to optimal, as we now show.

We denote by $\hat{T}_k$ the translated and scaled Chebyshev polynomial

$$\hat{T}_k(x) := \frac{T_k\left(\dfrac{\mu_n+\mu_1-2x}{\mu_n-\mu_1}\right)}{T_k\left(\dfrac{\mu_n+\mu_1}{\mu_n-\mu_1}\right)} \tag{3.6}$$

where $T_k(x)$ is the Chebyshev polynomial [e.g., Saad (1996)],

$$T_k(x) = \tfrac{1}{2}((x+\sqrt{x^2-1})^k + (x-\sqrt{x^2-1})^k), \qquad |x| \geqslant 1 \tag{3.7}$$

$\hat{T}_k$ satisfies the classic minimax problem on $[\mu_1, \mu_n]$:

$$m_k := \max_{\mu_1 \leqslant \mu \leqslant \mu_n} |\hat{T}_k(\mu)| = \min_{q \in \mathbb{P}^1_{k+1}} \max_{\mu_1 \leqslant \mu \leqslant \mu_n} |q(\mu)| \tag{3.8}$$

where $\mathbb{P}^1_{k+1}$ is the set of polynomials defined by

$$\mathbb{P}^1_{k+1} = \{q : q(0) = 1;\, q \in \mathbb{P}_{k+1}\} \tag{3.9}$$

Defining $\sigma := (\mu_n+\mu_1)/(\mu_n-\mu_1)$, note that

$$m_k = \frac{1}{T_k(\sigma)} \tag{3.10}$$

Consider the polynomial $Q_{k+1}(x) = \alpha\hat{T}_{k+1}(x) + \beta\hat{T}_k(x)$. Since both $\hat{T}_{k+1}(x)$ and $\hat{T}_k(x)$ have the minimum possible extrema on $[\mu_1, \mu_n]$, this is clearly a reasonable starting point for solving the minimax problem (3.5). In order to satisfy the interpolatory constraints for $Q_{k+1} \in \mathbb{P}^{1,0}_{k+1}$, we must have

$$\alpha + \beta = 1$$

$$\alpha\hat{T}'_{k+1}(0) + \beta\hat{T}'_k(0) = 0$$

Solving for α and β yields

$$Q_{k+1}(x) = \frac{\hat{T}'_{k+1}(0)\,\hat{T}_k(x) - \hat{T}'_k(0)\,\hat{T}_{k+1}(x)}{\hat{T}'_{k+1}(0) - \hat{T}'_k(0)}$$

Note that $\hat{T}'_k(0)$ and $\hat{T}'_{k+1}(0)$ are of the same sign, whereas $\hat{T}_{k+1}(\mu_n)$ and $\hat{T}_k(\mu_n)$ are of opposite sign. Thus, we have

$$\max_{\mu_1 \leqslant \mu \leqslant \mu_n} |Q_{k+1}(\mu)| = \frac{\hat{T}'_k(0)\,m_{k+1} + \hat{T}'_{k+1}(0)\,m_k}{\hat{T}'_{k+1}(0) - \hat{T}'_k(0)}$$

Using (3.6) and (3.10), one obtains the bound

$$\max_{\mu_1 \leqslant \mu \leqslant \mu_n} |Q_{k+1}(\mu)| = \frac{T'_{k+1} + T'_k}{T_k T'_{k+1} - T_{k+1} T'_k} \tag{3.11}$$

where the argument of the Chebyshev polynomials and their derivatives is taken to be σ.

To compare this bound to the original CG result (1.2), we recast (3.11) in terms of $\kappa_M := \mu_n/\mu_1$, the condition number of M. Defining $a := \sigma + \sqrt{\sigma^2 - 1}$ and $b := \sigma - \sqrt{\sigma^2 - 1}$, we note the following identities:

$$\begin{aligned} a &= \frac{\sqrt{\kappa_M} + 1}{\sqrt{\kappa_M} - 1} \\ a \cdot b &= 1 \\ a > b &> 0 \\ T_k(\sigma) &= \frac{1}{2}(a^k + b^k) \\ T'_k(\sigma) &= \frac{k}{a-b}(a^k - b^k) \end{aligned} \tag{3.12}$$

The denominator of (3.11) is then:

$$\begin{aligned} &T_{k+1} T'_k - T_k T'_{k+1} \\ &\quad = \left| \frac{k+1}{2(a-b)}(a^k + b^k)(a^{k+1} - b^{k+1}) - \frac{k}{2(a-b)}(a^{k+1} + b^{k+1})(a^k - b^k) \right| \\ &\quad = k + \frac{1}{2} + \frac{a^{2k+1} - b^{2k+1}}{2(a-b)} \end{aligned} \tag{3.13}$$

while the numerator becomes

$$T'_k + T'_{k+1} = \frac{k}{2(a-b)}(a^k - b^k) + \frac{k+1}{2(a-b)}(a^{k+1} - b^{k+1}) \tag{3.14}$$

Combining (3.13) and (3.14) yields

$$\begin{aligned}
\max_{\mu_1 \leqslant \mu \leqslant \mu_n} |Q_{k+1}(\mu)| &= \frac{k(a^k - b^k) + (k+1)(a^{k+1} - b^{k+1})}{(a-b)(k+\frac{1}{2}) + \frac{1}{2}(a^{2k+1} - b^{2k+1})} \\
&\leqslant \frac{k(a^k - b^k) + (k+1)(a^{k+1} - b^{k+1})}{\frac{1}{2}(a^{2k+1} - b^{2k+1})} \\
&= \frac{2k(a^k - b^k)}{(a^{2k+1} - b^{2k+1})} + \frac{2(k+1)(a^{k+1} - b^{k+1})}{(a^{2k+1} - b^{2k+1})} \\
&= \frac{2k(a^k - b^k)}{a^{k+1}a^k - a^{k+1}b^k + (a^{k+1}b^k - b^{k+1}b^k)} \\
&\quad + \frac{2(k+1)(a^{k+1} - b^{k+1})}{a^k a^{k+1} - a^k b^{k+1} + (a^k b^{k+1} - b^k b^{k+1})} \\
&\leqslant \frac{2k(a^k - b^k)}{a^{k+1}a^k - a^{k+1}b^k} + \frac{2(k+1)(a^{k+1} - b^{k+1})}{a^k a^{k+1} - a^k b^{k+1}} \\
&= \frac{2k}{a^{k+1}} + \frac{2(k+1)}{a^k} \\
&\leqslant \frac{2(2k+1)}{a^k}
\end{aligned} \tag{3.15}$$

Taking (3.15) in conjunction with the first of the identities (3.12), we obtain the desired result

$$\frac{\|\underline{x} - \underline{x}_k\|_A}{\|\underline{x}\|_A} \leqslant 2(2k+1)\left(\frac{\sqrt{\kappa_M} - 1}{\sqrt{\kappa_M} + 1}\right)^k \tag{3.16}$$

Although this bound has an extra factor of $(2k+1)$ not present in (1.2), the fact that it is based upon $\kappa_M = \sqrt{\kappa_A}$ implies that the modified approach should yield a much better convergence rate than the standard CG algorithm. A comparison of this new error bound (3.16), the previous error bound (1.4), the optimal error bound (1.3) and the CG error bound (1.2) is shown in Fig. 3.1.

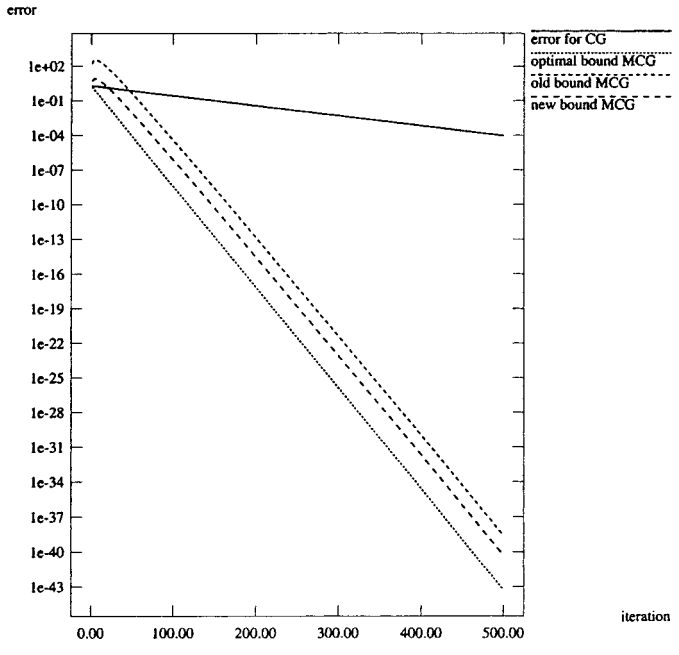

Fig. 3.1. Comparison of the CG and MCG error bounds. The graph shows the log(error) as a function of the number of iterations. Clearly, the MCG bounds show the error decaying at a rate which is a significant improvement over the error decay rate associated with CG.

4. NUMERICAL EXPERIMENTS

Example 1. The MCG and CG methods are applied to the problem $A\underline{x} = \underline{b}$ where $A \in \mathbb{R}^{1600 \times 1600}$ is the two dimensional Laplacian operator given by second-order finite differences on a regular array of points:

$$(Av)_{i,j} = 4v_{i,j} - v_{i+1,j} - v_{i-1,j} - v_{i,j+1} - v_{i,j-1}$$

with corresponding Dirichlet boundary conditions. The right hand side $\underline{b}$ is given by

$$b_j = 100 \sin(100 \cos(j))$$

In this case, the square-root of the matrix A is unknown, but is approximated to a high degree of accuracy, using a method developed by van der Vorst (1987). The performance of the MCG method is compared to that of the standard CG. The initial behavior of MCG is exactly as predicted (Fig. 4.1, right), following along the error-bound almost exactly.

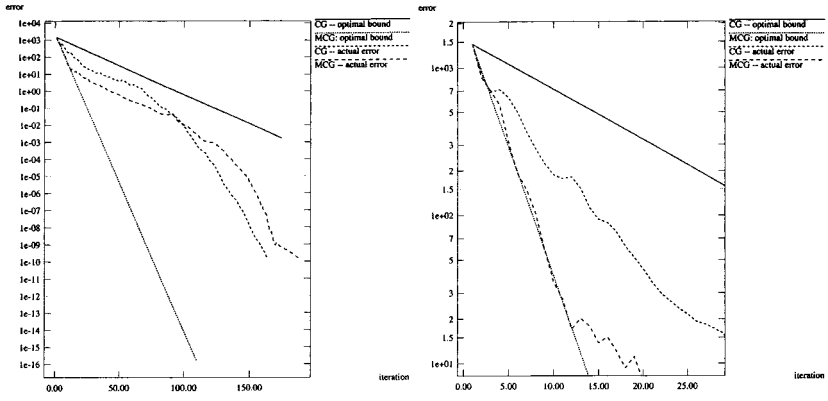

Fig. 4.1. This is a comparison of the performance of MCG and CG for Example 1, and the theoretical error bounds of MCG and CG. the initial convergence of MCG matches perfectly with the optimal error bound initially (right). MCG initially decays as predicted by the error bound and outperforms CG, but later (left) the error associated with the approximation of $\sqrt{A}\,\underline{b}$ grows and causes convergence to slow down. near 100 iterations MCG begins to converge at half the rate of CG, however, at no point does the error decay at twice the rate of the CG error bound.

Example 2. MCG and CG are applied to a series of problems of the form

$$B^T B\underline{x} = \underline{b}$$

where B are 150×150 matrices of the form:

$$B = \begin{pmatrix} 2.5 & -1+\varepsilon & 0 & 0 & 0 & \cdots & 0 \\ -1 & 2.5 & -1+\varepsilon & 0 & 0 & \cdots & 0 \\ 0 & -1 & 2.5 & -1+\varepsilon & 0 & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & -1 & 2.5 & -1+\varepsilon \\ 0 & 0 & 0 & 0 & 0 & -1 & 2.5 \end{pmatrix}$$

and ε, depending on the case, takes values $10^{-8} \leqslant \varepsilon \leqslant 10^{-2}$. In all cases, $\underline{b}$ is given by $b_j = \sin(100\cos(100j))$. The square-root initial vector $\sqrt{A}\,\underline{b}$ is approximated by

$$\sqrt{B^T B}\,\underline{b} \approx \frac{B^T + B}{2}\,\underline{b}$$

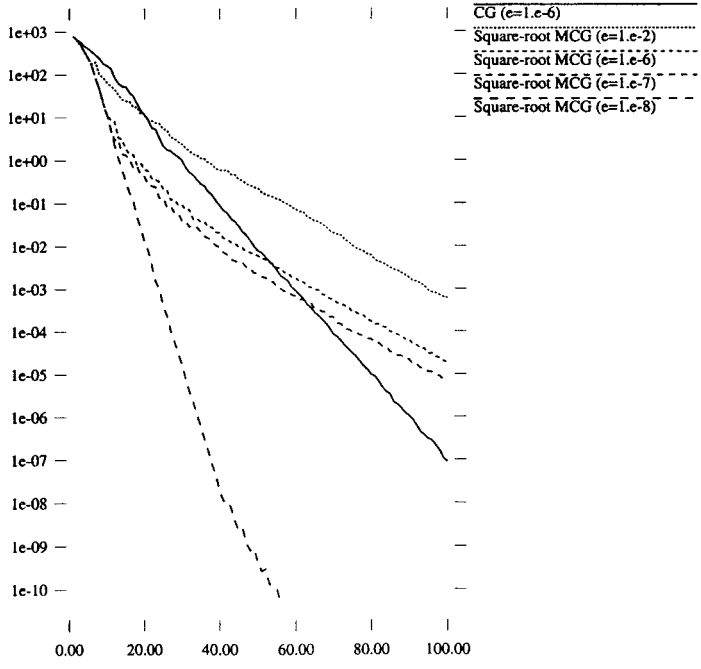

Fig. 4.2. CG and the square-root MCG were applied as in Example 2, for a few different values of ε. CG behaved almost exactly the same regardless of the value of ε, so one representative case is shown. The behavior of MCG differed widely depending on the value of ε. When $\varepsilon = 10^{-2}$, the error buildup diminished the convergence rate after very few iterations. At the level of $\varepsilon = 10^{-6}$ and $\varepsilon = 10^{-7}$, the initial convergence of MCG is a significant improvement for the first 60 or so iterations, at which point there is a crossover, and CG is better than MCG. However, for $\varepsilon = 10^{-8}$ MCG is a significant improvement over CG.

Figure 4.2 shows the effect of the error in the square-root by varying ε. When ε is small, this approximation is almost exact, and the error-vector E is small. When ε grows, E grows correspondingly. To limit the effect of round-off error, the MCG codes were run in 128-bit precision. The effect is remarkably as we expect. The MCG method converges significantly faster than CG when ε is very small. As ε grows, we see the MCG method exhibiting slowed convergence at an earlier iteration number. This diminished convergence causes MCG to converge slower than CG in the cases where $\varepsilon \geqslant 10^{-7}$.

Example 3. The cube-root MCG, square-root MCG and CG were used to solve $A\underline{x} = \underline{b}$ where A is a 900×900 diagonal matrix

$$A = \begin{pmatrix} \lambda_1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & 0 & \cdots & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & 0 & 0 & 0 & 0 & \lambda_{900} \end{pmatrix}$$

with λ_i given by

(a)

$$\lambda_1 = 0.034, \qquad \lambda_2 = 0.082, \qquad \lambda_3 = 0.127, \qquad \lambda_4 = 0.155, \qquad \lambda_5 = 0.19$$

$$\lambda_i = 0.2 + \frac{i-5}{895}\, i = 6, 900$$

(b)

$$\lambda_1 = 214.827, \qquad \lambda_2 = 57.4368, \qquad \lambda_3 = 48.5554, \qquad \lambda_4 = 35.0624$$

$$\lambda_5 = 27.3633, \qquad \lambda_6 = 21.8722 \qquad \lambda_7 = 17.7489$$

$$\lambda_i = 1.0 + 15.6624\, \frac{i-8}{892}\, i = 8, 900$$

Figure 4.3 compares the standard CG method, the square-root MCG and the cube-root MCG. The performance is as predicted, with the cube-root method converging fastest, the square-root MCG next and the standard method last. Although this was an idealized case, in which the matrix was a diagonal matrix and the square- and cube-roots were calculated directly, and the MCG codes were run with 128-bit precision, both MCG methods eventually suffer from instability after the error was below 10^{-11}. This did not seem to affect convergence, but if 128-bit precision was not used, the effect of round-off error would be seen much sooner. These two cases show that the ideal behavior of MCG is indeed as predicted.

5. CONCLUSIONS

We generalize the MCG method to cases where we have a available a sequence of vectors $\{\underline{b}, A^{1/p}\underline{b}, A^{2/p}\underline{b}, \ldots, A^{(p-1)/p}\underline{b}\}$, and discuss the approximation properties of such a method, as well as that of the non-iterative approximation based on this sequence. A sharpened bound is obtained for the square-root case, as well as an analysis of the effect of an error in the initial approximation of $\sqrt{A}\,\underline{b}$. We suspect that this effect may be a cause

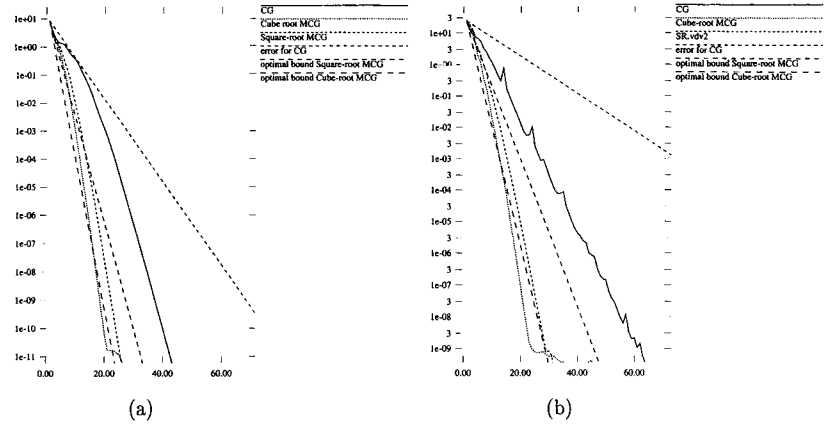

Fig. 4.3. The cube-root and square-root MCG methods are compared to CG for the two van der Vorst matrices (a) and (b) in Example 3, respectively. The MCG methods initially exceed the optimal bound, but later converge faster, verifying the claim that the bound will have some iteration-related term multiplied to the optima-bound term we're using. The diminished convergence; which we explain as the effect of round-off level error in the $\sqrt{A}$ is not visible, as it occurs after the residual was cut to below 10^{-11}, but it does occur a little later.

of instability in the MCG method, and suggest that a new approach to preconditioning may resolve this problem. This method is still applicable to cases in which we are solving $B^T B\underline{x} = \underline{b}$ where B is close to symmetric, and the number of iterations needed is small. In the numerical experiments, we verify that the initial convergence rate of the square-root ($p = 2$) and cube-root ($p = 3$) MCG is a significant improvement over CG, converging at a rate which depends upon with the pth-root of the condition number. We also observe, as predicted, the accumulating effect of an error in $\sqrt{A}$ at the initial stage, and at worst, equivalence of the convergence rate of MCG to CG at every second iteration.

ACKNOWLEDGEMENTS

This was supported by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, U.S. Department of Energy, under Contract W-31-109-Eng-38.

REFERENCES

Birkhoff, G., and Lynch, R. E. (1984). *Numerical Solution of Elliptic Problems*, SIAM, Philadelphia.

Fischer, B. (1996). *Polynomial Based Iteration Methods for Symmetric Linear Systems*, Wiley–Teubner, Chichester, Stuttgart.

Golub, G. H., and Van Loan, C. H. (1989). *Matrix Computations*, The John Hopkins University Press, Baltimore.

Gottlieb, S., and Fischer, P. F. (1998). A modified conjugate gradient method for the solution of $Ax = b$ based upon b and $A^{1/2}b$. *J. of Sci. Comput.* **13**(2), 173–183.

Lanczos, C. (1952). Solution of systems of linear equations by minimized iterations. *J. of Research of the National Bureau of Standards* **49**, 33–53.

O'Leary, D. P. (1980). The block conjugate gradient algorithm and related methods. *Linear Algebra and its Appl.* **29**, 293–322.

Simon, H. D. (1984). The Lanczos method with partial reorthogonalization. *Math. of Comp.* **42**, 115–142.

Van der Vorst, H. A. (1987). An iterative solution method for solving $f(A)\ \underline{x} = \underline{b}$, using Krylov subspace information obtained for the symmetric positive definite matrix A. *J. of Comp. and Appl. Math.* **18**, 249–263.