Universiteit Utrecht

Department of Mathematics

Accurate conjugate gradient methods for shifted systems

by

Jasper van den Eshof and Gerard L. G. Sleijpen

Preprint

nr. 1265

January, 2003

ACCURATE CONJUGATE GRADIENT METHODS FOR SHIFTED SYSTEMS

JASPER VAN DEN ESHOF* AND GERARD L. G. SLEIJPEN*

Abstract

We present an efficient and accurate variant of the conjugate gradient method for solving families of shifted systems. In particular we are interested in shifted systems that occur in Tikhonov regularization for inverse problems since these problems can be sensitive to roundoff errors. The success of our method in achieving accurate approximations is supported by theoretical arguments as well as several numerical experiments and we relate it to other implementations proposed in literature.

1 Introduction

Let $\mathbf{A}$ be a $n \times n$ and nonsingular real matrix. We are interested in solving real linear systems

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \tag{1.1}$$

and, additionally, for various real values of τ

$$(\mathbf{A} + \tau\mathbf{I})\mathbf{x}^{\tau} = \mathbf{b}, \tag{1.2}$$

where $\mathbf{I}$ is the identity matrix. These problems arise quite naturally in various applications. Krylov subspace methods are iterative methods for solving general linear systems as for example (1.1). These methods, with zero initial guess, construct their approximations in step j from the so–called j dimensional *Krylov subspace* defined as $\mathcal{K}_j(\mathbf{A}, \mathbf{b}) \equiv \mathrm{span}\{\mathbf{b}, \mathbf{A}\mathbf{b} \ldots, \mathbf{A}^{j-1}\mathbf{b}\}$. Modern Krylov subspace methods are often characterized by the way they incrementally build up a basis for the Krylov subspaces and how they construct their approximations from them using some optimality property. An important property of Krylov subspaces is that they are shift invariant, that is $\mathcal{K}_j(\mathbf{A}, \mathbf{b}) = \mathcal{K}_j(\mathbf{A} + \tau\mathbf{I}, \mathbf{b})$. By exploiting this property, Equation (1.2) can be solved for various values of the shift τ by constructing a basis for the Krylov subspace only once. This observation has led to many efficient implementations of known Krylov subspace methods that can handle multiple shifts. We refer the interested reader for further information to [3, 5, 8, 10, 15, 7, 19, 9, 20].

If the matrix $\mathbf{A}$ is symmetric, positive definite and τ is positive then the system (1.2) can be solved using the celebrated *conjugate gradient method* (CG) [14]. Equivalently, we can apply the *Lanczos method*, e.g., [11, Chapter 9], to $\mathbf{A}$ with starting vector $\mathbf{b}$ to construct an orthonormal basis for the Krylov subspace $\mathcal{K}_k(\mathbf{A}, \mathbf{b})$, which is summarized by the *Lanczos relation*

$$\mathbf{A}\mathbf{V}_k = \mathbf{V}_k T_k + \delta_{k-1}\mathbf{v}_k e_k^{\mathrm{T}} = \mathbf{V}_{k+1}\underline{T}_k,$$

where the columns $\mathbf{v}_0, \ldots, \mathbf{v}_k$ of $\mathbf{V}_{k+1}$, form an orthonormal basis for $\mathcal{K}_{k+1}(\mathbf{A}, \mathbf{b})$ and the symmetric $k \times k$ tridiagonal matrix T_k collects the coefficients computed during the execution

*Mathematical Institute, Utrecht University, P.O. Box 80.010, NL-3508 TA Utrecht, The Netherlands. `www.math.uu.nl/people/[eshof,sleijpen]` Email: `[eshof,sleijpen]@math.uu.nl`.

of the Lanczos algorithm. It is often convenient to include also δ_{k-1} into one $k+1$ by k tridiagonal matrix $\underline{T}_k$ by adding an additional row to T_k. The CG approximations for (1.2) are equal to

$$\mathbf{x}_k^\tau = \mathbf{V}_k(T_k + \tau I)^{-1} e_1 \sqrt{\phi_0}, \quad \text{with } \phi_0 \equiv \mathbf{r}_0{}^{\mathrm{T}}\mathbf{r}_0, \tag{1.3}$$

and $\mathbf{r}_0 = \mathbf{b}$ being the initial residual. A *multi–shift* conjugate gradient method constructs the orthonormal basis once and at the same time computes (1.3) for the required values of τ. Specific implementations are presented in [15] and [7]. We will discuss their relationship to our implementation in the course of this paper.

A reliable implementation of the multi–shift CG method can achieve the same accuracy for the shifted systems as when the CG method is directly applied to each individual system. In this paper we are interested in accurate implementations of (1.3) when the computations are affected by roundoff errors. To this purpose we restrict our attention to the more specific problem of computing solutions to

$$(\mathbf{A}^{\mathrm{T}}\mathbf{A} + \tau\mathbf{I})\mathbf{x}^\tau = \mathbf{A}^{\mathrm{T}}\mathbf{b}. \tag{1.4}$$

This system is more sensitive to the effects of using computer arithmetic since $\mathbf{A}^{\mathrm{T}}\mathbf{A} + \tau\mathbf{I}$ can be ill–conditioned. Furthermore, (1.4) has important applications in Tikhonov regularization [7] and the computation of the Overlap operator in QCD [16].

To understand the influence of finite precision arithmetic on the multi–shift CG method we will discriminate between rounding errors made in the construction of the basis for the Krylov subspace (the Lanczos part) and the inversion of the tridiagonal matrix in (1.3). This paper has the following structure. In Section 2 we review the CG method and its variant for least squares problems (CGLS) and we discuss their use as alternative Lanczos type methods. Section 3 deals with the influence on the approximation in (1.3) of rounding errors made in the "alternative" Lanczos method. The topic of Section 4 is the accurate computation of the inversion in (1.4). Finally, we show by several numerical experiments that, if all ingredients are chosen properly, we can achieve high accuracy for the shifted systems.

2 Conjugate gradient methods

In the conjugate gradient method of Hestenes and Stiefel [14] the *residuals* corresponding to the iterates, $\mathbf{r}_j = \mathbf{b} - \mathbf{A}\mathbf{x}_j$, are computed for $j = 1, \ldots, k$ using the recurrences

$$\mathbf{r}_j = \mathbf{r}_{j-1} - \alpha_{j-1}\mathbf{c}_{j-1}, \quad \mathbf{c}_{j-1} = \mathbf{A}\mathbf{p}_{j-1}, \quad \mathbf{p}_j = \mathbf{r}_j + \beta_{j-1}\mathbf{p}_{j-1}, \tag{2.1}$$

with the coefficients given by

$$\alpha_{j-1} \equiv \frac{\phi_{j-1}}{\mathbf{p}_{j-1}^{\mathrm{T}}\mathbf{c}_{j-1}}, \quad \beta_{j-1} \equiv \frac{\phi_j}{\phi_{j-1}}, \quad \phi_j \equiv \|\mathbf{r}_j\|^2,$$

and, initially, $\mathbf{p}_0 = \mathbf{r}_0 = \mathbf{b}$. The norms in this paper are Euclidean. The approximate solution then follows using the recurrence

$$\mathbf{x}_j = \mathbf{x}_{j-1} + \alpha_{j-1}\mathbf{p}_{j-1}, \quad \text{with } \mathbf{x}_0 = \mathbf{0}. \tag{2.2}$$

In practice nonzero starting vectors are sometimes used. We will assume that the initial guess, $\mathbf{x}_0$, is zero here. A key characterization of the CG method is that its iterates, $\mathbf{x}_j$, minimize the error in the energy norm (that is an $\mathbf{A}$-weighted norm) over all approximations from the j-th Krylov subspace $\mathcal{K}_j(\mathbf{A}, \mathbf{b})$. As a consequence of this, the residuals $\mathbf{r}_i$ for $i = 0, \ldots, j-1$ form an orthogonal basis for $\mathcal{K}_j(\mathbf{A}, \mathbf{b})$.

For some symmetric positive definite matrices it is proved, but it is often observed in experiments, that in finite precision arithmetic the computed vector $\mathbf{r}_j$ becomes much smaller than machine precision for large j. In these cases Greenbaum [12] showed that for the iterate, $\mathbf{x}_k$, we essentially have for large enough k that

$$\|\mathbf{b} - \mathbf{A}\mathbf{x}_k\| \leq \varepsilon\, \mathcal{O}(1)\, \|\mathbf{A}\|\|\mathbf{x}\|. \tag{2.3}$$

Here, ε is the *unit roundoff*, which for double precision computations is in the order of 10^{-16}. Note that Equation (2.3) implies the following bound on the relative error:

$$\|\mathbf{x} - \mathbf{x}_k\|/\|\mathbf{x}\| \leq \varepsilon\, \mathcal{O}(1)\, \|\mathbf{A}\|\|\mathbf{A}^{-1}\|.$$

For least squares problems the CG method can be directly applied to the normal equations. Nevertheless it is often suggested to use an alternative but mathematically equivalent variation of CG known as CGLS [14, Section 10] in this case. For $j = 1, \ldots, k$ this method is defined by the following recurrence relations:

$$\mathbf{z}_j = \mathbf{z}_{j-1} - \alpha_{j-1}\mathbf{c}_{j-1}, \quad \mathbf{c}_{j-1} = \mathbf{A}\mathbf{p}_{j-1}, \quad \mathbf{r}_j = \mathbf{A}^{\mathrm{T}}\mathbf{z}_j, \quad \mathbf{p}_j = \mathbf{r}_j + \beta_{j-1}\mathbf{p}_{j-1}, \tag{2.4}$$

with

$$\alpha_{j-1} \equiv \frac{\phi_{j-1}}{\mathbf{c}_{j-1}^{\mathrm{T}}\mathbf{c}_{j-1}}, \quad \beta_{j-1} \equiv \frac{\phi_j}{\phi_{j-1}}, \quad \phi_j \equiv \|\mathbf{r}_j\|^2, \tag{2.5}$$

and $\mathbf{z}_0 = \mathbf{b}$, $\mathbf{r}_0 = \mathbf{p}_0 = \mathbf{A}^{\mathrm{T}}\mathbf{z}_0$, and $\mathbf{x}_k$ as in (2.2).

The advantage of this method, compared to applying CG directly to the normal equations, is that here the least squares residuals, $\mathbf{z}_j = \mathbf{b} - \mathbf{A}\mathbf{x}_j$, are directly available. Furthermore it was shown in [12, Section 3.3] and [2], with similar arguments as used for (2.3) for CG, that recurring the residuals for the least squares problem, $\mathbf{z}_j$, improves the attainable accuracy of the method. Note that the CGLS residuals for the normal equation, $\mathbf{r}_i$, form an orthogonal basis for the Krylov subspace $\mathcal{K}_j(\mathbf{A}^{\mathrm{T}}\mathbf{A}, \mathbf{A}^{\mathrm{T}}\mathbf{b})$. For the CG method it is well-known that the coupled two–term recurrences of the CG method can be used as an alternative to the Lanczos method for constructing an orthonormal basis for the Krylov subspace, see [6, 1]. Similarly, the recurrences in (2.4) can be used to build an orthonormal basis for the Krylov subspace $\mathcal{K}_j(\mathbf{A}^{\mathrm{T}}\mathbf{A}, \mathbf{A}^{\mathrm{T}}\mathbf{b})$. For future convenience we work this out for this latter method.

First a little remark about notational conventions: with $\mathbf{R}_k$ we denote the $n \times k$ matrix with columns $\mathbf{r}_0, \ldots, \mathbf{r}_{k-1}$. Similarly, other capitals will be used to group together the corresponding vectors. Now, the relations in (2.4) can be summarized by the following matrix formulations

$$\mathbf{Z}_{k+1}\underline{J}_k = \mathbf{C}_k\Delta_k, \quad \mathbf{C}_k = \mathbf{A}\mathbf{P}_k, \quad \mathbf{R}_{k+1} = \mathbf{A}^{\mathrm{T}}\mathbf{Z}_{k+1}, \quad \mathbf{P}_kU_k = \mathbf{R}_k,$$

where

$$U_k \equiv \begin{bmatrix} 1 & -\beta_0 & & & \\ & 1 & -\beta_1 & & \\ & & \ddots & \ddots & \\ & & & \ddots & -\beta_{k-2} \\ & & & & 1 \end{bmatrix},$$

$\underline{J}_k$ is a $k+1 \times k$ lower bidiagonal matrix with 1 and -1 on, respectively, the diagonal and sub-diagonal, and $\Delta_k \equiv \mathrm{diag}(\alpha_0, \ldots, \alpha_{k-1})$. Substitution yields the *residual relation*:

$$(\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{R}_k = \mathbf{R}_{k+1}\underline{S}_k, \quad \text{with } \underline{S}_k \equiv \underline{J}_k \Delta_k^{-1} U_k. \tag{2.6}$$

If we introduce the diagonal matrix Φ_k with diagonal elements $\sqrt{\phi_0}, \ldots, \sqrt{\phi_{k-1}}$, then we find the *Lanczos relation*:

$$\begin{aligned} &(\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{V}_k = \mathbf{V}_k T_k - \frac{\sqrt{\beta_{k-1}}}{\alpha_{k-1}}\mathbf{v}_k e_k^{\mathrm{T}} = \mathbf{V}_{k+1}\underline{T}_k, \\ &\text{with} \quad T_k = L_k^{\mathrm{T}}\Delta_k^{-1}L_k, \quad \mathbf{V}_{k+1} = \mathbf{R}_{k+1}\Phi_{k+1}^{-1}, \quad \text{and} \quad L_k = \Phi_k U_k \Phi_k^{-1}. \end{aligned} \tag{2.7}$$

Since the ϕ_j are available in the CGLS method, the recurrences in (2.4) can be used as an alternative to applying the Lanczos method to $\mathbf{A}^{\mathrm{T}}\mathbf{A}$ with starting vector $\mathbf{A}^{\mathrm{T}}\mathbf{b}$ at virtually the same cost. We note that the vectors $\mathbf{v}_j$ (i.e., the columns of $\mathbf{V}_k$) are plus or minus the Lanczos vectors. We will ignore this slight difference.

Bai and Freund show in [1] that there are advantages of using a Lanczos method based on coupled two-term recurrences in applications as reduced order modeling. There, the matrix $\mathbf{A}$ is symmetric, positive semidefinite. Applying the standard Lanczos method can result in slightly indefinite tridiagonals T_k (due to roundoff errors). Constructing the tridiagonal in factorized form as in (2.7) cures this problem. They argue that for these applications the alternative Lanczos process is more accurate and robust. Their Lanczos method is of band-type but is similar to using the CG method and the relation in (2.7). Considering the recent work on the accurate computation of eigenvalues and eigenvectors of tridiagonal matrices, e.g., [4] for an overview and references, raises the question if the alternative Lanczos method based on two–term recurrences can offer relatively more accurate approximations to small eigenvalues.[1] The computation of the Lanczos approximation (1.3) is another potential example of the advantages of using a different Lanczos method. In the next section we collect evidence of the advantage of using the CGLS method as Lanczos type method over the standard method. In Section 4 we discuss the efficient and accurate computation of the vector $\mathbf{x}_k^\tau$ in (1.3).

3 The effect of errors in the "Lanczos" process

We recall that we assume that $\mathbf{A}$ is square and invertible. The Lanczos relation (2.7) is computed by constructing a diagonal scaling and, subsequently, scaling (2.6). We will assume that this is done exactly. It can be easily checked that this assumption is not essential. Furthermore, no rounding errors in the computation in (1.3) are considered in this section. We also restrict our attention here to the case $\tau = 0$. The analysis for this simple instance already demonstrates the differences between the alternatives. It also contains the essential ingredients for a theoretical analyis for general τ, but such an analysis is much more involved and is not given here. The effectiveness of our approach for general τ is demonstrated by numerical experiments in the Sections 4 and 5.

[1]Unfortunately for eigenvalue computations this can not be the case in general since an ordinary matrix–vector product leads to a perturbation that ruins the precision for the small eigenvalues. Numerical experiments (not shown here) suggest that when the matrix–vector product is almost exact the small eigenvalues can indeed be computed to high relative precision. This can be explained using similar arguments as in Section 3.

For the Lanczos method Paige [17] proved that in finite precision arithmetic the computed Lanczos vectors and tridiagonal matrix T_k satisfy a perturbed Lanczos relation. When applied to the normal equations this result reads

$$(\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{V}_k = \mathbf{V}_{k+1}\underline{T}_k + \widetilde{\mathbf{F}}_k \tag{3.1}$$

where, ignoring higher order terms,

$$\|\widetilde{\mathbf{f}}_j\| \leq \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}^{\mathrm{T}}\mathbf{A}\|, \tag{3.2}$$

and the constant, $\mathcal{O}(1)$, depends on the dimensions n and the number of nonzeros in $\mathbf{A}$. Using this relation we find for the approximation (1.3) that

$$\mathbf{A}^{\mathrm{T}}\mathbf{b} - (\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{x}_k^0 - \mathbf{V}_k(e_1 - \underline{T}_k T_k^{-1}e_1)\sqrt{\phi_0} = (\mathbf{A}^{\mathrm{T}}\mathbf{b} - \mathbf{V}_k e_1\sqrt{\phi_0}) - \widetilde{\mathbf{F}}_k T_k^{-1} e_1\sqrt{\phi_0}. \tag{3.3}$$

Following [21, 12, 2] we assume that the second term on the left is becoming much smaller than machine precision such that for k large enough, the residual for the normal equations is dominated by the term on the right. Hence, (2.3) can be achieved if we can show that

$$(\|\mathbf{b} - \mathbf{A}\mathbf{x}_k\| \approx) \quad \|\mathbf{A}^{-T}\left((\mathbf{A}^{\mathrm{T}}\mathbf{b} - \mathbf{V}_k e_1\sqrt{\phi_0}) - \widetilde{\mathbf{F}}_k T_k^{-1} e_1\sqrt{\phi_0}\right)\| \leq \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}\|\|\mathbf{x}\|. \tag{3.4}$$

In [7] a multi–shift CG method is presented based on the Lanczos method. This is not an "optimal choice" since for this Lanczos method (3.4) cannot be proved (see the estimate (3.2)). Therefore, the authors provide an alternative implementation of the three–term Lanczos method [7, Algorithm 6] without analysis. We discuss and compare this method to our implementation at the end of the next section.

We now consider if the CGLS recurrences can offer advantages in building up the basis as an alternative to the Lanczos method. Note that this does not follow automatically from the error analysis for CGLS in [12, 2]. The point is that these results for CGLS do not depend, for example, on rounding errors made in the update of the conjugate search direction. (Of course they influence the convergence speed.) When computing the Lanczos approximation (1.3) these errors become relevant and therefore these results are not sufficient.

Let $\mathbf{f}_j^c$ denote the perturbation in the computation of $\mathbf{c}_j$ caused by the use of computer arithmetic and assume similar notation for the other perturbations. Using standard rounding analysis e.g., [11, Section 2.4] we get, again by ignoring higher order terms,

$$\begin{aligned}
\|\mathbf{f}_j^z\| &\leq \varepsilon\,(\|\mathbf{z}_{j-1}\| + 2\|\alpha_{j-1}\mathbf{c}_{j-1}\|) && (3.5)\\
\|\mathbf{f}_j^c\| &\leq \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}\|\,\|\mathbf{p}_j\| && (3.6)\\
\|\mathbf{f}_j^r\| &\leq \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}^{\mathrm{T}}\|\,\|\mathbf{z}_j\| && (3.7)\\
\|\mathbf{f}_j^p\| &\leq \varepsilon\left(\|\mathbf{r}_j\| + 2\|\beta_{j-1}\mathbf{p}_{j-1}\|\right) \leq \varepsilon\left(3\|\mathbf{r}_j\| + 2\|\mathbf{p}_j\|\right). && (3.8)
\end{aligned}$$

The unspecified constants in (3.6) and (3.7) depend on the number of nonzeros per row in $\mathbf{A}$ and $\mathbf{A}^{\mathrm{T}}$ respectively. With the notation from the previous section we find the following matrix relations for CGLS

$$\mathbf{Z}_{k+1}\underline{J}_k = \mathbf{C}_k\Delta_k + \mathbf{F}_k^z, \quad \mathbf{C}_k = \mathbf{A}\mathbf{P}_k + \mathbf{F}_k^c, \quad \mathbf{R}_{k+1} = \mathbf{A}^{\mathrm{T}}\mathbf{Z}_{k+1} + \mathbf{F}_{k+1}^r, \quad \mathbf{P}_k U_k = \mathbf{R}_k + \mathbf{F}_k^p.$$

Substitution yields the perturbed Lanczos relation (3.1) where the perturbation is given by

$$\widetilde{\mathbf{F}}_k \equiv -(\mathbf{A}^{\mathrm{T}}\mathbf{A})\widetilde{\mathbf{F}}_k^p - \mathbf{A}^{\mathrm{T}}(\widetilde{\mathbf{F}}_k^c + \widetilde{\mathbf{F}}_k^z\Delta_k^{-1})L_k - \widetilde{\mathbf{F}}_{k+1}^r\underline{T}_k, \tag{3.9}$$

with $\widetilde{\mathbf{F}}_k^p \equiv \mathbf{F}_k^p \Phi_k^{-1}$, $\widetilde{\mathbf{F}}_k^c \equiv \mathbf{F}_k^c \Phi_k^{-1}$, $\widetilde{\mathbf{F}}_k^z \equiv \mathbf{F}_k^z \Phi_k^{-1}$, $\widetilde{\mathbf{F}}_k^r \equiv \mathbf{F}_k^r \Phi_k^{-1}$. Plugging everything into (3.4) we see that it is sufficient if we bound each of the following terms on $\varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}\|\|\mathbf{x}\|$:

$$\|\mathbf{A}^{-T}\widetilde{\mathbf{F}}_{k+1}^r(e_1 - \underline{T}_k T_k^{-1} e_1)\sqrt{\phi_0}\|, \quad \|\mathbf{A}\widetilde{\mathbf{F}}_k^p T_k^{-1} e_1 \sqrt{\phi_0}\|, \quad \|(\widetilde{\mathbf{F}}_k^c + \widetilde{\mathbf{F}}_k^z \Delta_k^{-1}) L_k T_k^{-1} e_1 \sqrt{\phi_0}\|.$$

Greenbaum [12] and Björck et al. [2] analyze CGLS by studying *the residual gap* $\mathbf{z}_j - (\mathbf{b} - \mathbf{A}\mathbf{x}_j)$ for the least squares problem. This means that they take into account the rounding errors that are collected in $\mathbf{f}_j^z$ and $\mathbf{f}_j^c$. Since they argue that (2.3) holds for CGLS we do not further study the effect of these perturbations and refer to these papers. Instead we only look at the effect of the perturbations $\widetilde{\mathbf{F}}_k^r$ and $\widetilde{\mathbf{F}}_k^p$.

A simple calculation shows that in exact arithmetic

$$\|(e_1 - \underline{T}_k T_k^{-1} e_1)\sqrt{\phi_0}\| = \sqrt{\phi_0}\prod_{i=0}^{k}\sqrt{\beta_i} = \sqrt{\phi_k}.$$

This equality still holds to relative machine precision when the division for computing β_j in (2.5) or the square root[2] is done in computer arithmetic. Therefore we have, ignoring higher order terms,

$$\|\mathbf{A}^{-T}\widetilde{\mathbf{F}}_{k+1}^r(e_1 - \underline{T}_k T_k^{-1} e_1)\sqrt{\phi_0}\| \le \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}^{-T}\|\|\mathbf{A}^{\mathrm{T}}\|\|\mathbf{z}_k\|.$$

Since we assume that $\|\mathbf{r}_k\|$ becomes orders of magnitude smaller than machine precision, the $\|\mathbf{z}_k\|$ have the same property and this term is therefore harmless.

It is a little more challenging to show that

$$\|\mathbf{A}\widetilde{\mathbf{F}}_k^p T_k^{-1} e_1 \sqrt{\phi_0}\| \le \varepsilon\,\mathcal{O}(1)\,\|\mathbf{A}\|\|\mathbf{x}\|, \tag{3.10}$$

or, using (3.8) and that $\sqrt{\phi_j}$ equals $\|\mathbf{r}_j\|$ to machine precision [11, Section 2.4.5] and subsequently ignoring higher order terms:

$$\left(3 + 2\frac{\|\mathbf{p}_j\|}{\|\mathbf{r}_j\|}\right)|e_{j+1}^{\mathrm{T}} T_k^{-1} e_1 \sqrt{\phi_0}| \le \mathcal{O}(1)\,\|\mathbf{x}\|. \tag{3.11}$$

We will show (3.11) by assuming that the occurring quantities are computed in an exact CGLS process. Using [14, Theorem 5.3] we have that $\|\mathbf{p}_j\|/\|\mathbf{r}_j\| = \|\mathbf{r}_j\|/\rho_j$ with $\rho_j \equiv (\sum_{i=0}^{j}\|\mathbf{r}_i\|^{-2})^{-1/2}$. The value ρ_j is essentially the norm of the *minimal residual* approximation from the subspace corresponding to the approximation $\mathbf{x}_j^{\mathrm{MR}}$, thus $\rho_j = \|\mathbf{A}^{\mathrm{T}}\mathbf{b} - (\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{x}_j^{\mathrm{MR}}\|$. The obvious estimates $|e_{j+1}^{\mathrm{T}} T_k^{-1} e_1| \le \|\mathbf{V}_k T_k^{-1} e_1\| = \|\mathbf{x}_k\|$ and $\|\mathbf{r}_j\| \le \|\mathbf{A}\|\|\mathbf{A}^{-1}\|\,\rho_j$ lead to the crude bound

$$(3 + 2\|\mathbf{r}_j\|/\rho_j)|e_{j+1}^{\mathrm{T}} T_k^{-1} e_1 \sqrt{\phi_0}| \le (3 + 2\|\mathbf{A}\|\|\mathbf{A}^{-1}\|)\|\mathbf{x}_k\|.$$

This is not sufficient for proving (3.11). However, we can show that large perturbations are canceled against smaller elements in the vector $T_k^{-1}e_1$. To see this we first write

$$\|\mathbf{r}_j\|\,|e_{j+1}^{\mathrm{T}} T_k^{-1} e_1|\sqrt{\phi_0} = |\mathbf{r}_j^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{r}_0|.$$

Then, we note that

$$|\mathbf{r}_j^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{A})^{-1}\mathbf{r}_0| = |(\mathbf{x} - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{A})\mathbf{x}| = |(\mathbf{x} - \mathbf{x}_j)^{\mathrm{T}}(\mathbf{A}^{\mathrm{T}}\mathbf{A})(\mathbf{x} - \mathbf{x}_j^{\mathrm{MR}})| \le \|\mathbf{x} - \mathbf{x}_j\|\rho_j \le \|\mathbf{x}\|\rho_j.$$

[2]The square roots are assumed exact since we disregard rounding errors in the transformation from (2.6) to (2.7).

Here, we used that $(\mathbf{A}^{\mathrm{T}}\mathbf{A})(\mathbf{x}-\mathbf{x}_j) = \mathbf{r}_j \perp \mathbf{x}_j^{\mathrm{MR}}$ and the fact that the errors for the CG method are also in 2-norm monotonically decreasing [14, Theorem 6.3]. Combining these expressions for all j we can prove (3.10).

We have simplified our analysis by using relations that are not necessarily preserved in the finite precision context. (This is not uncommon.) In a recent paper [22] Strakoš and Tichý give a detailed error analysis of the conjugate gradient method. Using some of their results it can be shown that many of the used identities are still valid (up to a constant) in the finite precision context and a more refined analysis than presented here is possible. However, the observations in this section coincide with our numerical experience and a detailed analysis is beyond the scope of this paper.

In this section we have collected evidence that constructing Lanczos approximations to the least squares problem build “on top of” the CGLS method can lead to accurate solutions for the least squares problem. Applying the standard Lanczos method to the normal equations can lead to a perturbed Lanczos relation as in (3.1) with a normwise smaller perturbation (since it is possible that $\|\mathbf{p}_j\|/\|\mathbf{r}_j\| \gg 1$). However, for the alternative Lanczos method the perturbation and $T_k^{-1}e_1$ have a desirable structure that allows much more accurate solutions.

Freund et al. present in [6] a version of the QMR method based on coupled two–term recurrences. In the QMR method there is a clear separation between the Lanczos part and the solution part as there is also for the multi–shift CG methods. They observe that, when compared to the three–term recurrence version of QMR, the difference is typically not very large [6, Section 9] for most problems. Similarly, we do not expect and see large differences between the various multi–shift CG implementations. However, the structure of the perturbation $\widetilde{\mathbf{F}}_k$ can become relevant for some problems as seen in this section.

We conclude this section by remarking that an alternative for the multi–shift CG method for regularized systems can be obtained based on *Lanczos bidiagonalization* e.g., [11, Section 9.3.3]. With similar arguments as given here, the accuracy of this method can be understood.

4 Solving the shifted system

In efficient implementations of the multi–shift conjugate gradient method it is required that the vectors $\mathbf{x}_k^\tau$ are constructed at the same time the Krylov subspace is build up in order to circumvent that we have to store all Lanczos vectors $\mathbf{v}_j$. In case the CG recurrences (2.1) or CGLS recurrences (2.4) are used as Lanczos process, then it is clear from (2.7) that also the $L^{\mathrm{T}}DL$ factorization of T_k is directly available. The so–called *quotient–difference* algorithms introduced by Rutishauser [18] provide a means to construct an $L^{\mathrm{T}}DL$ factorization of the shifted matrix $T_k + \tau I$ directly from the factors of T_k. These algorithms construct the factors D_k^τ and L_k^τ in a step–by–step fashion such that

$$L_k^{\mathrm{T}}\Delta_k^{-1}L_k + \tau I = (L_k^\tau)^{\mathrm{T}} D_k^\tau L_k^\tau,$$

where D_k^τ is diagonal with diagonal elements $d_0, \ldots, d_{k-1}$ and L_k^τ is upper bidiagonal with diagonal elements one and upper diagonal elements $l_0, \ldots, l_{k-2}$. If we take the *differential form* of the *stationary qd transformation* (dstqds) presented in [4, Algorithm 4.2], then we have, with $t_0 = \tau$, the following recurrence relations for computing the elements of D_k^τ and

L_k^τ

$$d_{j-1} = t_{j-1} + \alpha_{j-1}^{-1}, \quad l_{j-1} = -\frac{\sqrt{\beta_{j-1}}}{\alpha_{j-1}d_{j-1}}, \quad t_j = -l_{j-1}\sqrt{\beta_{j-1}}t_{j-1} + \tau. \tag{4.1}$$

Just as for the conjugate gradient method, the construction of the vector $\mathbf{x}_k^\tau$ can be done efficiently by introducing the auxiliary vectors $\mathbf{p}_j^\tau$ defined by the relation $\mathbf{P}_k^\tau = \mathbf{V}_k(L_k^\tau)^{-1}$. Starting with $\xi_0 = 1$, $\mathbf{p}_0^\tau = \mathbf{r}_0/\sqrt{\phi_0}$, $\mathbf{x}_0^\tau = \mathbf{0}$ this leads to

$$\mathbf{x}_j^\tau = \mathbf{x}_{j-1}^\tau + \frac{\xi_{j-1}}{d_{j-1}}\mathbf{p}_{j-1}^\tau, \quad \mathbf{p}_j^\tau = \mathbf{r}_j/\sqrt{\phi_j} - l_{j-1}\mathbf{p}_{j-1}^\tau, \quad \xi_j = -\xi_{j-1}l_{j-1}. \tag{4.2}$$

Notice that the norm the residual of the shifted system is given by the product of the off-diagonal elements of L_k^τ, which equals the temporary variable ξ_j in (4.2). Dhillon and Parlett present [4, Section 4.3] a roundoff error analysis of the dstqds algorithm that shows that the outcome of this algorithm is relatively close to the exact result when applied to factors relatively close to the original input. Our implementation of the multi–shift CG or CGLS method consists of adding (4.1) and (4.2) to the ordinary CG and CGLS methods.

We will now summarize an approach that is used at several places in literature in multi–shift versions of the (Bi-)CG method based on coupled two–term recurrences [15, 9]. For more details consult these references. An important observation is that the residuals for shifted systems are *colinear* for the CG method, that is, there exist constants γ_j such that $\mathbf{r}_j^\tau = \mathbf{r}_j/\gamma_j$. Writing out the three–term recurrence of the residuals $\mathbf{r}_k^\tau$ (similar to (2.6)) and comparing terms reveals, with $\gamma_{-1} = \gamma_0 = 1$, the three–term relation

$$\gamma_j = (1 + \alpha_{j-1}\tau)\gamma_{j-1} + \frac{\alpha_{j-1}}{\alpha_{j-2}}\beta_{j-2}(\gamma_{j-1} - \gamma_{j-2}), \tag{4.3}$$

and the recurrences for the iterates and search directions are given by

$$\mathbf{x}_j^\tau = \mathbf{x}_{j-1}^\tau + \alpha_{j-1}\left(\frac{\gamma_{j-1}}{\gamma_j}\right)\mathbf{p}_{j-1}^\tau, \quad \mathbf{p}_j^\tau = \mathbf{r}_j/\gamma_j + \beta_{j-1}\left(\frac{\gamma_{j-1}}{\gamma_j}\right)^2\mathbf{p}_{j-1}^\tau, \tag{4.4}$$

with initially $\mathbf{p}_0^\tau = \mathbf{r}_0$ and $\mathbf{x}_0^\tau = \mathbf{0}$.

If a stable method is applied for computing (1.3), then this leads to an approximate solution that is usually sufficiently accurate when dealing with ordinary linear systems. However, this might not be the case for normal equations, since then a dependence of the attainable precision on the square of the condition number of $\mathbf{A}$ may have been introduced. Therefore, a point of concern of the approach (4.3)–(4.4) is that it implicitly forms the ill–conditioned tridiagonal matrix S_k (cf., (2.6)) in the computation of the γ_j in (4.3) whereas the qd-method directly transforms the factorization of the unshifted problem to that of the shifted problem without forming the tridiagonal matrix.

Numerical experiments suggest that (4.4) is often remarkably accurate and in most cases as accurate as (4.2). Nevertheless, there are examples where differences are clear. We show this for two simple regularized systems. The matrix $\mathbf{A}$ has eigenvalues $\{1/250, 240, 241, \ldots, 250\}$ ($n = 12$) and the orthonormal eigenvector basis is random. The right-hand-side has equal components in all eigenvector directions except for the direction corresponding to the smallest eigenvalue, there the component is 250^2 times as large. The results for solving the system (1.4) are presented in Figure 1 for $\tau = 10^{-8}$ and $\tau = 1$. In this picture we have also presented the results for the CGLS method when applied to the regularized problems directly. An algorithm

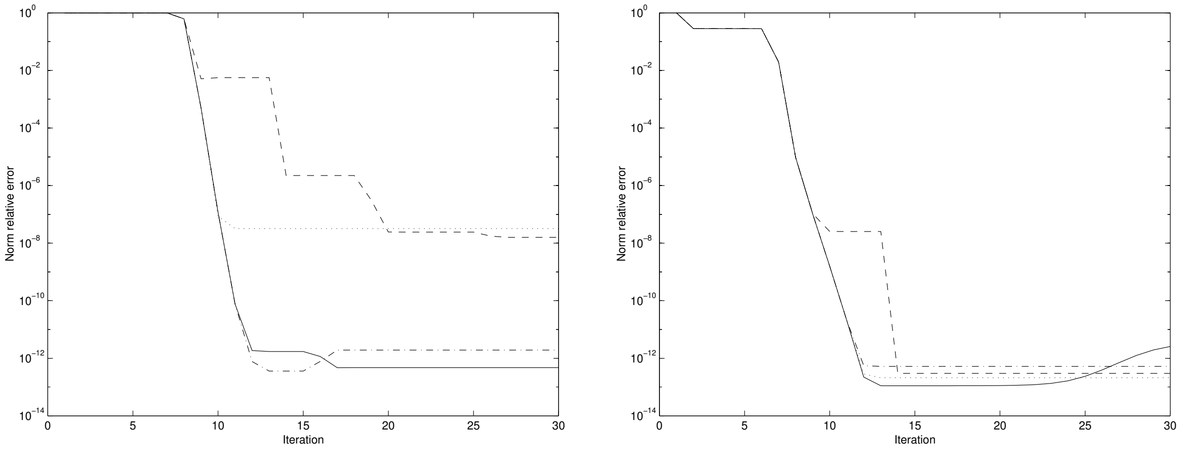

Figure 1: Relative error as function of k for the CGLS method (solid), Algorithm 6 from [7] (dashed), multi–shift CGLS with (4.2) (dash-dot) and (4.4) (dotted) for two different shifts: $\tau = 10^{-8}$ (left) and $\tau = 1$ (right).

for this is given in, e.g., [7, Algorithm 3]. In this case (4.2) clearly gives more accurate results than (4.4) for $\tau = 10^{-8}$. For larger values of τ and very small values the difference becomes smaller. For these results and the ones in the next section, we used Matlab.

The multi–shift CG method presented in [7, Alg. 6] uses a variant of the standard Lanczos method. This implementation uses a three–term recurrence for the least squares residuals and results in a tridiagonal matrix in standard form. So, even if the Lanczos part is more accurate due to these changes, the accuracy is expected to be limited by the inversion of the tridiagonal. The dashed lines in Figure 1 show this.

5 Numerical experiments

In this section we compare the attainable precision of the CGLS method (for regularized systems) to the approximations from our version of the multi–shift CGLS method (MCGLS), that is the CGLS method combined with (4.2) to solve the additional shifted system. The "exact" solution, $\mathbf{x}^\tau$, was computed using a singular value decomposition and we report the relative error given by

$$\|\mathbf{x}_k^\tau - \mathbf{x}^\tau\|/\|\mathbf{x}^\tau\|,$$

where $\mathbf{x}_k^\tau$ is the computed approximation with either method. The number of iterations, k, was chosen such that the error for the particular method was minimal. The results for various test problems from [13] are given in Table 1.

The results in this table confirm that this multi–shift CG method achieves a comparable accuracy to applying the CGLS method directly to the regularized system. However, there are a few interesting differences that occur now and then. One aspect of the CGLS method for regularized systems is that for large shifts the method tends to diverge after reaching its maximal precision. An interesting observation is that the multi–shift version of CGLS does not have this behavior. This is illustrated in the left figure in Figure 2. The computational costs per step are much lower for the multi–shift version of CGLS (no matrix-vector multiplication, no inner products, less vector updates for solving the shifted problems) than

τ	10^{-8}	10^{-4}	1	10^{4}
		HEAT(100)		
CGLS	4.9(-13)	5.1(-15)	6.6(-16)	6.5(-16)
MCGLS	4.8(-13)	5.2(-15)	6.1(-16)	7.0(-16)
		URSELL(100)		
CGLS	6.7(-14)	2.9(-15)	2.9(-16)	2.6(-16)
MCGLS	8.7(-14)	3.3(-15)	2.5(-16)	2.7(-16)
		FOXGOOD(100)		
CGLS	2.2(-13)	3.0(-15)	3.7(-16)	6.7(-16)
MCGLS	2.7(-13)	3.0(-15)	3.7(-16)	7.3(-16)
		ILAPLACE(100)		
CGLS	9.2(-13)	1.8(-14)	1.2(-15)	6.7(-16)
MCGLS	8.4(-13)	1.8(-14)	1.3(-15)	6.0(-16)

Table 1: Attainable relative errors for various problems and various choices for τ.

the direct application of CGLS. However, it is remarkable that, in addition, the multi–shift version sometimes needs less iteration steps. An example of this is given in the right picture in Figure 2.

Acknowledgment. The research of J. van den Eshof was financially supported by the Dutch scientific organization (NWO) through project 613.002.035.

References

[1] Zhaojun Bai and Roland W. Freund, *A symmetric band Lanczos process based on coupled recurrences and some applications*, SIAM J. Sci. Comput. **23** (2001), no. 2, 542–562 (electronic), Copper Mountain Conference (2000). MR 1 861 264

[2] Åke Björck, Tommy Elfving, and Zdeněk Strakoš, *Stability of conjugate gradient and Lanczos methods for linear least squares problems*, SIAM J. Matrix Anal. Appl. **19** (1998), no. 3, 720–736 (electronic). MR 99a:65051

[3] Biswa Nath Datta and Youcef Saad, *Arnoldi methods for large Sylvester-like observer matrix equations, and an associated algorithm for partial spectrum assignment*, Linear Algebra Appl. **154/156** (1991), 225–244. MR 92b:65032

[4] Inderjit Dhillon and Beresford Parlett, *Orthogonal eigenvectors and relative gaps*, Submitted.

[5] Roland W. Freund, *Solution of shifted linear systems by quasi-minimal residual iterations*, Numerical linear algebra (Kent, OH, 1992), de Gruyter, Berlin, 1993, pp. 101–121. MR 1 244 155

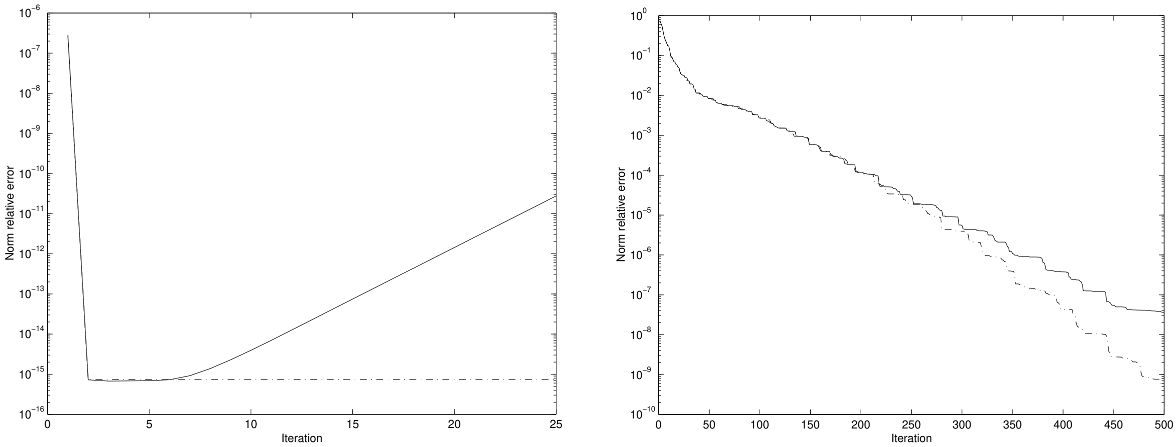

Figure 2: The relative error (as a function of k) of the CGLS method (solid) and multi–shift CGLS using (4.2) (dash-dot). Left: problem FOXGOOD(100) with $\tau = 10^4$. Right: HEAT(100) with $\tau = 10^{-8}$.

[6] Roland W. Freund and Noël M. Nachtigal, *An implementation of the QMR method based on coupled two-term recurrences*, SIAM J. Sci. Comput. **15** (1994), no. 2, 313–337, Iterative methods in numerical linear algebra (Copper Mountain Resort, CO, 1992). MR 95f:65067

[7] A. Frommer and P. Maass, *Fast CG-based methods for Tikhonov-Phillips regularization*, SIAM J. Sc. Comput. **20** (1999), 1831–1850.

[8] A. Frommer, B. Nöckel, S. Güsken, Th. Lippert, and K. Schilling, *Many masses on one stroke: Economic computation of quark*, Int. J. Modern Physics **C 6** (1995), 627–638.

[9] Andreas Frommer, *BICGSTAB(l) for families of shifted linear systems*, Preprint BUGHW-SC 02/04, Bergische Universität GH Wuppertal, Wuppertal, Germany, November 2002.

[10] Andreas Frommer and Uwe Glässner, *Restarted GMRES for shifted linear systems*, SIAM J. Sci. Comput. **19** (1998), no. 1, 15–26 (electronic), Special issue on iterative methods (Copper Mountain, CO, 1996). MR 99b:65033

[11] G. H. Golub and C. F. Van Loan, *Matrix computations*, 3rd ed., The John Hopkins University Press, Baltimore, London, 1996.

[12] Anne Greenbaum, *Estimating the attainable accuracy of recursively computed residual methods*, SIAM J. Matrix Anal. Appl. **18** (1997), no. 3, 535–551. MR 98c:65048

[13] Per Christian Hansen, *Regularization tools: a Matlab package for analysis and solution of discrete ill-posed problems*, Numer. Algorithms **6** (1994), no. 1-2, 1–35. MR 94k:65062

[14] Magnus R. Hestenes and Eduard Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards **49** (1952), 409–436 (1953). MR 15:651a

[15] B. Jegerlehner, *Krylov space solvers for shifted linear systems*, HEP-LAT hep-lat/9612014, 1996.

[16] H. Neuberger, *Overlap Dirac operator*, Numerical challenges in Lattice Quantum Chromodynamics (Berlin) (A. Frommer, Th. Lippert, B. Medeke, and K. Schilling, eds.), Springer-Verlag, 2000.

[17] C. C. Paige, *Error analysis of the Lanczos algorithm for tridiagonalizing a symmetric matrix*, J. Inst. Math. Appl. **18** (1976), no. 3, 341–349. MR 58:19082

[18] Heinz Rutishauser, *Der Quotienten-Differenzen-Algorithmus*, Mitt. Inst. Angew. Math. Zürich **1957** (1957), no. 7, 74. MR 19,686b

[19] V. Simoncini and F. Perotti, *On the numerical solution of* $(\lambda^2 A + \lambda B + C)x = b$ *and application to structural dynamics*, SIAM J. Sci. Comput. **23** (2002), no. 6, 1875–1897 (electronic). MR 1 923 717

[20] Valeria Simoncini, *Restarted full orthogonalization method for shifted linear systems*, Tech. report, 2002, To appear in BIT.

[21] Gerard L. G. Sleijpen, Henk A. van der Vorst, and Diederik R. Fokkema, *BiCGstab*(ℓ) *and other hybrid Bi-CG methods*, Numer. Algorithms **7** (1994), no. 1, 75–109. MR 95d:65030

[22] Zdeněk Strakoš and Petr Tichý, *On error estimation in the conjugate gradient method and why it works in finite precision computations*, ETNA **13** (2002), 56–80.