ENRICHED KRYLOV SUBSPACE METHODS FOR ILL-POSED PROBLEMS

D. CALVETTI*, L. REICHEL†, AND A. SHUIBI‡

Abstract. We describe a modification of the conjugate gradient method for the normal equations (CGNR) that allows us to enrich the Krylov subspaces, in which the iterates are determined, with vectors containing pertinent information about the desired solution. The enriched CGNR method easily can be adapted to the solution of linear systems arising from penalized least-squares problems and Tikhonov regularization. Applications to the solution of linear discrete ill-posed problems illustrate that enrichment of the Krylov subspaces can improve the quality of the computed approximate solutions and reduce the computational effort required for their determination.

1. Introduction. This paper presents a modification of the Conjugate Gradient (CG) method for the solution of the normal equations

$$A^T A x = A^T b^\delta \tag{1.1}$$

associated with the linear system of equations

$$Ax = b^\delta, \tag{1.2}$$

where $A \in \mathbb{R}^{m\times n}$, $x \in \mathbb{R}^n$ and $b^\delta \in \mathbb{R}^m$. Our modification makes it possible to enrich the Krylov subspaces, in which the computed approximate solutions live, with available information about the desired solution. The Krylov subspaces are enriched by adding a subspace that makes it possible to model certain known important properties of the solution. Several computed examples at the end of the paper illustrate how enrichment can increase the accuracy in the computed approximate solutions and reduce the computational work.

Throughout this paper, we assume that m and n are so large that factorization of the matrix A is undesirable or unfeasible. We are therefore concerned with solution of (1.1) by an iterative method. We are particularly interested in the computation of an approximate solution of (1.2) when the matrix A is of ill-determined rank and the right-hand side b^δ is contaminated by an error η. Linear systems of this kind arise when discretizing ill-posed problems, such as Fredholm integral equations of the first kind with a smooth kernel, and are commonly referred to as linear discrete ill-posed problems. Note that when (1.2) is a linear discrete ill-posed problems, so is (1.1). The error η in the right-hand side may stem from measurement or discretization errors, and is sometimes referred to as noise.

A matrix of ill-determined rank has many "tiny" singular values of different orders of magnitude; some singular values may vanish. Discrete ill-posed problems of the form (1.2) might not be consistent; however, we note that the associated normal equations (1.1) are consistent even if (1.2) is not.

The CG method is one of the most popular iterative methods for the solution of large linear systems of equations with a symmetric positive definite matrix. It can also be applied to the iterative solution of consistent linear systems of equations with

*Department of Mathematics, Case Western Reserve University, Cleveland, OH 44106. E-mail: dxc57@po.cwru.edu. Research supported in part by NSF grant DMS-0107841.

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242. E-mail: reichel@math.kent.edu. Research supported in part by NSF grant DMS-0107858.

‡Department of Mathematical Sciences, Kent State University, Kent, OH 44242. E-mail: ashuibi@math.kent.edu. Research supported in part by NSF grant DMS-0107858.

a positive semidefinite matrix. Therefore it can be applied to the solution of linear discrete ill-posed problems of the form (1.1). The CGNR method is an implementation of the CG method applied to the normal equations, that does not require the matrix A^TA to be formed. Instead, each iteration requires two matrix-vector product evaluations, one with the matrix A and one with A^T. The CGNR method is discussed, e.g., by Saad [13].

Let the matrix A be of ill-determined rank, and let $b \in \mathbb{R}^m$ denote the error-free right-hand side associated with the available right-hand side b^δ, i.e.,

$$b^\delta = b + \eta. \tag{1.3}$$

We assume that b is in the range of A and that the norm of the error

$$\delta := \|\eta\| \tag{1.4}$$

is explicitly known, but that the error η is not. Here and throughout this paper $\|\cdot\|$ denotes the Euclidean vector norm.

We would like to determine the solution x_* of minimal Euclidean norm of the error-free linear system of equations

$$Ax = b. \tag{1.5}$$

Since b is not available, we seek to compute an approximate solution of (1.1) that is a good approximation of x_*. By assumption A is of ill-determined rank, and therefore severely ill-conditioned. The minimal-norm least-squares solution of (1.2) typically is very sensitive to the error η in b^δ and, generally, is not an acceptable approximation of x_*.

A popular approach to determining an approximation of x_* when the matrix A is large is to apply suitably many (or few) steps of the CGNR method to (1.1). Early termination of the iterations by the CGNR method means that a system of equations that is less sensitive than (1.1) to the error η in b^δ is solved. The replacement of a linear discrete ill-posed problem by a linear system of equations with a less ill-conditioned matrix is commonly referred to as regularization; see Hanke and Hansen [4, 5, 9] for theoretical and practical issues in connection with the application of the CGNR method to the solution of linear discrete ill-posed problems.

Let $x_0 := 0$ be the initial approximate solution of (1.1), and let x_k, $k = 1, 2, 3, \ldots$, denote the sequence of approximate solutions determined by the CGNR method. The quantity

$$d_k := b^\delta - Ax_k \tag{1.6}$$

is known as the discrepancy associated with the approximate solution x_k. Hestenes and Stiefel [10] showed that

$$\begin{aligned} \|d_{k+1}\| &\leq \|d_k\|, \\ \|x_{k+1}\| &\geq \|x_k\|, \end{aligned} \tag{1.7}$$

for $k = 0, 1, 2, \ldots$.

The discrepancy principle suggests that the iterations by the CGNR method be terminated as soon as an approximate solution x_k has been determined, such that the associated discrepancy satisfies

$$\|d_k\| \leq \delta. \tag{1.8}$$

We will use this termination criterion in the present paper. Let k^δ denote the smallest index k, such that the inequality (1.8) is satisfied. Then x_{k^δ} is our computed approximation of x_*. Generally, k_δ increases as δ converges to zero. It can be shown that $\lim_{\delta\searrow 0} x_{k^\delta} = x_*$; see, e.g., Hanke [4] and Hansen [9, Chapter 7] for recent discussions on this stopping criterion.

When the error η in the right-hand side b^δ is of large norm, the termination criterion (1.8) may only allow a few iterations to be carried out by the CGNR method, and features of interest in the solution x_* of (1.5) might not be resolved adequately by the computed approximate solution x_{k^δ}. We therefore propose to enrich the Krylov subspaces in which the iterates x_k live by a linear space that allows certain known desirable features of x_* to be represented by the iterates already for small values of k. An algorithm for the enriched CGNR method is presented in Section 2.

Tikhonov regularization is possibly the most popular approach to replace a linear discrete ill-posed problem (1.2) by a linear system of equations that is less sensitive to the error in the right-hand side. Specifically, Tikhonov proposed to replace the solution of (1.2) by the solution of the penalized least-squares problem

$$\min_{x\in\mathbb{R}^n} (\|Ax - b^\delta\|^2 + \mu\|x\|^2). \tag{1.9}$$

The parameter $\mu \geq 0$ is known as the regularization parameter. It determines how close the solution x_μ of (1.9) is to the solution x_* of (1.5), and how sensitive x_μ is to the error η in b^δ. The term $\mu\|x\|^2$ penalizes the growth of the computed solution, thereby preventing the propagated error due to the error η in b^δ from dominating the solution.

For any fixed value of $\mu > 0$, the solution x_μ of (1.9) satisfies the linear system of equations

$$(A^T A + \mu I)x = A^T b^\delta. \tag{1.10}$$

This system can be solved by the CG method, without explicitly forming the matrix $A^T A + \mu I$. The kth iterate, $x_k^{(\mu)}$, determined in this manner, with initial approximate solution $x_0 = 0$ lives in the Krylov subspace

$$\mathcal{K}_k(A^T A + \mu I, A^T b^\delta) := \mathrm{span}\{A^T b^\delta, (A^T A + \mu I)A^T b^\delta, \ldots, (A^T A + \mu I)^{k-1} A^T b^\delta\}.$$

Note that this subspace is independent of $\mu \geq 0$, i.e.,

$$\mathcal{K}_k(A^T A + \mu I, A^T b^\delta) = \mathcal{K}_k(A^T A, A^T b^\delta).$$

It follows that $x_k^{(\mu)}$ and the iterate x_k determined by the CGNR method when applied to (1.1) with $x_0 := 0$ live in the same Krylov subspace. It is therefore natural to enrich the CG method for the solution of (1.10) in the same manner as we enrich the CGNR method. An enriched iterative method of CG-type for the solution of (1.10) is described in Section 3. A few computed examples that illustrate the numerical performance of the methods discussed in Sections 2 and 3 are presented in Section 4. The latter section also outlines some extensions.

2. An enriched CGNR method. Throughout this paper we assume that the initial approximate solution x_0 is the zero vector. We first review the CGNR method and then discuss a modification that yield the enriched CGNR method. The kth iterate, x_k, determined by the CGNR method applied to (1.1) belongs to the Krylov subspace $\mathcal{K}_k(A^TA, A^Tb^\delta)$ and is characterized by

$$\Phi(x_k) = \min_{x \in \mathcal{K}_k(A^TA, A^Tb^\delta)} \Phi(x),$$

where

$$\Phi(x) := \frac{1}{2} x^T A^T A x - x^T A^T b^\delta. \tag{2.1}$$

The minimizer x_k is determined by carrying out a sequence of linear searches along A^TA-conjugate search directions $p_0, p_1, \ldots, p_{k-1}$ that span $\mathcal{K}_k(A^TA, A^Tb^\delta)$ and are computed during the iterations. Specifically, the iterate x_k is determined from the previous iterate x_{k-1} and p_{k-1} according to

$$x_k := x_{k-1} + \alpha_{k-1} p_{k-1},$$

where $\alpha_{k-1} \in \mathbb{R}$ is the solution of the minimization problem

$$\min_{\alpha \in \mathbb{R}} \Phi(x_{k-1} + \alpha p_{k-1}). \tag{2.2}$$

Introduce the residual vector for the normal equations (1.1) associated with the iterate x_k,

$$r_k := A^T b^\delta - A^T A x_k.$$

The search direction p_k is computed from r_k and the previous search direction according to

$$p_k := r_k + \beta_{k-1} p_{k-1},$$

where $\beta_{k-1} \in \mathbb{R}$ is chosen so that p_k is A^TA-conjugate to all previously generated search directions. We remark that during the kth step of the CGNR method, the discrepancy d_k is evaluated, and then the residual r_k is computed according to $r_k := A^T d_k$. The availability of d_k makes it easy to implement the stopping criterion (1.8).

We turn to the enriched CGNR method. Let

$$\mathcal{Q} := \text{span}\{q^{(1)}, q^{(2)}, \ldots, q^{(\ell)}\} \subset \mathbb{R}^n$$

be an ℓ-dimensional vector space. The enriched CGNR method determines the kth iterate $\tilde{x}_k$ in the subspace $\mathcal{K}_k(A^TA, A^Tb^\delta) \cup \mathcal{Q}$. The performance of the enriched CGNR method, when compared with the (standard) CGNR method, depends on to what extent the vector space $\mathcal{Q}$ represents pertinent information about the solution x_* that is not represented by the Krylov subspaces $\mathcal{K}_k(A^TA, A^Tb^\delta)$ for small to moderate values of k. Simple extensions of the Krylov subspaces $\mathcal{K}_k(A^TA, A^Tb^\delta)$ may already give a substantial reduction in the number of iterations required to satisfy the chosen stopping criterion, such as (1.8). For instance, numerical examples in Section 4 show that for some problems good results can be achieved for $\mathcal{Q} := \text{span}\{b^\delta\}$.

Let the vectors $\{\tilde{q}_k^{(1)}, \tilde{q}_k^{(2)}, \ldots, \tilde{q}_k^{(\ell)}\}$ span $\mathcal{Q} \backslash \mathcal{K}_k(A^TA, A^Tb^\delta)$ and assume that they are A^TA-conjugate to the search directions $p_0, p_1, \ldots, p_{k-1}$, where we define $\tilde{q}_0^{(j)} := q^{(j)}$ for $1 \le j \le \ell$ and $\mathcal{K}_0(A^TA, A^Tb^\delta) = \emptyset$. For $k = 1, 2, 3, \ldots$, we compute

$$\tilde{q}_k^{(j)} := \tilde{q}_{k-1}^{(j)} - \gamma_{k-1}^{(j)} p_{k-1}, \qquad 1 \le j \le \ell,$$

where the coefficients $\gamma_{k-1}^{(j)}$ are chosen so that the $\tilde{q}_k^{(j)}$, $1 \leq j \leq \ell$, are A^TA-conjugate to p_{k-1}. Then the $\tilde{q}_k^{(j)}$, $1 \leq j \leq \ell$, also are A^TA-conjugate to $p_0, p_1, \ldots, p_{k-1}$.

Introduce the matrix

$$\tilde{Q}_k = [\tilde{q}_k^{(1)}, \tilde{q}_k^{(2)}, \ldots, \tilde{q}_k^{(\ell)}] \in \mathbb{R}^{n \times \ell}$$

and define the kth iterate determined by the enriched CGNR method by

$$\tilde{x}_k := x_k + \tilde{Q}_k \tilde{y}_k = x_{k-1} + \alpha_{k-1} p_{k-1} + \tilde{Q}_k \tilde{y}_k, \tag{2.3}$$

where $\tilde{y}_k \in \mathbb{R}^\ell$ solves the minimization problem

$$\min_{\tilde{y} \in \mathbb{R}^\ell} \Phi(x_k + \tilde{Q}_k \tilde{y}). \tag{2.4}$$

The vector $\tilde{y}_k$ can be computed as the solution of the least-squares problem

$$\min_{\tilde{y} \in \mathbb{R}^n} \|A\tilde{Q}_k \tilde{y} - d_k\|. \tag{2.5}$$

We remark that the A^TA-conjugacy of the columns of $\tilde{Q}_k$ to p_{k-1} yields that $\tilde{Q}_k$ is a rank-one modification of $\tilde{Q}_{k-1}$, i.e.,

$$\tilde{Q}_k := \tilde{Q}_{k-1} - p_{k-1} g_{k-1}^T,$$

where

$$g_{k-1} := \frac{(A^TA\tilde{Q}_{k-1})^T p_{k-1}}{p_{k-1}^T A^T A p_{k-1}}.$$

Proposition 2.1. *Let the iterates $\tilde{x}_k$, $k = 1, 2, 3, \ldots$, be determined by (2.3). Then the associated discrepancies $\tilde{d}_k := b^\delta - A\tilde{x}_k$ satisfy*

$$\|\tilde{d}_k\| \leq \|\tilde{d}_{k-1}\|, \qquad k = 1, 2, 3, \ldots . \tag{2.6}$$

Proof. Since $(\tilde{d}_k)^T \tilde{d}_k = 2\Phi(\tilde{x}_k) + (b^\delta)^T b^\delta$, $k = 0, 1, 2, \ldots$, we have to show that $\Phi(\tilde{x}_k) \leq \Phi(\tilde{x}_{k-1})$. Because the columns of $\tilde{Q}_k$ are A^TA-conjugate to p_{k-1}, it follows form (2.2) and (2.4) that

$$\Phi(\tilde{x}_k) = \min_{\substack{\tilde{y} \in \mathbb{R}^\ell \\ \alpha \in \mathbb{R}}} \Phi(x_{k-1} + \alpha p_{k-1} + \tilde{Q}_k \tilde{y}) = \min_{w \in \mathcal{K}_k(A^TA, A^Tb^\delta) \cup Q} \Phi(w), \qquad k = 1, 2, 3, \ldots .$$

The proposition now follows from

$$\Phi(\tilde{x}_k) = \min_{w \in \mathcal{K}_k(A^TA, A^Tb^\delta) \cup Q} \Phi(w) \leq \min_{w \in \mathcal{K}_{k-1}(A^TA, A^Tb^\delta) \cup Q} \Phi(w) = \Phi(\tilde{x}_{k-1}).$$

We remark that the inequality (2.6) holds for arbitrary initial approximate solutions $x_0 \in \mathbb{R}^n$. $\square$

The initial approximate solution in Proposition 2.1 is an arbitrary vector in $\mathbb{R}^n$. One can show that the norm of the iterates $\tilde{x}_k$ is not guaranteed to be an increasing function of k even when $\tilde{x}_0 = 0$. Thus, an analog of the inequality (1.7) does not hold. Algorithm 1 below summarizes how the computations for the enriched CGNR method can be organized. When $\mathcal{Q} = \emptyset$, lines 9-16 of Algorithm 1 can be removed, and the algorithm simplifies to the standard CGNR algorithm. Algorithm 1 yields the approximate solutions $\tilde{x}_k$ as well as the associated discrepancies $\tilde{d}_k := b^\delta - A\tilde{x}_k$ and residual vectors $\tilde{r}_k := A^T b^\delta - A^T A\tilde{x}_k$.

ALGORITHM 1. *(Enriched CGNR algorithm)*
Input: $A \in \mathbb{R}^{m\times n}$; $x_0 \in \mathbb{R}^n$, $b^\delta \in \mathbb{R}^m$, $\tilde{Q}_0 := [q^{(1)}, q^{(2)}, \dots, q^{(\ell)}] \in \mathbb{R}^{n\times \ell}$;
Output: x_k, $\tilde{x}_k$, d_k, $\tilde{d}_k$, r_k, $\tilde{r}_k$, $k = 0, 1, 2, \dots$;
$d_0 := b^\delta - Ax_0$; $\tilde{d}_0 := d_0$; $r_0 := A^T d_0$; $p_0 := r_0$; $V_0 = A\tilde{Q}_0$; $S_0 = A^T V_0$
for $k = 0, 1, 2, \dots,$ **until** *stopping criterion satisfied*
1. $w_k := Ap_k$
2. $\alpha_k := \frac{r_k^T r_k}{w_k^T w_k}$
3. $x_{k+1} := x_k + \alpha_k p_k$
4. $d_{k+1} := d_k - \alpha_k w_k$
5. $f_k := A^T w_k$
6. $r_{k+1} := r_k - \alpha_k f_k$
7. $\beta_k := \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
8. $p_{k+1} := r_{k+1} + \beta_k p_k$
9. $g_k := \frac{V_k^T w_k}{w_k^T w_k}$
10. $\tilde{Q}_{k+1} := \tilde{Q}_k - p_k g_k^T$
11. $V_{k+1} := V_k - w_k g_k^T$
12. Solve the least-squares problem $\min_{\tilde{y}\in R^n} \|V_{k+1}\tilde{y} - d_{k+1}\|$ *for* $\tilde{y}_{k+1}$
13. $\tilde{x}_{k+1} := x_{k+1} + \tilde{Q}_{k+1}\tilde{y}_{k+1}$
14. $\tilde{d}_{k+1} := d_{k+1} - V_{k+1}\tilde{y}_{k+1}$
15. $S_{k+1} := S_k - f_k g_k^T$
16. $\tilde{r}_{k+1} := r_{k+1} - S_{k+1}\tilde{y}_{k+1}$
end k □

The least-squares problem in line 12 of the algorithm can be solved by QR-factorization of the matrix V_{k+1}. Due to the relation on line 11, the QR-factorization of V_{k+1} can be computed inexpensively by updating the QR-factorization of V_k; see [2, 12] for details. When the number of columns ℓ of the matrices V_j is small, straightforward computation of the QR-factorization of each matrix V_{k+1} generated is also feasible. Typically, ℓ is small and the arithmetic work with the matrices $\tilde{Q}_k$, S_k and V_k amounts to a few vector operations with n-vectors in each iteration. Thus, for ℓ small each iteration with Algorithm 1 requires only a few vector operations with n-vectors, in addition the arithmetic work required for the (standard) CGNR method. We also note that Algorithm 1 requires the storage of a few n-vectors, in addition to the storage requirement for the CGNR method.

3. An enriched CG method for penalized least-squares problems. We describe how the enriched CGNR method of Section 2 can be modified to be used for the solution of the linear system of equations of the form (1.10). Algorithm 2 below shows how the computations can be arranged. The analogue of the minimization problem (2.5) is given by

$$\min_{\tilde{y}\in\mathbb{R}^\ell} \left\| \begin{bmatrix} V_{k+1} \\ \mu^{1/2}\tilde{Q}_{k+1} \end{bmatrix} \tilde{y} - \mu^{-1/2} \begin{bmatrix} 0 \\ r_{k+1} \end{bmatrix} \right\|. \tag{3.1}$$

The QR-factorizations of V_{k+1} and $\tilde{Q}_{k+1}$ can be computed by updating the QR-factorizations of V_k and $\tilde{Q}_k$ using techniques described in [2, 12]. Substituting the QR-factorizations of V_{k+1} and $\tilde{Q}_{k+1}$ into (3.1), and using the fact that the Euclidean norm is invariant under orthogonal transformations, allows us to reduce the (3.1) to a minimization problem of small size.

When $\mu = 0$, Algorithm 2 simplifies to Algorithm 1. The steps of Algorithm 2 that differ from the corresponding steps of Algorithm 1 are marked by "$\leftarrow$".

ALGORITHM 2. *(Enriched CG method for penalized least-squares problems).*
Input: $A \in \mathbb{R}^{m\times n}$; $x_0 \in \mathbb{R}^n$, $b^\delta \in \mathbb{R}^n$, $\tilde{Q}_0 \in \mathbb{R}^{n\times \ell}$; $\mu \geq 0$;
Output: *Computed approximate solutions to the system (1.10);*
$d_0 := b^\delta - Ax_0$; $\tilde{d}_0 := d_0$; $r_0 := A^T d_0 - \mu x_0; \leftarrow$
$p_0 := r_0$; $V_0 = A\tilde{Q}_0$; $S_0 = A^T V_0$
for $k = 0, 1, 2, \ldots,$ **until** *stopping criterion satisfied*
1. $w_k := Ap_k$
2. $\alpha_k := \frac{r_k^T r_k}{w_k^T w_k + \mu p_k^T p_k} \leftarrow$
3. $x_{k+1} := x_k + \alpha_k p_k$
4. $d_{k+1} := d_k - \alpha_k w_k$
5. $f_k := A^T w_k$
6. $r_{k+1} := r_k - \alpha_k(f_k + \mu p_k) \leftarrow$
7. $\beta_k := \frac{r_{k+1}^T r_{k+1}}{r_k^T r_k}$
8. $p_{k+1} := r_{k+1} + \beta_k p_k$
9. $g_k := \frac{V_k^T w_k + \mu \tilde{Q}_k^T p_k}{w_k^T w_k + \mu p_k^T p_k} \leftarrow$
10. $\tilde{Q}_{k+1} := \tilde{Q}_k - p_k g_k^T$
11. $V_{k+1} := V_k - w_k g_k^T$
12. Solve the minimization problem (3.1) for $\tilde{y}_{k+1} \leftarrow$
13. $\tilde{x}_{k+1} := x_{k+1} + \tilde{Q}_{k+1}\tilde{y}_{k+1}$
14. $\tilde{d}_{k+1} := d_{k+1} - V_{k+1}\tilde{y}_{k+1}$
15. $S_{k+1} := S_k - f_k g_k^T$
16. $\tilde{r}_{k+1} := r_{k+1} - (S_{k+1} + \mu\tilde{Q}_{k+1})\tilde{y}_{k+1} \leftarrow$

end k □

We remark that an analogue of Proposition 2.1 for the discrepancies $\tilde{d}_k$ determined by Algorithm 2 does not hold, i.e., there are linear systems of equations (1.9) for which $\tilde{d}_{k+1} > \tilde{d}_k$ for some index k.

Typically, when using Tikhonov regularization it is necessary to solve several linear systems of the form (1.10) for different values of the regularization parameter μ. Frommer and Maass [3] recently discussed how accurately each one of these systems should be solved by the CG method. We update the values of μ in the same manner as Frommer and Maass [3]; see Example 4.4 below for more details. The numerical method determines both a suitable value of μ and an approximate solution of (1.2) that satisfies the discrepancy principle.

4. Numerical examples. This section presents a few computed examples that illustrate how the enriched CGNR method makes it possible to supply information about the wanted solution by choosing appropriate subspaces $\mathcal{Q}$. All examples are concerned with the solution of linear discrete ill-posed problems (1.2) with a right-hand side b^δ that is contaminated by an error η. The quotient $\|\eta\|/\|b\|$, referred to as the noise level, is assumed to be known.

Two difficulties may arise when solving linear discrete ill-posed problems with a contaminated right-hand side by applying the CGNR method to the associated normal equations (1.1). The first one is that the CGNR method when applied to the iterative solution of linear systems of equations with an ill-conditioned matrix typically requires a large number of iterations. The second difficulty is that, in the presence of the error η in the right-hand side b^δ, it may only be possible to carry out a few iterations before the error η is propagated and amplified to such an extent so as to make the computed iterates meaningless. Both of these difficulties can be ameliorated by enriching the Krylov subspaces in which the computed solutions are determined.

The first three examples presented compare Algorithm 1 with the standard CGNR method; the fourth example compares Algorithm 2 with the CG method applied to the solution of (1.10) in the context of Tikhonov regularization. All examples were implemented in Matlab 6.0 and the computations were carried out with about 16 significant decimal digits.

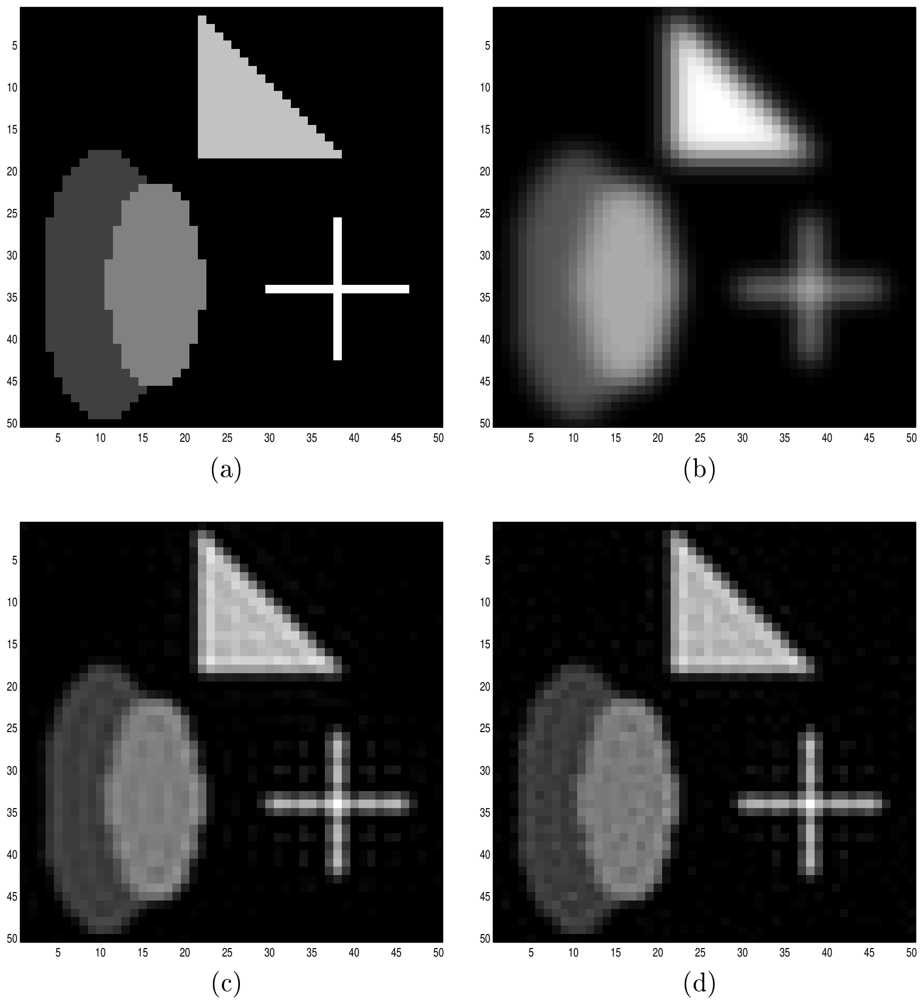


FIG. 4.1. *Example 4.1: (a) Blur- and noise-free image. (b) Available image contaminated by blur and noise. (c) Image restored by 145 iterations with the CGNR method. (d) Image restored by 53 iterations with the enriched CGNR method.*

Example 4.l. Consider the blur- and noise-free image shown in Figure 4.1(a). The image is represented by a 50×50 array of integers whose values range between

0 and 255, representing the gray level at each pixel. It is generated using Matlab code provided by Hansen [8]. The pixel values are stored row-wise in the vector $x_* \in \mathbb{R}^{2500}$. This image is assumed not to be available. The matrix $A \in \mathbb{R}^{2500\times 2500}$ represents a discretized blurring operator; it is the Kronecker product of the Toeplitz matrix $T = [t_{jk}] \in \mathbb{R}^{50\times 50}$ with itself, where

$$t_{jk} = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(j-k)^2}{2\sigma^2}\right), & |j-k| \le \rho, \\ 0, & \text{otherwise}. \end{cases} \tag{4.1}$$

The severity of the blur increases with σ. We let $\sigma := 1.5$ and $\rho := 12\sigma$. The matrix A so obtained is of ill-determined rank. In particular, it is numerically singular.

The blurred, but noise-free, image is given by $b := Ax_*$. Let the error vector η have normally distributed random entries with zero mean, normalized to yield the noise level $\|\eta\|/\|b\| = 1\cdot 10^{-3}$. Define b^δ by (1.3) and δ by (1.4). The image represented by b^δ is shown in Figure 4.1(b). This is the available image that we would like to restore.

We apply the CGNR and enriched CGNR methods, the latter with $\mathcal{Q} := \text{span}\{b^\delta\}$, to the normal equations (1.1) and terminate the iterations according to the discrepancy principle (1.8). The CGNR method yields the approximate solution x_{145}, which represents the image shown in Figure 4.1(c). The enriched CGNR method yields the approximate solution $\tilde{x}_{53}$. The image represented by the latter solution is shown in Figure 4.1(d) and is of similar quality as the image in Figure 4.1(c). The enriched CGNR method, however, requires about 1/3 of the iterations needed by the CGNR method. Typically, the evaluation of matrix-vector products is the dominating computational work in each iterations. Hence, the enriched CGNR method reduces the computational work by about 2/3, compared with the CGNR method, and gives a restored image of similar quality. □

The matrices A and A^T in Example 4.1 are discretizations of smoothing operators. Therefore the Krylov subspaces $\mathcal{K}_k(A^TA, A^Tb^\delta)$ of small dimension are not well suited for the approximation of vectors that are the discretization of discontinuous functions, such as x_*. Enrichment by the vector b^δ reduces the number of iterations required. This also can be observed in the following example.

Example 4.2. The blur- and noise-free image shown in Figure 4.2(a), made available by Nagy [11], is represented by 256×256 pixels. The pixel values are stored in $x_* \in \mathbb{R}^{256^2}$. This image is assumed not to be available. The matrix $A \in \mathbb{R}^{256^2\times 256^2}$ represents a discretized blurring operator, and is the Kronecker product of the Toeplitz matrix $T = [t_{jk}] \in \mathbb{R}^{256\times 256}$ with itself, where the entries t_{jk} are given by (4.1) with $\sigma := 3.5$ and $\rho := 12\sigma$. The matrix A so defined is of ill-determined rank.

Similarly as in Example 4.1, the blurred, noise-free, image is given by $b := Ax_*$. The error vector η has normally distributed random entries with zero mean. It is normalized so that $\|\eta\|/\|b\| = 1\cdot 10^{-3}$. The vector b^δ defined by (1.3) represents the image shown in Figure 4.2(b). We would like to restore this image. The value of δ is given by (1.4).

We apply the CGNR and enriched CGNR methods, the latter with $\mathcal{Q} := \text{span}\{b^\delta\}$, to the normal equations (1.1) and terminate the iterations according to the discrepancy principle (1.8). The CGNR method yields the approximate solution x_{238}, which represents the image shown in Figure 4.2(c). The enriched CGNR method determines the approximate solution $\tilde{x}_{81}$, which is displayed in Figure 4.2(d). The images in the Figures 4.2(c) and (d) are of similar quality, however, the enriched CGNR method requires only about 1/3 of the iterations needed by the (standard) CGNR method. □

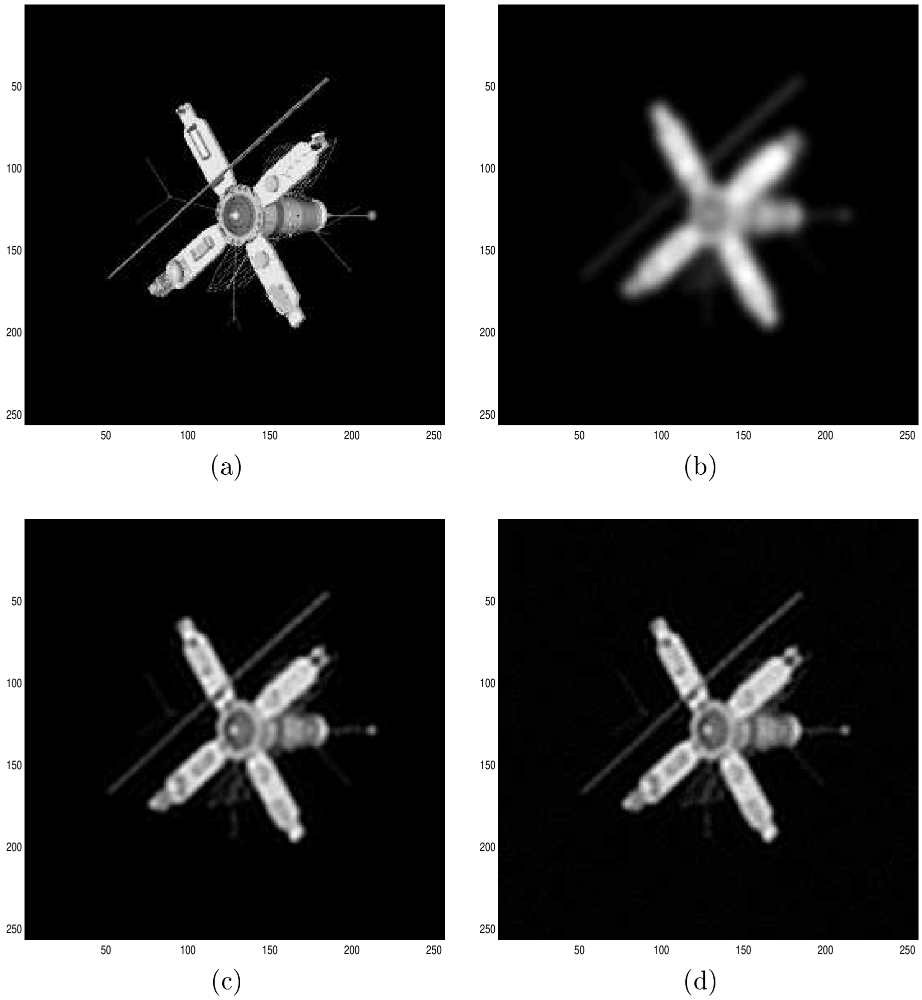


FIG. 4.2. *Example 4.2: (a) Blur- and noise-free image. (b) Available image contaminated by blur and noise. (c) Image restored by 238 iterations with the CGNR method. (d) Image restored by 81 iterations with the enriched CGNR method.*

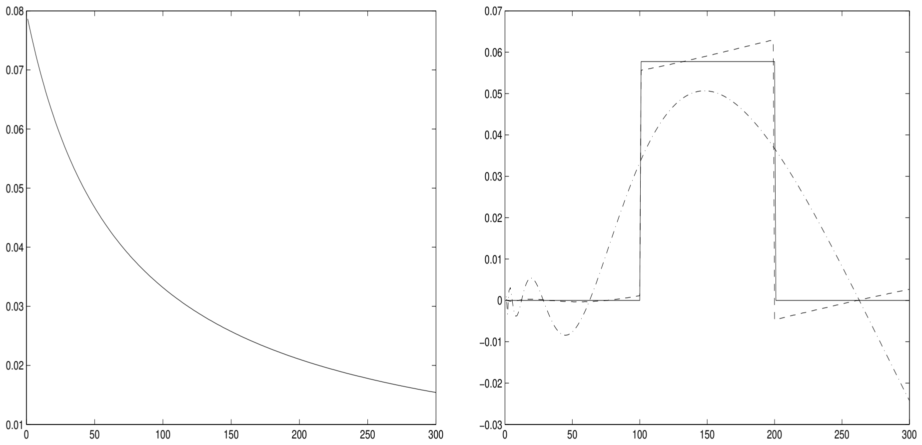
Example 4.3. Let $x_* \in \mathbb{R}^{300}$ be a discrete sample of the function

$$f(t) := \begin{cases} 1, & \text{if } \frac{1}{3} < t < \frac{2}{3}, \\ 0, & \text{if } 0 \leq t \leq \frac{1}{3} \text{ or } \frac{2}{3} \leq t \leq 1. \end{cases}$$

at 300 equidistant points. Notice that f has two jump discontinuities. Let $A \in \mathbb{R}^{300\times 300}$ be a Cauchy matrix with entries

$$a_{ij} := \frac{1}{i + 0.5j}.$$

The matrix A is of ill-determined rank; it is numerically singular. Define $b^\delta := Ax_* + \eta$, where the entries of η are normally distributed random numbers with zero mean, scaled to yield the noise level $1 \cdot 10^{-4}$. We use the discrepancy principle (1.8) to decide when to terminate the iterations.



(a) (b)

FIG. 4.3. *Example 4.3: (a) Right-hand side b^δ. (b) Exact solution x_* (solid curve), approximate solution x_{13} determined by the CGNR method (dash-dotted curve), approximate solution $\tilde{x}_3$ determined by the enriched CGNR method (dashed curve).*

Figure 4.3(a) shows the right-hand side vector b^δ, and Figure 4.3(b) displays the desired solution x_* (solid curve), the approximate solution x_{13} computed by the CGNR method (dash-dotted curve), and the approximate solution $\tilde{x}_3$ computed by the enriched CGNR method with $\mathcal{Q} := \mathrm{span}\{q^{(1)}, q^{(2)}, q^{(3)}\}$, where $q^{(1)} := b^\delta$ and the vectors $q^{(2)}$ and $q^{(3)}$ model the jump discontinuities of f (dashed curve). They have the components

$$q^{(2)}(i) := \begin{cases} 1, & \text{if } \ 100 < i, \\ 0, & \text{otherwise}, \end{cases}$$

and

$$q^{(3)}(i) := \begin{cases} 1, & \text{if } \ i < 200, \\ 0, & \text{otherwise}. \end{cases}$$

Figure 4.3(b) shows that the CGNR method without any additional a priori information about x_* is unable to recover the discontinuities, while the enriched CGNR method is able to compute a fairly accurate approximation of x_* with little arithmetic work. □

Example 4.4. Consider the Fredholm integral equation of the first kind,

$$\int_0^\pi \exp(s\cos(t))x(t)dt = 2\frac{\sinh(s)}{s}, \qquad 0 \le s \le \frac{\pi}{2}, \tag{4.2}$$

with solution

$$x(t) := \sin(t). \tag{4.3}$$

This equation is discussed by Baart [1]. We use the Matlab program baart in the REGULARIZATION TOOLS package by Hansen [8] to discretize the integral equation by a Galerkin method with 1000 orthonormal box functions. This gives a nonsymmetric

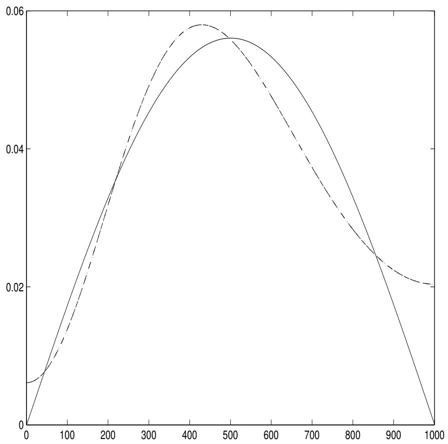


FIG. 4.4. *Example 4.4: Exact solution $\hat{x}$ (solid curve), approximate solution determined by CG method applied to the solution of (1.10) (dash-dotted curve), and approximate solution determined by the enriched CG method applied to (1.10) (dashed curve).*

matrix $A \in \mathbb{R}^{1000\times 1000}$ and a right-hand side vector $b \in \mathbb{R}^{1000}$. The matrix A is of ill-determined rank.

Let $t_j := (2j-1)\pi/2000$, $1 \leq j \leq 1000$, and define the scaled tabulation of the solution (4.3),

$$\hat{x} := \sqrt{\frac{\pi}{1000}}[x(t_1), x(t_2), \ldots, x(t_{1000})]^T. \tag{4.4}$$

This vector is a good approximation of the solution of the linear system of equations determined by the code baart. We consider it the exact solution of the system (1.5).

Let the entries of the error vector η be normally distributed with zero mean, and scaled so that we obtain the noise level $\|\eta\|/\|b\| = 1\cdot 10^{-4}$. The contaminated right-hand side vector b^δ in (1.2) is defined by (1.3).

We compute an approximation of x_* by Tikhonov regularization, using the discrepancy principle to determine a suitable value of the regularization parameter μ. Following Frommer and Maass [3], we solve a sequence of linear systems of equations of the form (1.10) for $\mu := \mu_k := 2^{-k}$, $k = 0, 1, \ldots$, until for some μ_k an associated approximate solution $x_j^{(\mu_k)}$ of (1.10), computed by applying j steps of the CG method to the solution of (1.10), satisfies

$$\|Ax_j^{(\mu_k)} - b^\delta\| \leq \delta. \tag{4.5}$$

As in [3], we carry out iterations with each linear system (1.10) until one of the inequalities (4.5) or

$$\|d_j^{(\mu_k)}\| - \frac{1}{2\sqrt{\mu}}\|r_j^{(\mu_k)}\| > \delta$$

holds, where $d_j^{(\mu_k)} := b^\delta - Ax_j^{(\mu_k)}$ is the discrepancy and $r_j^{(\mu_k)} := A^T b^\delta - A^T Ax_j^{(\mu_k)}$ the residual associated with $x_j^{(\mu_k)}$. When the value of the regularization parameter is updated, we use the last determined approximate solution for the previous linear system (1.10) as initial approximate solution, as suggested in [3]. Similarly as Frommer and Maass [3], we choose the initial value of μ to be one, and scale the right-hand side vector b^δ and matrix A so that $\|b^\delta\| = 1$ and $\|A^T b^\delta\| = 2$. This method requires 33 iterations. Figure 4.4 displays the computed approximate solution (dash-dotted curve) and the exact solution $\hat{x}$ (solid curve).

When replacing the CG method for the solution of the linear systems of equations (1.10) for the different values of $\mu = \mu_k$ by Algorithm 2 with $\mathcal{Q} := \text{span}\{b^\delta\}$, we obtain the approximate solution displayed by the dashed curve in Figure 4.4. The computation of the latter solution requires only 23 iterations. Thus, using the enriched CG method reduces the number of matrix-vector product evaluations with the matrices A and A^T by approximately 30% and gives a computed approximate solution of about the same quality as the dash-dotted curve. $\square$

The examples of this section, and computational experience from numerous other examples, indicate that more desirable approximate solutions often can be determined with less computational work by enriching the Krylov subspaces for the CGNR method by a space that allows the representation of certain known important features of the solution. For many discrete ill-posed problems the number of iterations can be reduced by enriching the Krylov subspaces used by the CGNR method by the right-hand side vector b^δ. When the matrix $A \in \mathbb{R}^{m\times n}$ is not square, restriction or prolongation of b^δ is required in order to obtain a vector in $\mathbb{R}^n$ that can enrich the Krylov subspaces generated. How beneficial this simple enrichment is depends on b^δ, A and x_*.

The matrices in the computed examples are all of ill-determined rank, however, we remark that Algorithms 1 and 2 also can be applied to linear systems of equations with matrices of well-defined rank.

We conclude with a few remarks on possible extensions of the methods of the present paper. While this paper is concerned with enriching the CGNR method and the CG method applied to the linear system of equations (1.10), analogous enrichments of other iterative methods may also be attractive. Enrichments of the CG method when applied to linear systems of equations with a symmetric positive definite matrix and of the MINRES and MR-II iterative methods when applied to linear systems of equations with a symmetric, possibly indefinite, matrix can be derived similarly as the methods of the present paper. Enrichment of the GMRES method is particularly easy to carry out, because bases of Krylov subspaces $\mathcal{K}_k(A, b^\delta)$, $k = 1, 2, 3, \ldots$, are explicitly orthogonalized and stored. We remark that enrichment can be used together with preconditioning; we refer to Hanke et al. [6] for the description of a preconditioner for linear discrete ill-posed problems of the form (1.1). Termination criteria for the iterations, other then the discrepancy principle, can also be used.

Acknowledgement. We would like to thank Bob Plemmons for helpful comments.

REFERENCES

[1] M. L. Baart, The use of auto-correlation for pseudo-rank determination in noisy ill-conditioned least-squares problems, IMA J. Numer. Anal. 2 (1982), pp. 241–247.

[2] J. Daniel, W. B. Gragg, L. Kaufman and G. W. Stewart, Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR-factorization, Math. Comp., 20 (1976), pp. 772–795.

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

[4] M. Hanke, Conjugate Gradient Type Methods for Ill-Posed Problems, Longman, Harlow, 1995.

[5] M. Hanke and P. C. Hansen, Regularization methods for large-scale problems, Surv. Math. Ind., 3 (1993), pp. 253–315.

[6] M. Hanke, J. G. Nagy and R. J. Plemmons, Preconditioned iterative regularization for ill-posed problems, in Numerical Linear Algebra, eds. L. Reichel, A. Ruttan and R. S. Varga, Berlin, de Gruyter, 1993, pp. 141–163.

[7] M. Hanke, J. G. Nagy and C. Vogel, Quasi-Newton approach to nonnegative image restorations, Linear Algebra Appl., 316 (2000), pp. 223–236.

[8] P. C. Hansen, Regularization tools: A Matlab package for analysis and solution of discrete ill-posed problems, Numer. Algorithms, 6 (1994), pp. 1–35.

[9] P. C. Hansen, Rank Deficient and Discrete Ill-Posed Problems, SIAM, Philadelphia, 1998.

[10] M. R. Hestenes and E. Stiefel, Methods of conjugate gradients for solving linear systems, J. Res. Nat. Bur. Standards, 49 (1952), pp. 409–436.

[11] J. G. Nagy, RestoreTools: An object oriented Matlab package for image restoration. Available at the web site `http://www.mathcs.emory.edu/~nagy/`.

[12] L. Reichel and W. B. Gragg, Algorithm 686: FORTRAN subroutines for updating the QR decomposition of a matrix, ACM Trans. Math. Software, 16 (1990), pp. 369–377.

[13] Y. Saad, Iterative Methods for Sparse Linear Systems, PWS, Boston, 1996.