CHOOSING REGULARIZATION PARAMETERS IN ITERATIVE METHODS FOR ILL-POSED PROBLEMS*

MISHA E. KILMER† AND DIANNE P. O'LEARY‡

Abstract. Numerical solution of ill-posed problems is often accomplished by discretization (projection onto a finite dimensional subspace) followed by regularization. If the discrete problem has high dimension, though, typically we compute an approximate solution by projection onto an even smaller dimensional space, via iterative methods based on Krylov subspaces. In this work we present efficient algorithms that regularize after this second projection rather than before it. We prove some results on the approximate equivalence of this approach to other forms of regularization and we present numerical examples.

Key words. ill-posed problems, regularization, discrepancy principle, iterative methods, L-curve, Tikhonov, TSVD, projection, Krylov subspace

65R30,65F20

Running Title: Choosing Regularization Parameters

1. Introduction.

Linear, discrete ill-posed problems of the form

$$Ax = b \tag{1}$$

or

$$\min_x \|Ax - b\|_2, \quad \text{or equivalently,} \quad A^*Ax = A^*b \tag{2}$$

arise, for example, from the discretization of first-kind Fredholm integral equations and occur in a variety of applications. We shall assume that the full-rank matrix A is $m \times n$, with $m \geq n$ in (2) and $m = n$ in (1). In discrete ill-posed problems, A is ill-conditioned and there is no gap in the singular value spectrum. Typically, the right hand side b contains noise due to measurement and/or approximation error. This noise, in combination with the ill-conditioning of A, means that the exact solution of (1) or (2) has little relationship to the noise-free solution and is worthless. Instead, we use a *regularization* method to determine a solution that approximates the noise-free solution. Regularization methods replace the original operator by a better-conditioned but related one in order to diminish the effects of noise in the data and produce a *regularized solution* to the original problem. Sometimes this regularized problem is too large to solve exactly. In that case, we typically compute an approximate solution by projection onto an even smaller dimensional space, perhaps via iterative methods based on Krylov subspaces.

The conditioning of the new problem is controlled by one or more *regularization parameters* specific to the method. A large regularization parameter yields a new well-conditioned problem, but its solution may be far from the noise-free solution since the new operator is a poor approximation to A. A small regularization parameter generally yields a solution very close to the noise-contaminated exact solution of (1) or (2), and hence its distance from the noise-free solution also can be large. Thus,

* This work was supported by the National Science Foundation under Grants CCR 95-03126 and CCR-97-32022 and by the Army Research Office, MURI Grant DAAG55-97-1-0013.

† Dept. of Computer and Electrical Engineering, Northeastern University, Boston, MA 02115 (mkilmer@ece.neu.edu)

‡ Dept. of Computer Science and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742 (oleary@cs.umd.edu).

a key issue in regularization methods is to choose a regularization parameter that balances the error due to noise with the error due to regularization.

A wise choice of regularization parameter is obviously crucial to obtaining useful approximate solutions to ill-posed problems. For problems small enough that a rank-revealing factorization or singular value decomposition of A can be computed, there are well-studied techniques for computing a good regularization parameter. These techniques include the Discrepancy Principle [8], generalized cross-validation (GCV) [9], and the L-curve [15]. For larger problems treated by iterative methods, though, the parameter choice is much less understood. If regularization is applied to the projected problem that is generated by the iterative method, then there are essentially two regularization parameters: one for the standard regularization algorithms, such as Tikhonov or truncated SVD, and one controlling the number of iterations taken. One subtle issue is that the standard regularization parameter that is correct for the discretized problem may not be the optimal one for the lower-dimensional problem actually solved by the iteration, and this observation leads to the research discussed in this paper. At first glance, there can appear to be a lot of work associated with the selection of a good regularization parameter, and many algorithms proposed in the literature are needlessly complicated. But by regularizing after projection by the iterative method, so that we are regularizing the lower dimensional problem that is actually being solved, much of this difficulty vanishes.

The purpose of this paper is to present parameter selection techniques designed to reduce the regularization work for iterative methods such as Krylov subspace techniques. Our paper is organized as follows. In §2, we will give an overview of the regularization methods we will be considering, and we follow up in §3 by surveying some methods for choosing the corresponding regularization parameters. In §4, we show how parameter selection techniques for the original problem can be applied instead to a projected problem obtained from an iterative method, greatly reducing the cost without much degradation in the solution. We give experimental results in §5 and conclusions and future work in §6.

2. Regularization background. In the following we shall assume that $b = b_{true} + e$, where b_{true} denotes the unperturbed data vector and e denotes zero-mean white noise. We will also assume that b_{true} satisfies the discrete Picard condition; that is, the spectral coefficients of b_{true} decay faster, on average, than the singular values.

Under these assumptions, it is easy to see why the exact solution to (1) or (2) is hopelessly contaminated by noise. Let $\hat{U}\Sigma\hat{V}^*$ denote the singular value decomposition of A, where the columns of $\hat{U}$ and $\hat{V}$ are the singular vectors, and the singular values are ordered as $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$. Then the solution to (1) or (2) is given by

$$x = \sum_{i=1}^{n} \frac{\hat{u}_i^* b}{\sigma_i} \hat{v}_i = \sum_{i=1}^{n} \left(\frac{\hat{u}_i^* b_{true}}{\sigma_i} + \frac{\hat{u}_i^* e}{\sigma_i} \right) \hat{v}_i. \tag{3}$$

As a consequence of the white noise assumption, $|\hat{u}_i^* e|$ is roughly constant for all i, while the discrete Picard condition guarantees that $|\hat{u}_i^* b_{true}|$ decreases with i faster than σ_i does . The matrix A is ill-conditioned, so small singular values magnify the corresponding coefficients $\hat{u}_i^* e$ in the second sum, and it is this large contribution of noise from the approximate null space of A that renders the exact solution x defined in (3) worthless. The following regularization methods try in different ways to lessen the contribution of noise to the solution. For further information on these methods, see, for example, [17].

2.1. Tikhonov regularization. One of the most common methods of regularization is Tikhonov regularization [34]. In this method, the problem (1) or (2) is replaced with the problem of solving

$$\min_{x} \|Ax - b\|_2^2 + \lambda^2 \|Lx\|_2^2 \tag{4}$$

where L denotes a matrix, often chosen to be the identity matrix I or a discrete derivative operator, and λ is a positive scalar regularization parameter. For ease in notation, we will assume that $L = I$. Solving (4) is equivalent to solving

$$(A^*A + \lambda^2 I)x_\lambda = A^*b. \tag{5}$$

In analogy with (3) we have

$$x_\lambda = \sum_{i=1}^{n} \left(\frac{\sigma_i \, \hat{u}_i^* b_{true}}{\sigma_i^2 + \lambda^2} + \frac{\sigma_i \, \hat{u}_i^* e}{\sigma_i^2 + \lambda^2} \right) \hat{v}_i. \tag{6}$$

In this solution, the contributions from noise components $\hat{u}_i^* e$ for values of $\sigma_i < \lambda$ are much smaller than they are in (3), and thus x_λ can be closer to the noise-free solution than x is. If λ is too large, however, $A^*A + \lambda^2 I$ is very far from the original operator A^*A, and x_λ is very far from x_{true}, the solution to (2) when $e = 0$. Conversely, if λ is too small, the singular values of the new operator $A^*A + \lambda^2 I$ are close to those of A^*A; thus $x_\lambda \approx x$, so small singular values again greatly magnify noise components.

2.2. Truncated SVD. In the truncated SVD method of regularization, the regularized solution is chosen simply by truncating the expansion in (3) as

$$x_\ell = \sum_{i=1}^{n-\ell} \frac{\hat{u}_i^* b}{\sigma_i} \hat{v}_i. \tag{7}$$

Here the regularization parameter is ℓ, the number of terms to be dropped from the sum. Observe that if ℓ is small, very few terms are dropped from the sum, so x_ℓ resembles x in that the effects of noise are large. If ℓ is too large, however, important information could be lost; such is the case if $\hat{u}_i^* b_{true} \gg \hat{u}_i^* e$ for some $i > n - \ell$.

An alternative, yet related, approach to TSVD is an approach introduced by Rust [31] where the truncation strategy is based on the value of each spectral coefficient $\hat{u}_i^* b$ itself. The strategy is to include in the sum (3) only those terms corresponding to a spectral coefficient $\hat{u}_i^* b$ whose magnitude is greater than or equal to some tolerance ρ, which can be regarded as the regularization parameter.

2.3. Projection and iterative methods. Solving (5) or (7) can be impractical if n is large, but fortunately, regularization can be achieved through projection onto a subspace; see, for example, [7]. The truncated SVD is an example of one such projection: the solution is constrained to lie in the subspace spanned by the singular vectors corresponding to the largest $n - \ell$ singular values. Other projections can be more economical. In general, we constrain our regularized solution to lie in some k-dimensional subspace of $\mathcal{C}^n$, spanned by the columns of an $n \times k$ matrix $Q^{(k)}$. For example, we choose $x_{reg}^{(k)} = Q^{(k)} y^{(k)}$ where $y^{(k)}$ solves

$$\min_{y \in \mathcal{C}^k} \|AQ^{(k)} y - b\|_2^2 \tag{8}$$

or equivalently

$$(9) \qquad (Q^{(k)})^* A^* A Q^{(k)} y = (Q^{(k)})^* A^* b.$$

The idea is that with an appropriately chosen subspace, the operator $(Q^{(k)})^* A^* A Q^{(k)}$ will be better conditioned than the original operator and hence that $x_{reg}^{(k)}$ will approximate x_{true} well on that subspace.

This projection is often achieved through the use of iterative methods such as conjugate gradients, GMRES, QMR, and other Krylov subspace methods. The matrix $Q^{(k)}$ then contains orthonormal columns generated via a Lanczos tridiagonalization or bidiagonalization process [27, 1]. In this case, $Q^{(k)}$ is a basis for some k-dimensional Krylov subspace (i.e., the subspace $\mathcal{K}_k(c, K)$ spanned by the vectors $c, Kc, \ldots, K^{k-1}c$ for some matrix K and vector c). The regularized solutions $x_{reg}^{(k)}$ are generated iteratively as the subspaces are built. Krylov subspace algorithms such as CG, CGLS, GMRES, and LSQR tend to produce, at early iterations, solutions that resemble x_{true} in the subspace spanned by (right) singular vectors of A corresponding to the largest singular values. At later iterations, however, these methods start to reconstruct increasing amounts of noise into the solution. This is due to the fact that for large k, the operator $(Q^{(k)})^* A^* A Q^{(k)}$ approaches the ill-conditioned operator A^*A. Therefore, the choice of the regularization parameter k, the stopping point for the iteration and the dimension of the subspace, is very important.[1]

2.4. Hybrid methods: projection plus regularization. Another important family of regularization methods, often referred to as hybrid methods [17], was introduced by O'Leary and Simmons [27]. These methods combine a projection method with a direct regularization method such as TSVD or Tikhonov regularization. The problem is projected onto a particular subspace of dimension k, but typically the restricted operator in (9) is still ill-conditioned. Therefore, a regularization method is applied to the projected problem. Since the dimension k is usually small relative to n, regularization of the restricted problem is much less expensive. Yet, with an appropriately chosen subspace, the end results can be very similar to those achieved by applying the same direct regularization technique to the original problem. We will become more precise about how "similar" the solutions are in §4.5. Because the projected problems are usually generated iteratively by a Lanczos method, this approach is useful when A is sparse or structured in such a way that matrix-vector products can be handled efficiently with minimal storage.

3. Existing parameter selection methods. In this section, we discuss a sampling of the parameter selection techniques that have been proposed in the literature. They differ in the amount of a priori information required as well as in the decision criteria.

3.1. The Discrepancy Principle. If some extra information is available – for example, an estimate of the variance of the noise vector e – then the regularization parameter can be chosen rather easily. Morozov's Discrepancy Principle [25] says that if δ is the expected value of $\|e\|_2$, then the regularization parameter should be chosen so that the norm of the residual corresponding to the regularized solution x_{reg} is $\tau\delta$; that is,

$$(10) \qquad \|Ax_{reg} - b\|_2 = \tau\delta,$$

[1] Usually, small values of the regularization parameter correspond to a closer solution to the noisy equation, but despite this, we will call k, rather than $1/k$, the regularization parameter.

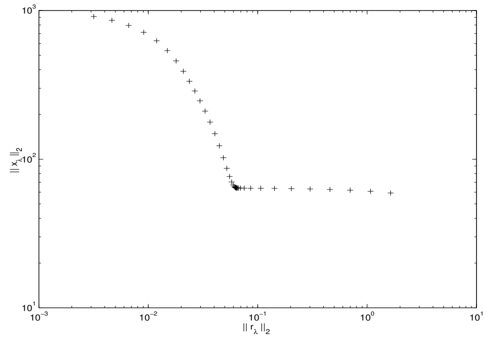

FIG. 1. *Example of a typical L-curve. This particular L-curve corresponds to applying Tikhonov regularization to the problem in Example 2*

where $\tau > 1$ is some predetermined real number. Note that as $\delta \to 0$, $x_{reg} \to x_{true}$. Other methods based on knowledge of the variance are given, for example, in [12, 5].

3.2. Generalized Cross-Validation. The Generalized Cross-Validation (GCV) parameter selection method does not depend on a priori knowledge about the noise variance. This idea of Golub, Heath, and Wahba [9] is to find the parameter λ that minimizes the GCV functional

$$G(\lambda) = \frac{\|(I - AA_\lambda^\sharp)b\|_2^2}{(\mathrm{trace}(I - AA_\lambda^\sharp))^2}, \tag{11}$$

where $A_\lambda^\sharp$ denotes the matrix that maps the right hand side b onto the regularized solution x_λ. In Tikhonov regularization, for example, $A_\lambda^\sharp$ is

$$(A^*A + \lambda^2 I)^{-1} A^*.$$

GCV chooses a regularization parameter that is not too dependent on any one data measurement [11, 12.1.3].

3.3. The L-Curve. One way to visualize the tradeoff between regularization error and error due to noise is to plot the norm of the regularized solution versus the corresponding residual norm for each of a set of regularization parameter values. The result is the *L-curve*, introduced by Lawson and popularized by Hansen [15]. See Figure 1 for a typical example. As the regularization parameter increases, noise is damped, so that the norm of the solution decreases while the residual increases. Intuitively, the best regularization parameter should lie on the corner of the L-curve, since for values higher than this, the residual increases without reducing the norm of the solution much, while for values smaller than this, the norm of the solution increases rapidly without much decrease in residual. In practice, only a few points on the L-curve are computed and the corner is located by approximate methods, estimating the point of maximum curvature [19].

Like GCV, this method of determining a regularization parameter does not depend on specific knowledge about the noise vector.

3.4. Disadvantages of these parameter choice algorithms. The appropriate choice of regularization parameter – especially for projection algorithms – is a difficult problem, and each method has severe flaws.

	Basic cost	Added Cost		
		Disc.	GCV	L-curve
Tikhonov	$O(mn^2)$	$O(p(m+n))$	$O(p(n+m))$	$\mathrm{O}(p(m+n))$
TSVD	$O(mn^2)$	$O(m)$	$O(m)$	$O(m+n)$
Rust's TSVD	$O(mn^2)$	$O(m \log m)$	$O(m \log m)$	$O(m \log m)$
Projection	$O(qk)$	0	$O(q)$	$O(q)$

TABLE 1

Summary of additional flops needed to compute the regularization parameter for each four regularization methods with various parameter selection techniques. Notation:
q is the cost of multiplication of a vector by A.
p is the number of discrete parameters that must be tried;
k is the dimension of the projection.
m and n are problem dimensions.

The Discrepancy Principle is convergent as the noise goes to zero, but it relies on knowing information that is often unavailable or incorrectly estimated. Even with a correct estimate of the variance, the solutions tend to be oversmoothed [20, pg. 96] (see also the discussion in §6.1 of [15]).

One noted difficulty with GCV is that G can have a very flat minimum, making it difficult to determine the optimal λ numerically [35].

The L-curve is usually more tractable numerically, but its limiting properties are nonideal. The solution estimates fail to converge to the true solution as $n \to \infty$ [36] or as the error norm goes to zero [6]. All methods that assume no knowledge of the error norm – including GCV – have this latter property [6].

For further discussion and references about parameter choice methods, see [5, 17]. The cost of these methods is tabulated in Table 1.

3.5. Previous work on parameter choice for hybrid methods. At first glance, it appears that for Tikhonov regularization, multiple systems of the form (5) must be solved in order to evaluate candidate values of λ for the Discrepancy Principle or the L-curve. Techniques have been suggested in the literature for solving these systems using projection methods.

Chan and Ng [4], for example, note that the systems involve the closely related matrices matrices $C(\lambda) = A^*A + \lambda I$, and they suggest solving the systems simultaneously using a Galerkin projection method on a sequence of "seed" systems. Although this is economical in storage, it can be unnecessarily expensive in time because they do not exploit the fact that for each fixed k, the Krylov subspace $\mathcal{K}_k(A^*b, C(\lambda))$ is the same for all values of λ.

Frommer and Maass [8] propose two algorithms for approximating the λ that satisfies the Discrepancy Principle (10). The first is a "truncated cg" approach in which they use conjugate gradients to solve k systems of the form (5), truncating the iterative process early for large λ and using previous solutions as starting guesses for later problems. Like Chan and Ng, this algorithm does not exploit any of the redundancy in generating the Krylov-subspaces for each λ_i. The second method they propose, however, does exploit the redundancy so that the CG iterates for all k systems can be updated simultaneously with no extra matrix-vector products. They stop their "shifted cg" algorithm when $\|Ax_\lambda - b\|_2 \le \tau\delta$ for one of their λ values. Thus the number of matrix-vector products required is twice the number of iterations for this particular system to converge. We note that while the algorithms we propose in §4 for finding a good value of λ are based on the same key observation regarding the Krylov subspace, our methods will usually require less work than the shifted cg algorithm.

Calvetti, Golub, and Reichel [3] compute upper and lower bounds on the L-curve generated by the matrices $C(\lambda)$ using a Lanczos bidiagonalization process. From this, they approximate the best parameter for Tikhonov regularization without projection. In §4, we choose instead to approximate the best parameter for Tikhonov regularization on the projected problem, since this is the approximation to the continuous problem that is actually being used.

Kaufman and Neumaier [21] suggest an envelope guided conjugate gradient approach for the Tikhonov L-curve problem. Their method is more complicated than the methods we propose because they maintain nonnegativity constraints on the variables.

Substantial work has also been done on TSVD regularization of the projected problems. Björck, Grimme, and van Dooren [2] use GCV to determine the truncation point for the projected SVD. Their emphasis is on stable ways to maintain an accurate factorization when many iterations are needed, and they use full reorthogonalization and implicit restart strategies. O'Leary and Simmons [27] take a somewhat different viewpoint that the problem should be preconditioned appropriately so that a massive number of iterations is unnecessary. That viewpoint is echoed in this current work, so we implicitly assume that the problem has been left-preconditioned or "filtered" [27]. For example, in place of (4), we solve

$$\min_x \|M^{-1}Ax - M^{-1}b\|_2^2 + \lambda^2\|x\|_2^2$$

for a square preconditioner M. See [14, 26, 24, 23] for preconditioners appropriate for certain types of ill-posed problems. Note that we could alternately have considered right preconditioning, which amounts to solving, in the Tikhonov case,

$$\min_y \| \begin{bmatrix} A \\ \lambda I \end{bmatrix} M^{-1}y - \begin{bmatrix} b \\ 0 \end{bmatrix} \|,$$

for y_λ then setting $x_\lambda = M^{-1}y_\lambda$. Note that either left or right preconditioning effectively changes the balance between the two terms in the minimization.

4. Regularizing the projected problem. In this section we develop nine approaches to regularization using Krylov methods. Many Krylov methods have been proposed; for ease of exposition we focus on just two of these: the LSQR algorithm of Paige and Saunders [29] and the GMRES algorithm of Saad and Schultz [33].

The LSQR algorithm of Paige and Saunders [29] iteratively computes the bidiagonalization introduced by Golub and Kahan [10]. Given a vector b, the algorithm is as follows [29, Alg. Bidiag 1]:

> Compute a scalar β_1 and a vector u_1 of length one so that $\beta_1 u_1 = b$.
> Similarly, determine α_1 and v_1 so that $\alpha_1 v_1 = A^T u_1$.
> For i = 1,2,...
>   Let $\beta_{i+1}u_{i+1} = Av_i - \alpha_i u_i$ and $\alpha_{i+1}v_{i+1} = A^T u_{i+1} - \beta_{i+1}v_i$,
>   where the non-negative scalars α_{i+1} and β_{i+1} are chosen
>   so that u_{i+1} and v_{i+1} have length one.
> End for

The vectors u_i, v_i are called the left and right Lanczos vectors respectively. The algorithm can be rewritten in matrix form by first defining the matrices

$$U_k \equiv [u_1, \ldots, u_k]; V_k \equiv [v_1, \ldots, v_k];$$

$$B_k \equiv \begin{bmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \beta_3 & \ddots & \\ & & \ddots & \alpha_k \\ & & & \beta_{k+1} \end{bmatrix}.$$

With e_i denoting the ith unit vector, the following relations can be established:

$$b = \beta_1 u_1 = \beta_1 U_{k+1} e_1 \,, \tag{12}$$

$$AV_k = U_{k+1} B_k \,, \tag{13}$$

$$A^T U_{k+1} = V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T \,, \tag{14}$$

$$V_k^* V_k = I_k \ , \qquad U_{k+1}^* U_{k+1} = I_{k+1} \,, \tag{15}$$

where the subscript on I denotes the dimension of the identity.

Now suppose we want to solve

$$\min_{x \in \mathcal{S}} \|b - Ax\|_2 \tag{16}$$

where $\mathcal{S}$ denotes the k-dimensional subspace spanned by the first k Lanczos vectors v_i. The solution we seek is of the form $x^{(k)} = V_k y^{(k)}$ for some vector $y^{(k)}$ of length k. Define $r^{(k)} = b - Ax^{(k)}$ to be the corresponding residual. From the relations above, observe that in exact arithmetic

$$\begin{aligned} r^{(k)} &= \beta_1 u_1 - AV_k y^{(k)} \\ &= U_{k+1}(\beta_1 e_1 - B_k y^{(k)}) \end{aligned}$$

Since U_{k+1} has, in exact arithmetic, orthonormal columns, we have

$$\|r^{(k)}\|_2 = \|\beta_1 e_1 - B_k y^{(k)}\|_2. \tag{17}$$

Therefore, the projected problem we wish to solve is

$$\min_{y^{(k)}} \|\beta_1 e_1 - B_k y^{(k)}\|_2. \tag{18}$$

Solving this minimization problem is equivalent to solving the normal equations involving the bidiagonal matrix:

$$B_k^* B_k y^{(k)} = \beta_1 B_k^* e_1. \tag{19}$$

Typically k is small, so reorthogonalization to combat the effects of inexact arithmetic might or might not be necessary. The matrix B_k may be ill-conditioned because some of its singular values approximate some of the small singular values of A. Therefore solving the projected problem might not yield a good solution $y^{(k)}$. However, we can use any of the methods of Section 3 to regularize this projected problem; we discuss options in detail below. As alluded to in §4, the idea is to generate $y_{reg}^{(k)}$, the regularized solution to (18), and then to compute a regularized solution to (16) as $x_{reg}^{(k)} = V_k y_{reg}^{(k)}$.

If we used the algorithm GMRES instead of LSQR, we would derive similar relations. Here, though, the U and V matrices are identical and the B matrix is upper Hessenberg rather than bidiagonal. Conjugate gradients would yield similar relationships.

For cost comparisons for these methods, see Tables 1 and 2. Storage comparisons are given in Tables 3 and 4.

4.1. Regularization by projection. As mentioned earlier, if we terminate the iteration after k steps, we have projected the solution onto a k dimensional subspace and this has a regularizing effect that is sometimes sufficient. Determining the best value of k can be accomplished, for instance, by one of our three methods of parameter choice:

1. **Discrepancy Principle.**
   In this case, we stop the iteration for the smallest value of k for which $\|r_k\| \leq \tau\delta$. Both LSQR and GMRES have recurrence relations for determining $\|r_k\|$ using scalar computations, without computing either r_k or x_k [29, 32].
2. **GCV.**
   For the projected problems (see §4.1) defined by either LSQR or GMRES, the operator $AA^\sharp$ is given by
   $$U_{k+1} B_k B_k^\dagger U_{k+1}^*$$
   where $B_k^\dagger$ is the pseudo-inverse of the matrix B_k. Thus from (11), the GCV functional is [17]
   $$G(k) = \frac{\|r^{(k)}\|_2^2}{(m-k)^2}.$$
   We note that there are in fact two distinct definitions for $B_k^\dagger$ and hence two definitions for the denominator in $G(k)$; for small enough k, the two are comparable, and the definition we use here is less expensive to calculate [18, §7.4].
3. **L-Curve.**
   To determine the L-curve associated with LSQR or GMRES, estimates of $\|r_k\|_2$ and $\|x_k\|_2$ are needed for several values of k. Using either algorithm, we can compute $\|r_k\|_2$ with only a few scalar calculations. Paige and Saunders give a similar method for computing $\|x_k\|_2$ [29], but, with GMRES, the cost for computing $\|x_k\|_2$ is $O(k^2)$. In using this method or GCV, one must go a few iterations beyond the optimal k in order to verify the optimum [19].

4.2. Regularization by projection plus TSVD. If projection alone does not regularize, then we can compute the TSVD regularized solution to the projected problem (19). We need the SVD of the $(k+1) \times k$ matrix B_k. This requires $O(k^3)$ operations, but can also be computed from the SVD of B_{k-1} in $O(k^2)$ operations [13].

Clearly, we still need to use some type of parameter selection technique to find a good value of $\ell(k)$. First, notice that it is easy to compute the norms of the residual and the solution resulting from neglecting the ℓ smallest singular values. If ξ_{jk} is the component of e_1 in the direction of the j-th left singular vector of B_k, and if γ_j is the j-th singular value (ordered largest to smallest), then the residual and solution 2-norms are

$$\beta_1 \left(\sum_{j=k-\ell(k)+1}^{k+1} \xi_{jk}^2 \right)^{1/2} \quad \text{and} \quad \beta_1 \left(\sum_{j=1}^{k-\ell(k)} \left(\frac{\xi_{jk}}{\gamma_j} \right)^2 \right)^{1/2} . \tag{20}$$

Using this fact, we can use any of our three sample methods:

1. **Discrepancy Principle.**
   Let $r_\ell^{(k)}$ denote the quantity $b - Ax_\ell^{(k)}$ and note that by (13) and orthonormality, $\|r_\ell^{(k)}\|_2$ is equal to the first quantity in (20). Therefore, we choose $\ell(k)$ to be the largest value for which $\|r_\ell^{(k)}\| \leq \tau\delta$, if such a value exists.

2. **GCV.**
Another alternative for choosing $\ell(k)$ is to use GCV to compute $\ell(k)$ for the projected problem. The GCV functional for the kth projected problem is obtained by substituting B_k for A and $B_k^\sharp$ for $A^\sharp$, and substituting the expression of the residual in (20) for the numerator in (11):
$$G_k(\ell) = \frac{\beta_1^2 \sum_{j=k-\ell+1}^{k+1} \xi_{jk}^2}{(\ell+1)^2}.$$
3. **L-Curve.**
We now have many L-curves, one for each value of k. The coordinate values in (20) form the discrete L-curve for a given k, from which the desired value of $\ell(k)$ can be chosen without forming the approximate solutions or residuals.

As k increases, the value $\ell(k)$ chosen by the Discrepancy Principle will be monotonically nondecreasing.

4.3. Regularization by projection plus Rust's TSVD. As in standard TSVD, to use Rust's version of TSVD for regularization of the projected problem requires that we compute the SVD of the $(k+1)\times k$ matrix B_k. Using the previous notation, Rust's strategy is to set
$$y_\rho^{(k)} = \sum_{i\in\mathcal{I}_\rho^{(k)}} \frac{\xi_{ik}}{\gamma_i} q_i^{(k)}$$
where $q_j^{(k)}$ are the right singular vectors of B_k and $\mathcal{I}_\rho^{(k)} = \{i < k+1 : |\xi_{ik}| > \rho\}$. We focus on three ways to determine ρ:

1. **Discrepancy Principle.**
Using the notation from the previous section, the norm of the regularized solution is given by $\beta_1(\sum_{i\notin\mathcal{I}_\rho^{(k)}} \xi_{ik}^2)^{1/2} = \|r_\rho^{(k)}\|_2$. According to the discrepancy principle, we must choose ρ so that the residual is less than $\tau\delta$. In practice, this would require that the residual be evaluated by sorting the values $|\xi_{ik}|$ and adding terms in that order until the residual norm is less than $\tau\delta$.
2. **GCV.**
Let us denote by $\mathrm{card}(\mathcal{I}_\rho^{(k)})$ the cardinality of the set $\mathcal{I}_\rho^{(k)}$. From (11), it is easy to show that the GCV functional corresponding to the projected problem for this regularization technique is given by
$$G_k(\rho) = \frac{\beta_1^2 \sum_{i\in\mathcal{I}_\rho^{(k)}} \xi_{ik}^2}{(k+1-\mathrm{card}(\mathcal{I}_\rho^{(k)}))^2}.$$
In practice, for each k we first sort the values $|\xi_{ik}|, i = 1,\ldots,k$ from smallest to largest. Then we define k discrete values ρ_j to be equal to these values with ρ_1 being the smallest. We set $\rho_0 = 0$. Note that because the values of $\rho_j, j = 1,\ldots,k$ are the sorted magnitudes of the SVD expansion coefficients, we have
$$G_k(\rho_j) = \frac{\beta_1^2(|\xi_{(k+1),k}|^2 + \sum_{i=1}^{j} \rho_j^2)}{(j+1)^2}, \quad j = 0,\ldots,k.$$
Finally, we take the regularization parameter to be the ρ_j for which $G_k(\rho_j)$ is a minimum.

3. **L-Curve.**
As with standard TSVD, we now have one L-curve for each value of k. For fixed k, if we define the $\rho_j, j = 0, \ldots, k$ as we did for GCV above and we reorder the γ_i in the same way that the $|\xi_{ik}|$ were reordered when sorted, then we have

$$\|x_{\rho_j}^{(k)}\|_2^2 = \beta_1^2 \sum_{i=j+1}^{k} \left(\frac{\rho_i}{\gamma_i}\right)^2 \quad ; \quad \|r_{\rho_j}^{(k)}\|_2^2 = \beta_1^2(|\xi_{(k+1),k}|^2 + \sum_{i=1}^{j} \rho_j^2) \; j = 0, \ldots, k.$$

When these solution and residual norms are plotted against each other as functions of ρ, the value of ρ_j corresponding to the corner is selected as the regularization parameter.

4.4. Regularization by projection plus Tikhonov. Finally, let us consider using Tikhonov regularization to regularize the projected problem (18) for some integer k. Thus, for a given regularization parameter λ, we would like to solve

$$\min_y \|\beta_1 e_1 - B_k y\|_2^2 + \lambda^2 \|y\|_2^2, \tag{21}$$

or, equivalently,

$$\min_y \| \begin{bmatrix} \beta_1 e_1 \\ 0 \end{bmatrix} - \begin{bmatrix} B_k \\ \lambda I \end{bmatrix} y \|_2. \tag{22}$$

The solution $y_\lambda^{(k)}$ to either formulation satisfies

$$(B_k^* B_k + \lambda^2 I) y_\lambda^{(k)} = \beta_1 B_k^* e_1. \tag{23}$$

Using (13) and (15), we see that $y_\lambda^{(k)}$ also satisfies

$$(V_k^* A^* A V_k + \lambda^2 I) y_\lambda^{(k)} = V_k^* A^* b. \tag{24}$$

Therefore,

$$y_\lambda^{(k)} = \operatorname{argmin}_y \| \begin{bmatrix} A \\ \lambda I \end{bmatrix} V_k y - \begin{bmatrix} b \\ 0 \end{bmatrix} \|_2.$$

Using $x_\lambda^{(k)} = V_k y_\lambda^{(k)}$, we have

$$x_\lambda^{(k)} = \operatorname{argmin}_{x \in S} \|Ax - b\|_2^2 + \lambda^2 \|x\|_2^2.$$

Thus as $k \to n$, the backprojected regularized solution $x_\lambda^{(k)}$ approaches the solution to (4).

We need to address how to choose a suitable value of λ.

1. **Discrepancy Principle.**
Note that in exact arithmetic, we have

$$r_\lambda^{(k)} = b - A x_\lambda^{(k)} = U_{k+1}^* (\beta_1 e_1 - B_k y_\lambda^{(k)}). \tag{25}$$

Hence $\|B_k y_\lambda^{(k)} - \beta_1 e_1\|_2 = \|r_\lambda^{(k)}\|_2$. Therefore, to use the Discrepancy Principle requires we choose λ so that $\|r_\lambda^{(k)}\|_2 \le \tau\delta$, with p discrete trial values λ_j. For a given k, we take λ to be the largest value λ_j for which $\|r_\lambda^{(k)}\|_2 < \tau\delta$, if it exists; if not, we increase k and test again.

2. **GCV.**
Let us define $(B_k)^\dagger_\lambda$ to be the operator mapping the right hand side of the projected problem onto the regularized solution of the projected problem:

$$(B_k)^\dagger_\lambda = (B_k^* B_k + \lambda^2 I)^{-1} B_k^*.$$

Given the SVD of B_k as above, the denominator in the GCV functional defined for the projected problem (refer to (11)) is

$$\left(k + 1 - \sum_{j=1}^{k} \frac{\gamma_j^2}{\gamma_j^2 + \lambda^2} \right)^2 .$$

The numerator is simply $\|r_\lambda^{(k)}\|_2^2$. For values of $k \ll n$, it is feasible to compute the singular values of B_k.
3. **L-Curve.**
The L-curve is comprised of the points $(\|B_k y_\lambda^{(k)} - \beta_1 e_1\|_2, \|y_\lambda^{(k)}\|_2)$. But using (25) and the orthonormality of the columns of V_k, we see these points are precisely $(\|r_\lambda^{(k)}\|_2, \|x_\lambda^{(k)}\|_2)$. For p discrete values of λ, $\lambda_i, 1 \le i \le p$, the quantities $\|r_{\lambda_i}^{(k)}\|_2$ and $\|x_{\lambda_i}^{(k)}\|_2$ can be obtained by updating their respective estimates at the $(k-1)$st iteration.[2]

4.5. Correspondence between Direct Regularization and Projection Plus Regularization. In this section, we argue why the projection plus regularization approaches can be expected to yield regularized solutions nearly equivalent to the direct regularization counterpart. The following theorem establishes the desired result for the case of Tikhonov vs. projection plus Tikhonov.

THEOREM 4.1. *Fix $\lambda > 0$ and define $x_\lambda^{(k)}$ to be the kth iterate of conjugate gradients applied to the Tikhonov problem*

$$(A^*A + \lambda^2 I)x = A^*b.$$

Let $y_\lambda^{(k)}$ be the exact solution to the regularized projected problem

$$(B_k^* B_k + \lambda^2 I)y = B_k^*(\beta e_1)$$

where B_k, V_k are derived from the original problem $A^*A = A^*b$, and set $z_\lambda^{(k)} = V_k y_\lambda^{(k)}$.

Then $z_\lambda^{(k)} = x_\lambda^{(k)}$.

Proof: By the discussion at the beginning of §4.4 and equations (23) and (24), it follows that $y_\lambda^{(k)}$ solves

$$V_k^*(A^*A + \lambda^2 I)V_k y = V_k^* A^* b.$$

Now the columns of V_k are the Lanczos vectors with respect to the matrix A^*A and right-hand side A^*b. But these are the same as the Lanczos vectors generated with respect to the matrix $A^*A + \lambda^2 I$ and right-hand side A^*b. Therefore $V_k y_\lambda^{(k)}$ is precisely the kth iterate of conjugate gradients applied to $(A^*A + \lambda^2 I)x = A^*b$ [11, pg. 495]. Hence $z_\lambda^{(k)} = x_\lambda^{(k)}$. □

---

[2] The technical details of the approach are found in [28, pp. 197-198], from which we obtain $\|r_\lambda^{(k)}\| = \sqrt{\|\bar{r}_\lambda^{(k)}\|^2 + \lambda^2 \|x_\lambda^{(k)}\|^2}$. The implementation details for estimating $\|x_\lambda^{(k)}\|$ and $\|\bar{r}_\lambda^{(k)}\|$ were taken from the Paige and Saunders algorithm at `http://www.netlib.org/linalg/lsqr`.

Projection plus –	Disc.	GCV	L-curve
Tikhonov	$O(pk)$	$O(k^3)$	$O(pk)$
TSVD	$O(k^3)$	$O(k^3)$	$O(k^3)$
Rust's	$O(k^3)$	$O(k^3)$	$O(k^3)$

TABLE 2

Summary of flops for projection plus inner regularization with various parameter selection techniques, in addition to the $O(qk)$ flops required for projection itself. Here k is the number of iterations (ie. the size of the projection) taken and p is the number of discrete parameters that must be tried.

Let us turn to the case of TSVD regularization applied to the original problem vs. the projection plus TSVD approach. Direct computation convinces us that the two methods compute the same regularized solution if $k = n$ and arithmetic is exact. An approximate result holds in exact arithmetic when we take k iterations, with $n - \ell = j < k < n$. Let the singular value decomposition of B_k be denoted by $B_k = Z_k \Gamma_k Q_k^T$ and define the $s \times j$ matrix $W_{s,j}$ as

$$W_{s,j} = \begin{bmatrix} I \\ 0 \end{bmatrix}.$$

Then the regularized solution obtained from the TSVD regularization of the projected problem is

$$x_{reg}^{(k)} = V_k (Q_k W_{k,j} \Gamma_{k,1}^{-1} W_{k+1,j}^T Z_k^T U_k^T b),$$

where $\Gamma_{k,1}$ denotes the leading $j \times j$ principle submatrix of Γ_k. If k is taken to be enough larger than j so that $V_k Q_k W_{k,j} \approx \hat{V} W_{n,j}$, $W_{k+1,j}^T Z_k^T U_{k+1}^T \approx W_{n,j}^T \hat{U}^T$ and $\Gamma_{k,1} \approx \Sigma_1$ with Σ_1 the leading principle submatrix of Σ, then we expect $x_{reg}^{(k)}$ to be a good approximation to x_ℓ. This is made more precise in the following theorem.

THEOREM 4.2. *Let $k > j$ such that*

$$(V_k Q_k W_{k,j}) = \hat{V}_1 + E_1 \text{ with } \|E_1\| \le \delta_1 \ll 1;$$

$$(U_{k+1} Z_k W_{k+1,j}) = \hat{U}_1 + E_2 \text{ with } \|E_2\| \le \delta_2 \ll 1;$$

where $\hat{V}_1$ and $\hat{U}_1$ contain the first j columns of $\hat{V}$ and $\hat{U}$ respectively. Let $D = diag(d_1, \ldots, d_j)$ satisfy

$$\Gamma_{k,1} = \Sigma_1 + D \text{ with } |d_i| \le \delta_3 \ll 1.$$

Then

$$\|x_{reg}^{(k)} - x_\ell\| \le \max_{1 \le i \le j} \frac{1}{\sigma_i + d_i} \left(\frac{\delta_3}{\sigma_j} + 3 \max(\delta_1, \delta_2) \right) \|b\|.$$

Proof: Using the representations $x_\ell = \hat{V}_1 \Sigma_1^{-1} \hat{U}_1^T b$ and $x_{reg}^{(k)} = (\hat{V}_1 + E_1) \Gamma_{k,1}^{-1} (\hat{U}_1^T + E_2^T) b$, we obtain

$$\|x_{reg}^{(k)} - x_\ell\| \le (\|\Gamma_{k,1}^{-1} - \Sigma_1^{-1}\| + \|\Gamma_{k,1}^{-1}\| \, \|E_2\| + \|E_1\| \, \|\Gamma_{k,1}^{-1}\| + \|E_1\| \, \|\Gamma_{k,1}^{-1}\| \, \|E_2\|) \|b\|,$$

and the conclusion follows from bounding each term. □

Note that typically $\sigma_j \gg \sigma_n$ so that $1/\sigma_j$ is not too large. For some results relating to the value of k necessary for the hypothesis of the theorem to hold, the interested reader is referred to theory of the Kaniel-Paige and Saad [30, §12.4].

	Basic cost	Added Cost		
		Disc.	GCV	L-curve
Tikhonov	$O(\hat{q})$	$O(1)$	$O(p)$	$O(p)$
TSVD	$O(\hat{q})$	$O(1)$	$O(m)$	$O(m)$
Rust's TSVD	$O(\hat{q})$	$O(m)$	$O(m)$	$O(m)$
Projection	$O(kn)$	$O(1)$	$O(k)$	$O(k)$

TABLE 3

Summary of additional storage for each of four regularization methods under each of three parameter selection techniques. The original matrix is $m \times n$ with q nonzeros, p is the number of discrete parameters that must be tried, k iterations are used in projection, and the factorizations are assumed to take $\hat{q}$ storage.

Projection plus –	Disc.	GCV	L-curve
Tikhonov	$O(1)$	$O(p)$	$O(p)$
TSVD	$O(1)$	$O(k)$	$O(k)$
Rust's TSVD	$O(k)$	$O(k+p)$	$O(k+p)$

TABLE 4

Summary of storage, not including storage for the matrix, for projection plus inner regularization approach, various parameter selection techniques. Here p denotes the number of discrete parameters tried. Each of these regularization methods also requires us to save the basis V or else regenerate it in order to reconstruct x.

5. Numerical results. In this section, we present two numerical examples. All experiments were carried out using Matlab and Hansen's Regularization Tools [16], with IEEE double precision floating point arithmetic. Since the exact, noise-free solutions were known in both examples, we evaluated the methods using the two-norm difference between the regularized solutions and the exact solutions. In both examples when we applied Rust's method to the original problem, the ρ_i were taken to be the magnitudes of the spectral coefficients of b sorted in increasing order.

5.1. Example 1. The 200×200 matrix A and true solution x_{true} for this example were generated using the function `baart` in Hansen's Regularization Toolbox. We generated $b_{true} = Ax_{true}$ and then computed the noisy vector b as $b + e$, where e was generated using the Matlab `randn` function and was scaled so that the noise level, $\frac{\|e\|}{\|b_{true}\|}$, was 10^{-3}. The condition number of A was on the order of 10^{19}.

Many values of λ were tested: $\log_{10}\lambda = -6, -5.9, \ldots, -2$. Table 5 displays the values of the regularization parameters chosen when the three parameter selection techniques were applied together with one of the four regularization methods on the original problem. Since $\|e\|_2 = 5.3761\text{E}{-4}$, we set $\tau\delta$ that defines the discrepancy principle as the very close approximation $5.5\text{E}{-4}$.

The last column in the table gives the value of the parameter that yielded a regularized solution with the minimum relative error when compared against the true solution. The relative error values for regularized solutions corresponding to the parameters in Table 5 are given in Table 6. Note that using GCV to determine a regularization parameter for Rust's TSVD resulted in an extremely noisy solution with huge error.

The corners of the L-curves for the Tikhonov, projection, and TSVD methods were determined using Hansen's `lcorner` function, with the modification that points corresponding to solution norms greater than 10^6 for the TSVD methods were not

	Disc.	GCV	L-curve	optimal
Tikhonov λ	1.259E−3	1.995E−4	2.512E−4	5.012E−5
TSVD ℓ	197	197	196	196
Rust's TSVD ρ	1.223E−4	9.645E−7	1.223E−4	1.259E−4 or 1.223E−4
Projection k	4	4	6	6

TABLE 5

Example 1: parameter values selected for each method.

	Disc.	GCV	L-curve	optimal
Tikhonov	.1330	.1110	.1084	.0648
TSVD	.1663	.1213	.1663	.1213
Rust's TSVD	.1213	7E+14	.1213	.1213
Projection	.1134	.1207	.1134	.1134

TABLE 6

Example 1: comparison of $||x_{true} - x_{reg}||_2/||x_{true}||_2$ for each of 4 regularization methods on the original problem, where the regularization method was chosen using methods indicated.

considered (otherwise, a false corner resulted).

Next, we projected using LSQR and then regularized the projected problem with one of the three regularization methods considered. For each of the three methods, we computed regularization parameters for the projected problem using Discrepancy, GCV, and L-curve, then computed the corresponding regularized solutions; the parameters that were selected in each case at iterations 10 and 40 are given in Tables 7 and 9 respectively. As before, the `lcorner` routine was used to determine the corners of the respective L-curves.

Comparing Table 6 and 8, we observe that computing the regularized solution via projection plus Tikhonov for projection size of 10 using either the Discrepancy Principle or the L-curve to find the regularization parameter gives results as good as if those techniques had been used with Tikhonov on the original problem to determine a regularized solution. Similar statements can be made for projection plus TSVD and projection plus Rust's TSVD. We should also note that for Tikhonov, with and without projection, none of the errors in the tables is optimal; that is, no parameter selection techniques ever gave the parameter for which the error was minimal.

5.2. Example 2. The 255×255 matrix A for this example was a symmetric Toeplitz matrix with bandwidth 16 and exponential decay across the band.[3] The true solution vector x_{true} is displayed as the top picture in Figure 2. We generated $b_{true} = Ax_{true}$ and then computed the noisy vector b as $b + e$, where e was generated using the Matlab `randn` function and was scaled so that the noise level, $\frac{||e||}{||b_{true}||}$, was 10^{-3}. The vector b is shown in the bottom of Figure 2. The condition number of A was 1.65×10^7.

We generated our discrete λ_i using $\log_{10} \lambda = -5, -4.9, \ldots, -1$. The norm of the noise vector was 7.16E−2, so we took the value of $\tau\delta$ that defines the discrepancy principle to be 8.00E−2.

In this example, it took 61 iterations for LSQR to reach a minimum relative error of 9.48E−2, and several more iterations were needed for the L-curve method to

[3] It was generated using the Matlab command $A = (1/(2 * pi * sigma)) * toeplitz(t)$, where $sigma = 5$ and $t = [exp(-([0 : band - 1]. \wedge 2)/(2 * sigma \wedge 2)), zeros(1, N - band)]$ with $band = 16$.

	Disc.	GCV	L-curve	optimal
Tikhonov $\lambda(k)$	1.259E−3	1.995E−3	1.995E−4	5.012E−5
TSVD $\ell(k)$	2	3	2	2
Rust's TSVD $\rho(k)$	1.679E−4	1.773E−4	1.679E−5	1.679E−5

TABLE 7

Example 1, iteration 10: regularization parameters selected for projection plus Tikhonov, TSVD, and Rust's TSVD.

	Disc.	GCV	L-curve	optimal
Tikhonov	.1330	.1486	.1084	.0648
TSVD	.1663	.3451	.1663	.1213
Rust's TSVD	.1213	.1663	.1213	.1213

TABLE 8

Example 1, iteration 10: comparison of $\|x_{true} - x_{reg}\|_2/\|x_{true}\|_2$ *for projection plus Tikhonov, TSVD, and Rust's TSVD.*

	Disc.	GCV	L-curve	optimal
Tikhonov $\lambda(k)$	1.259E−3	1.995E−3	1.995E−4	5.012E−5
TSVD $\ell(k)$	10	13	8	9
Rust's TSVD $\rho(k)$	9.201E−5	1.225E−4	9.201E−5	9.201E−5

TABLE 9

Example 1, iteration 40: regularization parameters selected for projection plus Tikhonov, TSVD, and Rust's TSVD.

	Disc.	GCV	L-curve	optimal
Tikhonov	.1330	.1486	.1084	.0648
TSVD	.1679	.1986	.1206	.1165
Rust's TSVD	.1162	.1162	.1162	.1162

TABLE 10

Example 1, iteration 40: comparison of $\|x_{true} - x_{reg}\|_2/\|x_{true}\|_2$ *for projection plus Tikhonov, TSVD, and Rust's TSVD.*

FIG. 2. *Example 2: Top: exact solution. Bottom: noisy right hand side b.*

	Disc.	GCV	L-curve	optimal
Tikhonov λ	1.259E−2	1.259E−2	1.995E−3	3.9811E−3
TSVD ℓ	216	254	201	201
Rust's TSVD ρ	2.183E−2	2.586E−6	1.477E−2	1.527E−2
Projection k	2	18	5	5

TABLE 11

Example 2: parameter values selected for each method. The projection was performed on a left preconditioned system.

	Disc.	GCV	L-curve	optimal
Tikhonov	9.909E−2	9.909E−2	1.050E−2	9.394E−2
TSVD	1.102E−1	8.121E−1	9.074E−2	9.0744E−2
Rust's TSVD	1.025E−1	22.67	1.011E−1	1.011E−1
Projection	1.030E−1	9.85E−2	1.15E−1	9.479E−2

TABLE 12

Example 2: comparison of $||x_{true} - x_{reg}||_2/||x_{true}||_2$ *for each of 4 regularization methods on the original problem.*

estimate a stopping parameter. Likewise, the dimension k of the projected problem had to be around 60 to obtain good results with the projection-plus-regularization approaches, and much larger than 60 for the L-curve applied to the projected, Tikhonov regularized problem to give a good estimate of the corner with respect to the Tikhonov regularized original problem. Therefore, for the projection based techniques, we chose to work with a left preconditioned system (refer to the discussion at the end of § 3.5). Our preconditioner was chosen as in [22] where the parameter defining the preconditioner was taken to be $m = 50$.

The values of the regularization parameters chosen when the three parameter selection techniques were applied together with one of the four regularization methods on the original problem are given in Table 11. The last column in the table gives the value of the parameter that gave a regularized solution with the minimum relative error over the range of discrete values tested, with respect to the true solution. The relative errors that resulted from computing solutions according to the parameters in Table 11 are in Table 12. We note that GCV with TSVD and Rust's TSVD were ineffective.

The corners of the L-curves for the Tikhonov, projection, and TSVD methods were determined using Hansen's `lcorner` function, with the modification that points corresponding to the largest solution norms for the TSVD methods were not considered (otherwise, a false corner was detected by the `lcorner` routine).

Next, we projected using LSQR (note that since the matrix and preconditioner were symmetric, we could have used MINRES as in [22]) and then regularized the projected problem with one of the three methods considered. For each of the three methods, we computed regularization parameters for the projected problem using Discrepancy, GCV, and L-curve, then computed the corresponding regularized solutions; the parameters that were selected in each case at iterations 15 and 25 are given in Tables 13 and 15, respectively. The relative errors of the regularized solutions generated accordingly are given in Tables 14 and 16.

Again, we used the `lcorner` routine to determine the corners of the respective L-curves, except in the case of Rust's TSVD method. In the latter case, there was

	Disc.	GCV	L-curve	optimal
Tikhonov $\lambda(k)$	2.512E−2	1.585E−2	1.9953E−3	3.981E−3
TSVD $\ell(k)$	5	5	4	1
Rust's TSVD $\rho(k)$	3.558E−2	3.558E−2	3.558E−2	3.558E−2

TABLE 13

Example 2, iteration 15: regularization parameters selected for projection plus Tikhonov, TSVD, and Rust's TSVD.

	Disc.	GCV	L-curve	optimal
Tikhonov	1.0001E−1	9.9511E−2	1.061E−1	9.530E−2
TSVD	9.595E−1	9.595E−1	1.004E−1	9.357E−2
Rust's TSVD	1.004E−1	1.004E−1	1.004E−1	1.004E−1

TABLE 14

Example 2, iteration 15: comparison of $||x_{true} - x_{reg}||_2/||x_{true}||_2$ *for projection plus Tikhonov, TSVD, and Rust's TSVD.*

always a very sharp corner that could be picked out visually.

Comparing Table 11 with Tables 13 and 15, we see that the parameter chosen by applying the L-curve method to projected-plus-Tikhonov problem was the same parameter chosen by applying the L-curve to the original problem. Moreover, a comparison of Table 12 with Tables 14 and 16 shows that relative errors of the regularized solutions computed accordingly are comparable to applying Tikhonov to the original problem with that same parameter. Similar results are shown for the other cases, with the exception that the discrepancy principle did not work well for the projection-plus-TSVD problems, and GCV was not effective for the projected problems when $k = 25$.

6. Conclusions. In this work we have given methods for determining the regularization parameter and regularized solution to the original problem based on regularizing a projected problem. The proposed approach of applying regularization and parameter selection techniques to a projected problem is economical in time and storage. We presented results that in fact the regularized solution obtained by backprojecting the TSVD or Tikhonov solution to the projected problem is almost equivalent to applying TSVD or Tikhonov to the original problem, where "almost" depends on the size of k. The examples indicate the practicality of the method, and illustrate that our regularized solutions are usually as good as those computed using the original system and can be computed in a fraction of the time, using a fraction of the storage. We note that similar approaches are valid using other Krylov subspace methods for computing the projected problem.

In this work, we did not address potential problems from loss of orthogonality as the iterations progress. In this discussion, we did, however, assume that either k was naturally very small compared to n or that preconditioning had been applied to enforce this condition. Possibly for this reason, we found that for modest k, round-off did not appear to degrade either the LSQR estimates of the residual and solution norms or the computed regularized solution in the following sense: the regularization parameters chosen via the projection-regularization and the corresponding regularized solutions were comparable to those chosen and generated for the original discretized problem.

For the Tikhonov approach in this paper, we have assumed that the regularization

	Disc.	GCV	L-curve	optimal
Tikhonov $\lambda(k)$	2.512E−2	1.259E−2	1.995E−3	3.982E−3
TSVD $\ell(k)$	9	9	8	3
Rust's TSVD $\rho(k)$	4.828E−2	7.806E−3	4.828E−2	4.828E−2

TABLE 15

Example 2, iteration 25: regularization parameters selected for projection plus Tikhonov, TSVD, and Rust's TSVD.

	Disc.	GCV	L-curve	optimal
Tikhonov	1.000E−1	9.909E−2	1.061E−1	9.530E−2
TSVD	9.164E−1	9.595E−1	1.004E−1	9.145E−2
Rust's TSVD	1.004E−1	2.728	1.004E−1	1.004E−1

TABLE 16

Example 2, iteration 25: comparison of $||x_{true} - x_{reg}||_2/||x_{true}||_2$ *for projection plus Tikhonov, TSVD, and Rust's TSVD.*

operator L was the identity or was related to the preconditioning operator; this allowed us to efficiently compute $||r_\lambda^{(k)}||$ and $||x_\lambda^{(k)}||$ for multiple values of λ efficiently for each k. If L is not the identity but is invertible, we can first implicitly transform the problem to "standard form" [17]. With $\bar{A} = AL^{-1}$, $\bar{x} = Lx$, we can solve the equivalent system

$$min_{\bar{x}} = ||\bar{A}\bar{x} - b||_2^2 + \lambda^2 ||\bar{x}||_2^2.$$

Then the projection plus regularization schemes may be applied to this transformed problem. Clearly the projection based schemes will be useful as long as solving systems involving L can be done efficiently.

REFERENCES

[1] A. Björck, *A bidiagonalization algorithm for solving large and sparse ill-posed systems of linear equations*, BIT, 28 (1988), pp. 659–670.

[2] A. Björck, E. Grimme, and P. V. Dooren, *An implicit shift bidiagonalization algorithm for ill-posed systems*, BIT, 34 (1994), pp. 510–534.

[3] D. Calvetti, G. Golub, and L. Reichel, *Estimation of the L-curve via Lanczos bidiagonalization*, Tech. Report SCCM-97-12, Scientific Computing and Computational Mathematics Program, Computer Science Depart., Stanford University, 1997.

[4] T. Chan and M. Ng, *Galerkin projection method for solving multiple linear systems*, Tech. Report 96-31, Computational and Applied Mathematics Dept., UCLA, Los Angeles, CA, 1996.

[5] L. Desbat and D. Girard, *The 'minimum reconstruction error' choice of regularization parameters: Some more efficient methods and their application to deconvolution problems*, SIAM J. Sci. Comput., 16 (1995), pp. 1387–1403.

[6] H. W. Engl and W. Grever, *Using the L-curve for determining optimal regularization parameters*, Numer. Math., 69 (1994), pp. 25–31.

[7] H. E. Fleming, *Equivalence of regularization and truncated iteration in the solution of ill-posed image reconstruction problems*, Lin. Alg. and Applics., 130 (1990), pp. 133–150.

[8] A. Frommer and P. Maass, *Fast CG-based methods for Tikhonov-Phillips regularization*, SIAM J. Sci. Comput., (to appear).

[9] G. Golub, M. Heath, and G. Wahba, *Generalized cross-validation as a method for choosing a good ridge parameter*, Technometrics, 21 (1979), pp. 215–223.

[10] G. Golub and W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal., 2(Series B) (1965), pp. 205–224.

[11] G. Golub and C. V. Loan, *Matrix Computations*, The Johns Hopkins University Press, Baltimore, MD, 1989.

[12] W. Groetsch, *Theory of Tikhonov Regularization for Fredholm equations of the First Kind*, Pitman Publishing Limited, 1984.

[13] M. Gu and S. Eisenstat, *A stable and fast algorithm for updating the singular value decomposition*, Tech. Report RR-939, Yale University, Department of Computer Science, New Haven, CT, 1993.

[14] M. Hanke, J. Nagy, and R. Plemmons, *Preconditioned iterative regularization for ill-posed problems*, Numerical Linear Algebra and Sci. Computing, (1993), pp. 141–163. L. Reichel, A. Ruttan, and R. S. Varga, editors.

[15] P. C. Hansen, *Analysis of discrete ill-posed problems by means of the L-curve*, SIAM Review, 34 (1992), pp. 561–580.

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

[17] ———, *Rank-Deficient and Discrete Ill-Posed Problems*, SIAM Press, Philadelphia, 1998.

[18] ———, *Rank Deficient and Discrete Ill-Posed Problems*, PhD thesis, Technical University of Denmark, July 1995. UNIC Report UNIC-95-07.

[19] P. C. Hansen and D. P. O'Leary, *The use of the L-curve in the regularization of discrete ill-posed problems*, SIAM J. Sci. Comput., 14 (1993), pp. 1487–1503.

[20] B. Hofmann, *Regularization for Applied Inverse and Ill-Posed Problems*, Teubner-Texte Mathe., Teubner, Leipzig, 1986.

[21] L. Kaufman and A. Neumaier, *Regularization of ill-posed problems by envelope guided conjugate gradients*, Journal of Computational and Graphical Statistics, (1997).

[22] M. Kilmer, *Symmetric Cauchy-like preconditioners for the regularized solution of 1-d ill-posed problems*, Tech. Report CS-TR-3851, University of Maryland, College Park, 1997.

[23] ———, *Cauchy-like preconditioners for 2-dimensional ill-posed problems*, SIAM J. Matrix Anal., (1998). to appear.

[24] M. Kilmer and D. P. O'Leary, *Pivoted Cauchy-like preconditioners for regularized solution of ill-posed problems*, SIAM J. Sci. Stat. Comput., (1998). to appear.

[25] V. A. Morozov, *On the solution of functional equations by the method of regularization*, Soviet Math. Dokl., 7 (1966), pp. 414–417. cited in [17].

[26] J. Nagy, R. Plemmons, and T. Torgersen, *Iterative image restoration using approximate inverse preconditioning*, IEEE Trans. Image Proc., 5 (96), pp. 1151–1163.

[27] D. P. O'Leary and J. A. Simmons, *A bidiagonalization-regularization procedure for large scale discretization of ill-posed problems*, SIAM J. Sci. Stat. Comput., 2 (1981), pp. 474–489.

[28] C. C. Paige and M. A. Saunders, *Algorithm 583, LSQR: Sparse linear equations and least squares problems*, ACM Transactions on Mathematical Software, 8 (1982), pp. 43–71.

[29] ———, *LSQR: An algorithm for sparse linear equations and sparse least squares*, ACM Transactions on Mathematical Software, 8 (1982), pp. 43–71.

[30] B. N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice-Hall, Englewood Cliffs, NJ, 1980.

[31] B. W. Rust, *Truncating the singular value decomposition for ill-posed problems*, Tech. Report NISTIR 6131, Mathematical and Computational Sciences Division, National Institute of Standards and Technology, Gaithersburg, MD, 1998.

[32] Y. Saad, *Iterative Methods for Sparse Linear Systems*, PWS Publishing Company, Boston, 1996.

[33] Y. Saad and M. H. Schultz, *GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems*, SIAM J.Sci. Stat. Comput., 7 (1986), pp. 856–869.

[34] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*, Wiley, New York, 1977.

[35] J. M. Varah, *Pitfalls in the numerical solution of linear ill-posed problems*, SIAM J. Sci. Stat. Comput., 4 (1983), pp. 164–176.

[36] C. R. Vogel, *Non-convergence of the L-curve regularization parameter selection method*, Inverse Problems, 12 (1996), pp. 535–547.