Extension of GKB-FP algorithm to large-scale general-form Tikhonov regularization

Fermín S. Viloche Bazán [*], Maria C. C. Cunha and Leonardo S. Borges [†]

Department of Mathematics, Federal University of Santa Catarina,
88040-900, Florianópolis SC, Brazil
e-mail: fermin@mtm.ufsc.br
Department of Applied Mathematics, IMECC-UNICAMP,
University of Campinas, CP 6065, 13081-970, Campinas SP, Brazil
e-mail:lsbplsb@yahoo.com

Abstract

In a recent paper [24] an algorithm for large-scale Tikhonov regularization in standard form called GKB-FP was proposed and numerically illustrated. In this paper, further insight into the convergence properties of this method is provided and extensions to general-form Tikhonov regularization are introduced. In addition, as alternative to Tikhonov regularization, a preconditioned LSQR method coupled with an automatic stopping rule is proposed. Preconditioning seeks to incorporate smoothing properties of the regularization matrix into the computed solution. Numerical results are reported to illustrate the methods on large-scale problems.

KEY WORDS: Tikhonov regularization, Large-scale problems, Ill-posed problems

1 Introduction

We are concerned with the computation of stable solutions to large-scale discrete ill-posed problems of the form

$$x = \underset{x\in\mathbb{R}^n}{\operatorname{argmin}} \|b - Ax\|_2^2 \tag{1}$$

where the matrix $A \in \mathbb{R}^{m\times n}$, $m \geq n$, is severely ill-conditioned. Matrices of this type arise, for example, when discretizing first kind integral equations with smooth kernel, e.g., in signal processing and image restoration. In applications the vector b represents the data and is assumed to be of the form $b = b^{\text{exact}} + e$, where e denotes a noise vector due to measurement or approximation errors, b^{exact} denotes the unknown error-free data, and $x^{\text{exact}} = A^{\dagger}b^{\text{exact}}$ (where $A^{\dagger}$ denotes the Moore-Penrose Pseudo Inverse of A) is the noise-free solution of (1). In these cases, due to the noise in the data, the least squares solution $x_{\text{LS}} = A^{\dagger}b$ is dominated by the noise and regularization methods are required to

[*] The work of this author is supported by CNPq, Brazil, grants 308709/2011-0, 477093/2011-6.

[†] This research is supported by FAPESP, Brazil, grant 2009/52193-1.

construct stable approximations to x^{exact}. In Tikhonov regularization [1], we replace x_{LS} by the regularized solution

$$x_\lambda = \underset{x\in\mathbb{R}^n}{\operatorname{argmin}} \left\{ \|b - Ax\|_2^2 + \lambda^2 \|Lx\|_2^2 \right\}, \tag{2}$$

where $\lambda > 0$ is the regularization parameter and $L \in \mathbb{R}^{p\times n}$ is referred to as the regularization matrix. When $L = I_n$, the $n \times n$ identity matrix, the Tikhonov regularization problem is said to be in standard form, otherwise, it is in general form. Solving (2) is equivalent to solve the so-called regularized normal equations

$$(A^TA + \lambda^2 L^TL)x = A^Tb, \tag{3}$$

whose solution x_λ is unique when $\mathcal{N}(A) \cap \mathcal{N}(L) = \{0\}$. In this paper $\mathcal{N}(\cdot)$ denotes the null space of $(\cdot)$ and $\mathcal{R}(\cdot)$ denotes its column space. When A and L are small, the regularized solution can readily be computed using the GSVD of the pair (A, L), and good approximate solutions can be obtained provided that λ and L are properly chosen. There are many Tikhonov parameter choice methods. These include the discrepancy principle (DP) of Morozov [2], which depends on a priori knowledge of the norm of the error e, and the so called noise-level free or heuristic parameter choice rules such as the L-Curve criterion of Hansen and O'Leary [3], the Generalized Cross-Validation (GCV) of Heath, Golub and Wahba [4] and the algorithmic realization of Regińska's rule [5] via the fixed-point (FP) method by Bazán [6], among others.

It is well-known that the regularization parameter selected by DP depends on the norm of the error $\|e\|_2$ and that the error in the corresponding regularized solution converges to zero as $\|e\|_2 \to 0$ (under certain conditions), see, e.g., [7]. Unfortunately, this is not the case with heuristic rules for which the regularization parameter is not a function of the noise level. As a result, heuristic rules are not without difficulties and cannot be successful in all problems. This is in accordance with a famous result of Bakushinski [8], which asserts that parameter choice rules that do not depend explicitly on the noise level should fail, at least for some problems. Nevertheless, noise-level free rules are widely used in real applications, as for example the L-curve method. For discussions and analyses of heuristic rules the reader is referred to [7, 9]. A numerical comparison of many heuristic parameter choice rules can be found in [10], and, recently in [11].

From the practical point of view, all of the above and many other methods from the literature can be readily implemented using the GSVD when A and L are small or of moderate size. However, for large-scale problems the GSVD is computationally demanding, and one can use instead iterative methods such as LSQR or CGLS to approximate x_λ provided that the parameter λ is known a priori. Another way to proceed is to use projection methods based on Lanczos/Arnoldi iterations; these include CGLS/LSQR and the family of minimum residual methods [12, 13, 14, 15, 16, 17, 18, 19], where the number of iterations k plays the role of the regularization parameter and where the choice of the "best" k can be done using the discrepancy principle or L-curve [15, 20]. However, it is well known that DP is likely to fail if the error norm $\|e\|$ is poorly estimated, and that the L-curve criterion works well except possibly when the curve is not L-shaped, see e.g., Morigi [21] or when the curvature of the L-curve has more than a one local maximum, see, e.g., Hansen et al. [20] or [22].

The difficulty in determining reliable stopping criteria for iterative methods can be alleviated by combining them with an inner regularization method, which gives rise to

Hybrid methods. For the case $L = I_n$, two successful hybrid methods that do not require knowledge of $\|e\|_2$ and combine projection over the Krylov subspace generated by the Golub-Kahan bidiagonalization (GKB) method and standard Tikhonov regularization at each iteration, are W-GCV [23] and GKB-FP [24]. Here the difference is on the projected problem: while W-GCV uses weighted GCV as parameter choice rule, GKB-FP uses the FP method. The case $L \neq I_n$ has received less attention and there are few efficient methods for large-scale regularization. An interesting approach that uses a joint bidiagonalization (JBD) procedure applied to the pair (A, L) and minimizes the Tikhonov functional over the generated Krylov subspace is proposed by Kilmer et al. [15]. This method may be infeasible for large-scale problems because the JBD procedure is computationally demanding. More recently, J. Lampe et. al [25], and Reichel et. al [26], propose approximate solutions by minimizing the Tikhonov functional over generalized Krylov subspaces; in both cases the regularization parameter for the projected problem is determined by the discrepancy principle. A related method which also uses the discrepancy principle as parameter choice rule can be found in [27].

In this paper we assume that no estimate of $\|e\|_2$ is available and concentrate on extensions of the GKB-FP algorithm to large-scale general-form Tikhonov regularization. Two approaches are considered. The first approach relies on the observation that the general-form Tikhonov regularization can be transformed into a Tikhonov problem in standard form which can be handled efficiently by GKB-FP; therefore our first approach is to apply GKB-FP to the transformed Tikhonov problem. As for the second approach, it constructs regularized solutions by minimizing the general-form Tikhonov problem over the Krylov subspace generated by the GKB algorithm applied to matrix A, as suggested in [27]. In all cases, the underlying philosophy of GKB-FP is preserved, i.e., the fixed-point method is always used on the projected problem. Also, to overcome possible difficulties associated with the Tikhonov regularization parameter selection problem at each iteration, as done by GKB-FP or W-GCV, we propose an iterative regularization algorithm based on LSQR applied to the least squares problem $\min \|\bar{b} - \bar{A}\bar{x}\|_2$, where $\bar{b}$ and $\bar{A}$ arise from the transformed Tikhonov problem, coupled with a stopping rule that does not require knowledge of the norm of the noise $\|e\|_2$ and that can be regarded as a discrete counterpart of Regińska's rule. Theoretically, we follow a paper by Hanke and Hansen [13], see also [28], where it is shown how to construct regularized solutions to (2) via CGLS/LSQR in such a way that the smoothing properties of L are incorporated into the iterative process, and hence into the computed solution.

The paper is organized as follows. Section 2 presents an analysis of GKB-FP within the framework of projection methods which provides valuable insight into the convergence properties of the algorithm. In Section 3 the extensions of GKB-FP to general-form Tikhonov regularization are presented and discussed. Our alternative to general-form Tikhonov regularization comes in Section 4, and Section 5 contains numerical examples devoted to illustrate the potential of the methods. Conclusions are in Section 6.

2 GKB-FP as a projection method

The purpose of this section is to provide further insight into the convergence properties of GKB-FP. Recall that after $k < n$ steps, the GKB algorithm applied to A with initial vector $b/\|b\|_2$ yields two matrices $U_{k+1} = [u_1, \ldots, u_{k+1}] \in \mathbb{R}^{m\times(k+1)}$ and $V_k = [v_1, \ldots, v_k] \in$

$\mathbb{R}^{n\times k}$ with orthonormal columns, and a lower bidiagonal matrix $B_k \in \mathbb{R}^{(k+1)\times k}$,

$$B_k = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \beta_3 & \ddots & \\ & & \ddots & \alpha_k \\ & & & \beta_{k+1} \end{pmatrix}, \tag{4}$$

such that

$$\beta_1 U_{k+1} e_1 = b = \beta_1 u_1, \tag{5}$$
$$A V_k = U_{k+1} B_k, \tag{6}$$
$$A^T U_{k+1} = V_k B_k^T + \alpha_{k+1} v_{k+1} e_{k+1}^T, \tag{7}$$

where e_i denotes the i-th unit vector in $\mathbb{R}^{k+1}$ [29]. The columns of V_k provide an orthonormal basis for the generated Krylov subspace $\mathcal{K}_k(A^TA, A^Tb)$, which is an excellent choice for use when solving discrete ill-posed problems [30]. GKB-FP combines the GKB algorithm with standard form Tikhonov regularization in $\mathcal{K}_k(A^TA, A^Tb)$, choosing the regularization parameter on the projected problem via the FP method [6]. The FP method relies on an earlier work of Regińska [5], where the regularization parameter is chosen as a minimizer of the function

$$\Psi(\lambda) = \|r_\lambda\|_2^2 \, \|x_\lambda\|_2^{2\mu}, \ \mu > 0, \tag{8}$$

where $r_\lambda = b - Ax_\lambda$. Regińska proved that if the curvature of the L-curve is maximized at $\lambda = \lambda^*$, and if the slope of the L-curve corresponding to $\lambda = \lambda^*$ is $-1/\mu$, then $\Psi(\lambda)$ is minimized at $\lambda = \lambda^*$. However, no method to compute the minimum was done afterward. More recently, Bazán [6] investigated the issue and, with a proper choice of μ, showed that the minimum can be calculated via fixed-point iterations, giving rise to the FP method.

The key idea behind this parameter choice rule can be explained as follows. Let $R_\lambda = (A^TA + \lambda^2 I_n)^{-1}A^T$. Then the error in the regularized solution x_λ can be expressed as $x^{\text{exact}} - x_\lambda = x^{\text{exact}} - R_\lambda\, b^{\text{exact}} - R_\lambda e$, and therefore

$$\|x^{\text{exact}} - x_\lambda\|_2 \le \|x^{\text{exact}} - R_\lambda\, b^{\text{exact}}\|_2 + \|R_\lambda e\|_2 \equiv E_1(\lambda) + E_2(\lambda). \tag{9}$$

The term $E_1(\lambda)$ represents the regularization error and increases with λ. The second term denotes the error caused by the noise and decreases with λ, see, e.g., [31]. Thus, in order to obtain a small error we should balance the terms as in this event the bound is approximately minimized. However, neither $E_1(\lambda)$ nor $E_2(\lambda)$ is available, hence, an alternative is to minimize a model for the bound. The motivation for using $\Psi(\lambda)$ as a model is that minimizing $\log(\Psi(\lambda))$ we minimize the sum of competing terms, since $\|x_\lambda\|_2$ decreases with λ while $\|r_\lambda\|_2$ increases, as illustrated in Figure 1.

The FP method can be summarized as follows:

- Given an initial guess, FP takes $\mu = 1$ as a default value and proceeds by computing the iterates

$$\lambda_{j+1} = \phi(\lambda_j), \ j \ge 0, \quad \text{where} \quad \phi(\lambda) = \sqrt{\mu}\|r_\lambda\|_2/\|x_\lambda\|_2, \ \lambda > 0, \tag{10}$$

until the largest convex fixed-point of ϕ is reached.

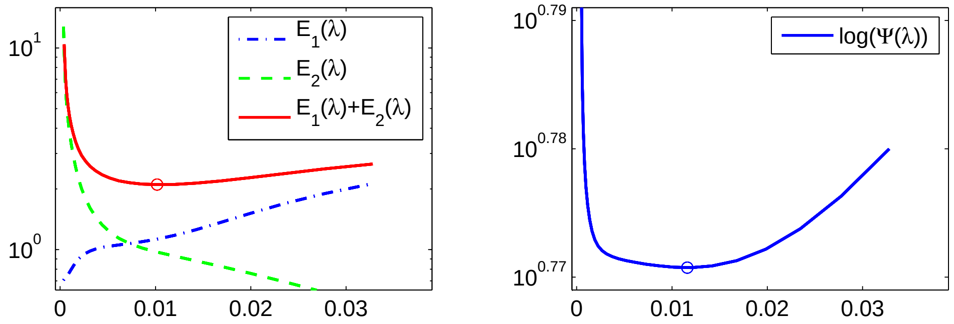

Figure 1: Bound (9) and $\Psi(\lambda)$ for shaw problem from [32], $n = 512$ and $b = b^{\text{exact}} + e$ where e is white noise with $\|e\|_2 = 0.005\|b^{\text{exact}}\|_2$. The small circle points out the location of $\lambda_{\text{optimal}} = \text{argmin}\{E_1(\lambda) + E_2(\lambda)\} = 0.0102$ and of $\lambda_{\text{FP}} = \text{argmin}\{\Psi(\lambda),\ \mu = 1\} = 0.0116$. The relative errors in $x_{\lambda_{\text{optimal}}}$ and $x_{\lambda_{\text{FP}}}$ are 5.34% and 5.36% respectively.

- If $\mu = 1$ does not work, μ is adjusted and the iterations restart; see [6, 33] for details.

The GKB-FP algorithm computes approximations to the sought fixed-point by computing a finite sequence of fixed-points of functions $\phi^{(k)}(\lambda) = \sqrt{\mu}\|r_\lambda^{(k)}\|_2/\|x_\lambda^{(k)}\|_2$, where $x_\lambda^{(k)}$ solves the constrained problem

$$x_\lambda^{(k)} = \underset{x \in \mathcal{K}_k(A^TA, A^Tb)}{\text{argmin}} \{\|Ax - b\|_2^2 + \lambda^2\|x\|_2^2\} \tag{11}$$

and $r_\lambda^{(k)} = b - Ax_\lambda^{(k)}$. Based on (5)-(7), it is straightforward to prove that

$$x_\lambda^{(k)} = V_k y_\lambda^{(k)}, \quad \text{with} \quad y_\lambda^{(k)} = \underset{y \in \mathbb{R}^k}{\text{argmin}}\{\|B_k y - \beta_1 e_1\|_2^2 + \lambda^2\|y\|_2^2\}, \tag{12}$$

and the regularized solution norm and the corresponding residual norm satisfy

$$\|x_\lambda^{(k)}\|_2 = \|y_\lambda^{(k)}\|_2, \quad \text{and} \quad \|r_\lambda^{(k)}\|_2 = \|B_k y_\lambda^{(k)} - \beta_1 e_1\|_2. \tag{13}$$

The evaluation of $\phi^{(k)}(\lambda)$ for each λ can be done approximately in $\mathcal{O}(k)$ arithmetic operations, which corresponds to the cost of solving the projected problem (12). Numerical examples on large-scale problems reported in [24] show that the largest convex fixed-point of $\phi(\lambda)$ is quickly captured. For a detailed description of GKB-FP, see [24] again; here we summarize the main steps of GKB-FP, since, as we will see later, they will be also followed by the methods proposed in this work.

GKB-FP Algorithm:

1. Perform $p_0 > 1$ GKB steps applied to A with initial vector $b/\|b\|_2$.

2. For $k \geq p_0$ compute the fixed-point $\lambda_{\text{FP}}^{(k)}$ of $\phi^{(k)}(\lambda)$ until a termination criterion is satisfied.

3. Once convergence is achieved compute $x_\lambda^{(k)}$ using (12).

The value of p_0 can be freely chosen by the user. In our numerical experiments we take: $p_0 = 10$ for small problems, $p_0 = 15$ for relatively large problems, and $p_0 = 20$ for larger problems. As for the initial guess of λ when computing the first fixed point at step $k = p_0$, it is set equal to 10^{-4}. For further details about the initial guess, see [6, 24].

To provide further insight into the convergence properties of GKB-FP, we will concentrate on the question of how well x_λ can be approximated by "projecting" the large-scale problem onto a subspace of small dimension. More specifically, assume that $\{\mathcal{V}_k\}_{k\geq 1}$ is a family of k dimensional subspaces such that $\mathcal{V}_1 \subset \mathcal{V}_2 \subset \cdots \subset \mathcal{V}_k \subset \mathcal{V}_{k+1} \subset \cdots$, and for $k > 1$ consider approximations $x_\lambda^{(k)}$ defined by

$$x_\lambda^{(k)} = \underset{x\in\mathcal{V}_k}{\operatorname{argmin}}\{\|Ax - b\|_2^2 + \lambda^2\|Lx\|_2^2\}. \tag{14}$$

Then our goal is to bound the error $\|x_\lambda - x_\lambda^{(k)}\|_2$ for the case where the regularization parameter is selected by the FP method. To this end we first introduce some notation that will be used later. Let the reduced singular value decomposition of A be given by

$$A = \mathsf{U}\Sigma\mathsf{V}^T, \tag{15}$$

where $\mathsf{U} = [\mathsf{u}_1, \ldots, \mathsf{u}_n] \in \mathbb{R}^{m\times n}$ and $\mathsf{V} = [\mathsf{v}_1, \ldots, \mathsf{v}_n] \in \mathbb{R}^{n\times n}$ have orthonormal columns and $\Sigma = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$ is a diagonal matrix with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n \geq 0$. In addition, let U_k, V_k and $\mathcal{S}_k$ be defined by

$$\mathsf{U}_k = [\mathsf{u}_1, \ldots, \mathsf{u}_k], \qquad \mathsf{V}_k = [\mathsf{v}_1, \ldots, \mathsf{v}_k], \qquad \mathcal{S}_k = \operatorname{span}\{\mathsf{v}_1, \ldots, \mathsf{v}_k\},$$

and notice that if $L = I_n$, then the regularized Tikhonov solution x_λ can be written as

$$x_\lambda = \sum_{i=1}^{n} \frac{\sigma_i(\mathsf{u}_i^T b)}{\sigma_i^2 + \lambda^2}\mathsf{v}_i. \tag{16}$$

We start with the following technical result.

Lemma 2.1. *Let P_k denote the orthogonal projection on $\mathcal{V}_k$. Define $\gamma_k = \|A - AP_k\|_2$ and $\delta_k = \|L - LP_k\|_2$. Then $\forall\lambda > 0$ there holds,*

$$\|Lx_\lambda - Lx_\lambda^{(k)}\|_2 \leq \sqrt{\delta_k^2 + \frac{\gamma_k^2}{\lambda^2}}\|(I_n - P_k)x_\lambda\|_2. \tag{17}$$

Proof : Since $\mathcal{N}(A) \cap \mathcal{N}(L) = \{0\}$, it follows that the function $< u, v >_\# =< Au, Av > +\lambda^2 < Lu, Lv >$, where $< \cdot, \cdot >$ denotes the usual inner product in $\mathbb{R}^n$, defines an inner product in $\mathbb{R}^n$. Note that (3) implies $< Ax_\lambda - b, Ax > +\lambda^2 < Lx_\lambda, Lx >= 0, \forall x \in \mathbb{R}^n$ and that $< Ax_\lambda^{(k)} - b, A\varphi > +\lambda^2 < Lx_\lambda, L\varphi >= 0, \forall\varphi \in \mathcal{V}_k$. Combining these we have

$$< x_\lambda - x_\lambda^{(k)}, \varphi >_\# = 0, \quad \forall\varphi \in \mathcal{V}_k. \tag{18}$$

Therefore, $x_\lambda^{(k)} = \mathcal{P}_k x_\lambda$, where $\mathcal{P}_k$ denotes the orthogonal projector onto the subspace $\mathcal{V}_k$ with respect to the inner product $< \cdot, \cdot >_\#$ and induced norm $\|\cdot\|_\#$. Then

$$\begin{array}{rcl} \|x_\lambda - x_\lambda^{(k)}\|_\#^2 & = & \|x_\lambda - \mathcal{P}_k x_\lambda\|_\#^2 \\ & \leq & \|x_\lambda - P_k x_\lambda\|_\#^2 \\ & = & \|A(I_n - P_k)x_\lambda)\|_2^2 + \lambda^2\|L(I_n - P_k)x_\lambda\|_2^2 \\ & \leq & \|A(I_n - P_k)\|_2^2\|(I_n - P_k)x_\lambda\|_2^2 + \lambda^2\|L(I_n - P_k)\|_2^2\|(I_n - P_k)x_\lambda\|_2^2 \\ & = & (\gamma_k^2 + \lambda^2\delta_k^2)\|(I_n - P_k)x_\lambda\|_2^2. \end{array} .$$

Now it suffices to use the inequality $\lambda^2\|L(x_\lambda - x_\lambda^{(k)})\|_2^2 \leq \|x_\lambda - x_\lambda^{(k)}\|_{\#}^2$ and the proof is complete. □

We are in a position to state the main result of the section.

Theorem 2.1. *Let P_k and P_k be, respectively, the orthogonal projector onto $\mathcal{V}_k$ and $\mathcal{S}_k$, and let Ω_k be the subspace angle between $\mathcal{V}_k$ and $\mathcal{S}_k$. Then for arbitrary $\lambda > 0$ there holds*

$$\|(I_n - P_k)x_\lambda\| \leq \sin(\Omega_k)\|x_\lambda\|_2 + \gamma_k \frac{\|\tilde{b}_k\|_2}{\lambda^2}, \tag{19}$$

where $\tilde{b}_k = b - \mathsf{U}_k\mathsf{U}_k^T b$. As a consequence, whenever $L = I_n$ and the regularization parameter is selected by the fixed-point method, which is denoted by λ_{FP}, there holds

$$\frac{\|x_{\lambda_{\text{FP}}} - x_{\lambda_{\text{FP}}}^{(k)}\|_2}{\|x_{\lambda_{\text{FP}}}\|_2} \leq \sqrt{1 + \frac{\gamma_k^2}{\lambda_{\text{FP}}^2}}\left[\sin(\Omega_k) + \frac{\gamma_k}{\lambda_{\text{FP}}}\frac{\|\tilde{b}_k\|_2}{\|r_{\lambda_{\text{FP}}}\|_2}\right], \tag{20}$$

Proof : The regularized solution x_λ can be written as $x_\lambda = \mathsf{P}_k x_\lambda + (I_n - \mathsf{P}_k)x_\lambda$. Multiplying on both sides of this equality by $(I_n - P_k)$ and taking norms

$$\|(I_n - P_k)x_\lambda\| \leq \|(I_n - P_k)\mathsf{P}_k x_\lambda\|_2 + \|(I - P_k)(I_n - \mathsf{P}_k)x_\lambda\|_2 \tag{21}$$

The first term can be bounded as

$$\|(I_n - P_k)\mathsf{P}_k x_\lambda\|_2 \leq \sin(\Omega_k)\|x_\lambda\|_2, \tag{22}$$

where we used the fact that $\|(I - P_k)\mathsf{P}_k\|_2 = \sin(\Omega_k)$, see [34, Theorem 2.6.1, p. 76]. To bound the second term, observe that from (16)

$$(I_n - \mathsf{P}_k)x_\lambda = \sum_{i=k+1}^{n} \frac{\sigma_i(\mathsf{u}_i^T b)}{\sigma_i^2 + \lambda^2}\mathsf{v}_i = (A^TA + \lambda^2 I_n)^{-1}A^T\tilde{b}_k = A^T(AA^T + \lambda^2 I_m)^{-1}\tilde{b}_k,$$

where we used the SVD of A to obtain $(A^TA + \lambda^2 I_n)^{-1}A^T = A^T(AA^T + \lambda^2 I_m)^{-1}$. Thus,

$$\|(I_n - P_k)(I_n - \mathsf{P}_k)x_\lambda\|_2 = \|(I_n - P_k)A^T(AA^T + \lambda^2 I_m)^{-1}\tilde{b}_k\|_2 \leq \gamma_k \frac{\|\tilde{b}_k\|_2}{\lambda^2}. \tag{23}$$

The inequality (19) follows by replacing (22) and (23) in (21).

On the other hand, if the regularization parameter λ is selected by the fixed-point method, which means, $\lambda_{\text{FP}} = \|r_{\lambda_{\text{FP}}}\|_2/\|x_{\lambda_{\text{FP}}}\|_2$, then (19) becomes

$$\frac{\|(I_n - P_k)x_{\lambda_{\text{FP}}}\|}{\|x_{\lambda_{\text{FP}}}\|_2} \leq \left(\sin(\Omega_k) + \frac{\gamma_k}{\lambda_{\text{FP}}}\frac{\|\tilde{b}_k\|_2}{\|r_{\lambda_{\text{FP}}}\|_2}\right), \tag{24}$$

The second part of the proof concludes using this inequality in (17). □

Theorem 2.1 shows that if the subspace $\mathcal{V}_k$ approximates $\mathcal{S}_k$ well, in which case $\sin(\Omega_k)$ is small, then the solution of the projected problem (14), $x_{\lambda_{\text{FP}}}^{(k)}$, will be a good approximation to $x_{\lambda_{\text{FP}}}$ provided that the ratio $\gamma_k/\lambda_{\text{FP}}$ is sufficiently small. We shall illustrate this by computing the relative error $\|(I_n - P_\ell)x_{\lambda_{\text{FP}}}\|_2/\|x_{\lambda_{\text{FP}}}\|_2$ and its bound (24) for the case where $\mathcal{V}_\ell$ coincides with $\mathcal{S}_\ell$, as well as the bound for the case where we take approximations to $\mathcal{S}_\ell$ instead. To this end, we consider two test problems from [32] and for fixed k, take $\widetilde{\mathcal{S}}_\ell$, $\ell = 1, \ldots, k$, to be the subspace spanned by the first ℓ columns of $\widetilde{\mathsf{V}}_k = V_k \mathbf{v}$, where V_k is from (6) and $\mathbf{v} \in \mathbb{R}^{k\times k}$ is the matrix of right singular vectors of B_k also introduced in (6). It is known that for the first values of ℓ the subspaces $\widetilde{\mathcal{S}}_\ell$ approximate $\mathcal{S}_\ell$ well and that the quality of the approximation deteriorates as ℓ approaches k, see, e.g., [30, Chapter 6]. Computed quantities for $k = 80$ are displayed in Figure 2.

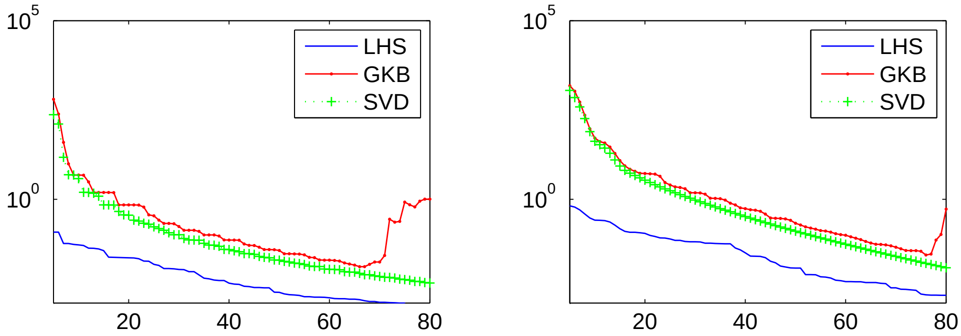

Figure 2: Relative error and estimates (24) for phillips (left) and heat (right) test problems from [32], for $n = 512$ and $b = b^{\text{exact}} + e$ where e is white noise such that $\|e\|_2 = 0.005\|b^{\text{exact}}\|_2$. Here, LHS stands for left-hand side of (24).

We now turn to describe the quantity γ_k as a function of k in connection with the GKB process. To this end we shall assume exact arithmetic and that GKB runs to completion, i.e., we are able to run n GKB steps without interruption, which is reasonable for discrete ill-posed problems since the singular values decay gradually to zero without any gap.

Theorem 2.2. *Assume that the GKB algorithm applied to A runs to completion, and let G_k and $\widehat{G}_k$ be the bidiagonal matrices defined respectively by*

$$G_k = \begin{pmatrix} \alpha_{k+1} & & & \\ \beta_{k+2} & \alpha_{k+2} & & \\ & \ddots & \ddots & \\ & & \beta_n & \alpha_n \\ & & & \beta_{n+1} \end{pmatrix} = \begin{pmatrix} \widehat{G}_k \\ \beta_{n+1} e_{n-k}^T \end{pmatrix} \tag{25}$$

Then for $k = 1, \ldots, n-1$ we have $\gamma_k = \begin{cases} \|G_k\|_2, & \text{if } m > n, \\ \|\widehat{G}_k\|_2, & \text{if } m = n, \end{cases}$ *, and $\gamma_{k+1} \leq \gamma_k$.*

Proof : Assume that $m > n$. If the GKB algorithm runs to completion, from (6) it follows that $A = U_{n+1} B_n V_n$, where $U_{n+1} \in \mathbb{R}^{m\times(n+1)}$ and $V_n \in \mathbb{R}^{n\times n}$ have orthonormal columns, and $B_n \in \mathbb{R}^{(n+1)\times n}$ is lower bidiagonal. But matrix A can be rewritten as

$$A = U \begin{pmatrix} B \\ \mathbf{0} \end{pmatrix} V^T, \tag{26}$$

where $U = [U_{n+1}\, U^\perp] \in \mathbb{R}^{m\times m}$ with $U^\perp$ chosen so that U is orthogonal, $V = V_n$, $B = B_n$, and $\mathbf{0} \in \mathbb{R}^{(m-n-1)\times n}$ is a zero matrix. Then $P_k = V_k V_k^T$, $AP_k = AV_k V_k^T = U_{k+1} B_k V_k^T$ by (5), and $A - AP_k = U(B - U^T U_{k+1} B_k V_k^T V)V^T$. Hence

$$\|A - AP_k\|_2 = \|B - U^T U_{k+1} B_k V_k^T V\|_2 = \|G_k\|_2,$$

which proves the first statement of the theorem.

Now if $m = n$ and the GKB algorithm runs n steps, then we have that $A = U\widehat{B}_nV$, where $U,\ V \in \mathbb{R}^{n\times n}$ have orthonormal columns, and $\widehat{B}_n \in \mathbb{R}^{n\times n}$ defined by

$$\widehat{B}_n = \begin{pmatrix} \alpha_1 & & & \\ \beta_2 & \alpha_2 & & \\ & \ddots & \ddots & \\ & & \beta_n & \alpha_n \end{pmatrix}, \tag{27}$$

and the proof proceeds in the same way as above.

Finally, the second statement is a consequence of the fact that G_{k+1} (resp. $\widehat{G}_{k+1}$) is a submatrix of G_k (resp. $\widehat{G}_k$), which completes the proof. □

Remark 1: The number γ_k can be approximated using the 2-norm of the first column of G_k, that is, if e_1 denotes the unit vector in $\mathbb{R}^{n-k}$, then

$$\gamma_k \gtrapprox ||G_k e_1||_2 = \sqrt{\alpha_{k+1}^2 + \beta_{k+2}^2}. \tag{28}$$

Remark 2: Provided that $\text{rank}(AP_k) = k$, we have

$$\gamma_k \geq \sigma_{k+1}, \tag{29}$$

the equality being attained when $\mathcal{V}_k$ is spanned by the first k right singular vectors of A. In this case AP_k is the matrix of rank k that is closest to A in the 2-norm sense, see, e.g., [34, Thm. 2.5.3]. In practice, because $\mathcal{V}_k$ carries relevant information on the first k right singular vectors, both γ_k and $\sqrt{\alpha_{k+1}^2 + \beta_{k+2}^2}$ approximate σ_{k+1}, with γ_k decreasing as fast as σ_{k+1}, see Figure 3. As a result, whenever the fixed point of $\phi^{(k)}(\lambda)$, $\lambda_{\text{FP}}^{(k)}$, is close to λ_{FP}, the bound (17) becomes small because γ_k decreases quickly near zero. This explains why GKB-FP often converges rapidly, as reported in [24].

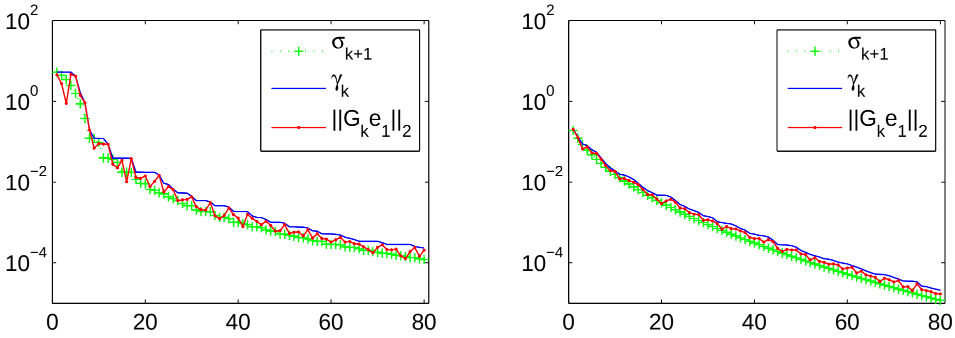

Figure 3: Estimates (28) and (29) for phillips (left) and heat (right) test problems from [32], for $n = 512$ and $b = b^{\text{exact}} + e$ where e is white noise such that $\|e\|_2 = 0.005\|b^{\text{exact}}\|_2$.

3 Extensions of GKB-FP

We shall now consider extensions of GKB-FP to general-form Tikhonov regularization, concentrating, in particular, on problems that involve rank-deficient matrices L with much more rows than columns, as often seen, e.g., in deblurring problems. Two well-distinguished approaches shall be considered. One approach is based on the fact that we can apply GKB-FP to a related transformed problem. The second approach constructs regularized solutions by solving the constrained problem (14).

3.1 Extensions of GKB-FP via standard-form transformation

Theoretically, we can always transform the general-form Tikhonov problem (2) into a problem in standard form

$$\bar{x}_\lambda = \underset{\bar{x}\in\mathbb{R}^p}{\operatorname{argmin}} \left\{ \|\bar{b} - \bar{A}\bar{x}\|_2^2 + \lambda^2 \|\bar{x}\|_2^2 \right\}. \tag{30}$$

If L is invertible we use $y = Lx \Leftrightarrow x = L^{-1}y$, $\bar{b} = b$, and $\bar{A} = AL^{-1}$. When L is not invertible the transformation takes the form

$$x_\lambda = L_A^\dagger \bar{x}_\lambda + x_N, \quad \bar{A} = AL_A^\dagger, \quad \bar{b} = b - Ax_N, \tag{31}$$

where x_N lies in the null space of L and

$$L_A^\dagger = \Big(I_n - \left(A \left(I_n - L^\dagger L \right) \right)^\dagger A \Big) L^\dagger,$$

is the A-weighted generalized inverse of L. If we know a full rank matrix W whose columns span the null space $\mathcal{N}(L)$, then $x_N = W\,(AW)^\dagger\, b$, and the A-weighted generalized inverse of L reduces to $L_A^\dagger = \Big(I_n - W\,(AW)^\dagger A \Big) L^\dagger$ [28]. Thus, for the GKB-FP algorithm to be successfully applied to (30), the matrix $\bar{A}$ must not be explicitly formed and the matrix-vector products with $\bar{A}$ and $\bar{A}^T$ must be performed as efficiently as possible. Is is clear that these matrix-vector products depend on the way the products with $L^\dagger$ and $L^{T^\dagger}$ are performed, and that computation of the product $L^\dagger u$ requires determining the minimum 2-norm least squares solution of the problem

$$t_{\text{LS}} = \operatorname{argmin} \|Lt - u\|_2 \tag{32}$$

as quickly and efficiently as possible; the same observations holds for $L^{T^\dagger} v$. Searching for this efficiency, one can try iterative methods such as CGLS or LSQR and its subspace preconditioned version [16]. Unfortunately, our computational experience with these techniques when $\|Lx\|_2$ is a Sobolev norm has been unsatisfactory due to the slow convergence of the iterates; however, for applications involving sparse regularization matrices e.g., banded regularization matrices with small bandwidth, among others, the following approaches can be considered:

3.1.1 LU-based approach

If L is assumed to be rank deficient with $\operatorname{rank}(L) = q < \min\{p, n\}$, we can determine permutation matrices $P_L \in \mathbb{R}^{p\times p}$ and $Q_L \in \mathbb{R}^{n\times n}$ such that

$$P_L L Q_L = \hat{L}\hat{U} \tag{33}$$

where $\hat{L} \in \mathbb{R}^{p\times q}$ is "unit lower triangular", $\hat{U} \in \mathbb{R}^{q\times n}$ is "upper triangular", and both have rank q. Thus the linear squares problem (32) requires the minimum 2-norm least squares solutions of the full-rank subproblems:

$$y_{\text{LS}} = \operatorname{argmin} \|\hat{L}y - P_L u\|_2, \quad \text{and} \quad t_{\text{LS}} = \operatorname{argmin} \|\hat{U}t - y_{\text{LS}}\|_2. \tag{34}$$

Once t_{LS} is determined, we then get $t = L^\dagger u = Q_L t_{\text{LS}}$. The approach becomes attractive specially when L is banded with small bandwidth; in this case the two above subproblems can be handled efficiently [28, 30], an important condition for GKB-FP to work well.

3.1.2 L_D-based approach

In this case we consider regularization matrices L of the form

$$L = \begin{bmatrix} I_{\tilde{n}} \otimes L_{d_1} \\ L_{d_2} \otimes I_{\tilde{m}} \end{bmatrix}$$

with $L_{d_1} \in \mathbb{R}^{(\tilde{n}-d_1)\times\tilde{n}}$, and $L_{d_2} \in \mathbb{R}^{(\tilde{m}-d_2)\times\tilde{m}}$. The key idea is that if we use the SVDs of the small matrices L_{d_1} and L_{d_2}: $L_{d_1} = U_{d_1}\Sigma_{d_1}V_{d_1}^T$, $L_{d_2} = U_{d_2}\Sigma_{d_2}V_{d_2}^T$, then the smoothing seminorm satisfies the property

$$\|Lx\|_2 = \|L_D x\|_2, \quad \text{with} \quad L_D = D(V_{d_1} \otimes V_{d_2})^T, \tag{35}$$

where D is a nonnegative diagonal such that [14]

$$D^2 = \Sigma_{d_1}^T\Sigma_{d_1} \otimes I_m + I_n \otimes \Sigma_{d_2}^T\Sigma_{d_2}. \tag{36}$$

The null space of L_D, $\mathcal{N}(L_D)$, is spanned by the columns of the matrix $V_{d_1} \otimes V_{d_2}$ associated with the zero entries of D, see [14] again. As a consequence, we can now solve the standard Tikhonov problem (30) associated with

$$x_\lambda = \operatorname*{argmin}_{x\in\mathbb{R}^n} \left\{ \|b - Ax\|_2^2 + \lambda^2 \|L_D x\|_2^2 \right\}, \tag{37}$$

taking advantage of the fact that $V_{d_1} \otimes V_{d_2}$ is orthogonal; in this case the products $L_D^\dagger u$ and $L_D^{\dagger^T} v$ required in (32) can be performed efficiently as

$$L_D^\dagger u = (V_{d_1} \otimes V_{d_2})D^\dagger u, \quad L_D^{\dagger^T} v = D^\dagger (V_{d_1} \otimes V_{d_2})^T v. \tag{38}$$

The main advantage here is that the cost of the subproblem (32) reduces essentially to a matrix-vector product involving the Kronecker product $V_{d_1} \otimes V_{d_2}$, which is important for the efficiency of GKB-FP. Further savings are possible when the matrix A is a Kronecker product of the form $A_1 \otimes A_2$, as seen in image processing and numerical analysis. To see how this can be done, recall that $V_{d_1} \otimes V_{d_2}$ is orthogonal and consider the transformation

$$\check{x} = (V_{d_1} \otimes V_{d_2})^T x. \tag{39}$$

Then (37) reduces to

$$\check{x}_\lambda = \operatorname*{argmin}_{\check{x}\in\mathbb{R}^n} \left\{ \|b - \check{A}\check{x}\|_2^2 + \lambda^2 \|D\check{x}\|_2^2 \right\}, \quad \text{with} \quad \check{A} = (A_1V_{d_1}) \otimes (A_2V_{d_2}), \tag{40}$$

where only the matrices $A_1V_{d_1}$ and $A_2V_{d_2}$ need to be stored. Now it is evident that the Tikhonov problem (40) can be solved much more efficiently than the one in (37). Numerical results that illustrate the efficiency of this approach on deblurring problems are postponed to the next section.

3.2 PROJ-L: GKB-FP free of standard-form transformation

We now consider regularized solutions obtained by minimizing the general-form Tikhonov functional over the subspace $\mathcal{K}_k(A^TA, A^Tb)$, as suggested in [27]. Therefore the approximate solution is now determined as

$$x_\lambda^{(k)} = V_k y_\lambda^{(k)}, \quad y_\lambda^{(k)} = \underset{y\in\mathbb{R}^k}{\text{argmin}}\{\|AV_k y - b\|_2^2 + \lambda^2\|LV_k y\|_2^2\}. \tag{41}$$

Note that if we use the QR factorization of the product LV_k, $LV_k = Q_k R_k$, using (5)-(6) the least squares problem in (41) reduces to

$$y_\lambda^{(k)} = \underset{y\in\mathbb{R}^k}{\text{argmin}}\{\|B_k y - \beta_1 e_1\|_2^2 + \lambda^2\|R_k y\|_2^2\}, \tag{42}$$

which can be computed efficiently in several ways, e.g., by a direct method or by first transforming the stacked matrix $[B_k^T \ \ \lambda R_k^T]^T$ to upper triangular form, as done when implementing GKB-FP [24]. Then it follows that

$$\|Lx_\lambda^{(k)}\| = \|R_k y_\lambda^{(k)}\|_2, \quad \|r_\lambda^{(k)}\|_2 = \|B_k y_\lambda^{(k)} - \beta_1 e_1\|, \tag{43}$$

and the function $\phi^{(k)}(\lambda)$ associated with the projected problem is

$$\phi^{(k)}(\lambda) = \sqrt{\mu}\frac{\|B_k y_\lambda^{(k)} - \beta_1 e_1\|_2}{\|R_k y_\lambda^{(k)}\|_2}. \tag{44}$$

Unlike the approach in [27] where the Tikhonov regularization parameter is determined by the discrepancy principle, our proposal denoted by PROJ-L is to follow the same ideas as GKB-FP. That is, for chosen $p_0 > 1$ and $k \geq p_0$, we determine the largest convex fixed-point $\lambda_{\text{FP}}^{(k)}$ of $\phi^{(k)}(\lambda)$, repeating the process until a stopping criterion is satisfied. Numerical examples have shown that the largest fixed-point of $\phi(\lambda)$ associated with the large-scale problem is captured in a relatively small number of GKB steps.

To make our proposal computationally feasible, the following aspects must be considered

- the initial guess of the fixed-point method on the projected problem at step $k+1$ is taken to be the fixed-point $\lambda_{\text{FP}}^{(k)}$ and
- the QR factorization $LV_k = Q_k R_k$ is calculated only once at step $k = p_0$ and is updated in subsequent steps.

Algorithms for updating the QR factorization can be found in [34, Chapter 12].

4 Alternative to Tikhonov regularization: smoothed preconditioned LSQR

We have seen that GKB-FP requires the projected problem to be solved repeatedly in order to calculate solution and residual norms for distinct values of the regularization parameter λ. This may be expensive in connection with large-scale problems. An alternative is to use iterative regularization in which the number of iterations plays the role of the regularization parameter. Methods in this class include CGLS/LSQR, GMRES, MINRES, and Landweber iterations, among many others, see., e.g., [14, 15, 16, 18, 19, 30, 35]. While the regularizing properties of these methods are reasonably well understood, less clear is the situation when the task is to determine a proper number of iterations without a priori knowledge about the norm of the noise. The purpose of this section is to show how to construct regularized solutions via a version of LSQR which incorporates the smoothing properties of the regularization matrix L into the iterative process and automatically stops the iterations without requiring the norm of the noise as input data.

4.1 Stopping rule for LSQR

Recall that LSQR constructs a sequence of approximations for the solution of (1) by minimizing the residual over the Krylov subspace $\mathcal{K}_k(A^TA, A^Tb)$. That is, the LSQR approximations are defined by

$$x_k = \underset{x\in\mathcal{K}_k(A^TA,A^Tb)}{\text{argmin}} \|Ax-b\|_2, \tag{45}$$

and can be expressed as

$$x_k = V_k y_k, \quad y_k = \underset{y\in\mathbb{R}^k}{\text{argmin}} \|B_k y - \beta_1 e_1\|_2, \tag{46}$$

where B_k and V_k are from (4)-(6) [17]. To describe our stopping rule, we first consider a truncation criterion for the method of truncated SVD (TSVD) solutions. Using the notation introduced in (15) the TSVD solution x_k is defined as

$$\mathsf{x}_k = \mathsf{A}_k^\dagger b = \sum_{j=1}^{k} \frac{\mathsf{u}_j^T b}{\sigma_j}\mathsf{v}_j, \tag{47}$$

where $\mathsf{A}_k = \sum_{j=1}^k \sigma_j \mathsf{u}_j \mathsf{v}_j^T$ and k is the truncation parameter. We shall assume that the noise free data b^{exact} satisfies the discrete Picard condition (DPC) [36], i.e., the coefficients $|u_j^T b^{\text{exact}}|$, on the average, decay to zero faster than the singular values, and that the noise is Gaussian zero mean. These assumptions imply that there exists a integer k^* such that

$$|\mathsf{u}_j^T b| = |\mathsf{u}_j^T b^{\text{exact}} + \mathsf{u}_j^T e| \approx |\mathsf{u}_j^T e| \approx \text{constant}, \ \text{ for } \ j > k^*. \tag{48}$$

We now note that an error estimate for x_k can be expressed as

$$\|\mathsf{x}_k - x^{\text{exact}}\| \leq \|A_k^\dagger b^{\text{exact}} - x^{\text{exact}}\|_2 + \|A_k^\dagger e\|_2 \equiv E_1(k) + E_2(k), \tag{49}$$

with

$$E_1(k) = \left(\sum_{j=k+1}^{n} \frac{|\mathsf{u}_j^T b^{\text{exact}}|^2}{\sigma_j^2}\right)^{1/2}, \quad E_2(k) = \left(\sum_{j=1}^{k} \frac{|\mathsf{u}_j^T e|^2}{\sigma_j^2}\right)^{1/2}. \tag{50}$$

The first term, called the regularization error, decreases with k and can be small when k is large. The second term measures the noise magnification error; it increases with k

and can be large for $\sigma_j \approx 0$. Thus the choice of the truncation parameter requires a balance between these two errors in order to make the overall error small. A closer look at the two types of errors reveals that for $k > k^*$, the error $E_2(k)$ increases dramatically while $E_1(k)$ remains under control due to the DPC, and the overall error should not be minimized for $k > k^*$. Conversely, for $k < k^*$, $E_1(k)$ increases regularly with k while $E_2(k)$ remains under control since the singular values σ_j dominate the coefficients $|\mathsf{u}_j^T e|$; again the overall error is not minimized. Therefore the error estimate should be minimized at $k = k^*$. However, neither $E_1(k)$ nor $E_2(k)$ is available and k^* must be estimated by other means. We propose to do this by minimizing the product

$$\Psi_k = \|\mathsf{r}_k\|_2 \|\mathsf{x}_k\|_2 \ \ k \geq 1, \tag{51}$$

where $\mathsf{r}_k = b - A\mathsf{x}_k = \sum_{j=k+1}^{n} (\mathsf{u}_j^T b)\mathsf{u}_j + b^{\perp}$ with $b^{\perp} = (I_m - \mathsf{U}\mathsf{U}^T)b$; this can be explained as follows. Note from (48) that if $k > k^*$, the residual norm decreases approximately linearly while the solution norm increases dramatically as $0 \approx \sigma_k < |\mathsf{u}_k^T e|$. Thus Ψ_k cannot be minimized for $k > k^*$. On the other hand, if $k < k^*$, the residual norm gets relatively large since the coefficients $|\mathsf{u}_k^T b|$ are large, while the solution norm increases slowly with k. Hence Ψ_k cannot be minimized for $k < k^*$. Thus, the sequence Ψ_k should be minimized at $k = k^*$. We can thus conclude that a good choice of the truncation parameter for TSVD is the minimizer of Ψ_k. Our experience is that this truncation criterion produces a regularization parameter that very often coincides with the optimal parameter, the minimizer of the relative error $\mathrm{RE}_k = \|x^{\mathrm{exact}} - \mathsf{x}_k\|_2 / \|x^{\mathrm{exact}}\|_2$, as seen in Fig. 4.

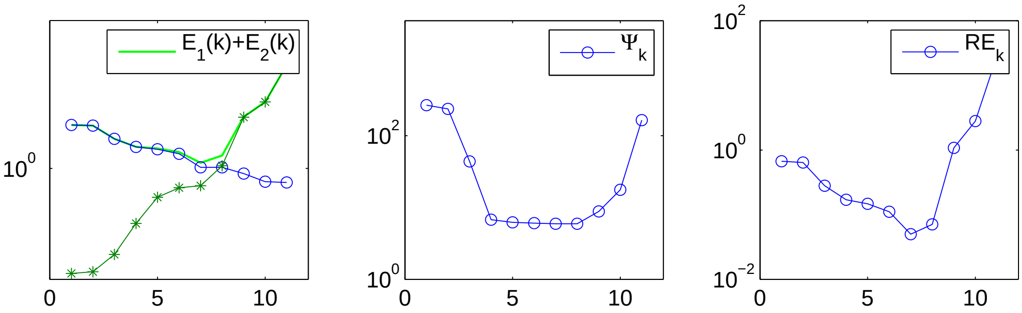

Figure 4: Errors in x_k, Ψ_k and relative errors for shaw test problem using the same data as in with Fig. 1. The optimal relative error is 4.98% and reached at $k^* = 7 = \mathrm{argmin}\, \Psi_k$. The reader should compare this result with that of Tikhonov regularization, see Fig. 1

However, the SVD is infeasible for large-scale problems and the TSVD method may not be of practical utility. As an alternative to TSVD for large-scale problems, we propose to use LSQR coupled with a stopping rule defined similarly as the truncation criterion for TSVD above. Therefore, our stopping rule for LSQR selects as stopping index the *first* integer $\widehat{k}$ satisfying :

$$\widehat{k} = \mathrm{argmin}\, \Psi_k, \quad \Psi_k = \|b - Ax_k\|_2 \|x_k\|_2. \tag{52}$$

The choice of the stopping rule can be supported as follows. First, the residual and solution LSQR norms behave monotonically like the residual and solution TSVD norms, i.e., while $\|b - Ax_k\|_2$ decreases with k, $\|x_k\|_2$ increases, and second, the LSQR iterate x_k lives in the Krylov subspace $\mathcal{K}_k(A^T A, A^T b)$ which very often carries relevant information on the dominant k right singular vectors of A [30], in which case x_k approximates the k-th TSVD solution well, and a similar observation applies to the sequences Ψ_k for TSVD and LSQR, respectively, as we see in Figure 5.

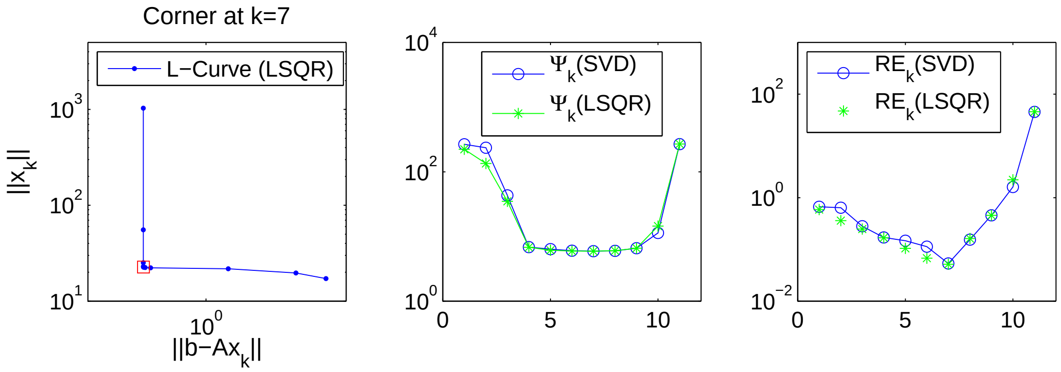

Figure 5: L-curve, sequences Ψ_k associated with TSVD and LSQR for shaw test problem with the same data of the previous figure and error histories of a few iterates.

From the practical point of view, our proposal for determining the stopping index $\widehat{k}$ is to evaluate the finite forward differences

$$\nabla\Psi_k = \Psi_{k+1} - \Psi_k, \quad k \geq 1, \tag{53}$$

and then select the first index k for which $\nabla\Psi_k$ changes sign. More specifically, our proposal is to select the first k such that $\nabla\Psi_{k-1} \leq 0$ and $\nabla\Psi_k \geq 0$. The main feature of this stopping rule is that only $\widehat{k}+1$ GKB steps are required to construct the approximation $x_{\widehat{k}}$ to the noise free solution of (1). All our numerical results to be presented in the sequel are obtained with this stopping rule.

The following result bounds the error in the LSQR iterate x_k as an approximation to the k-th TSVD solution.

Theorem 4.1. *Assume that the GKB algorithm runs to completion. Then the relative distance between the LSQR and the TSVD solution can be bounded as*

$$\frac{\|\mathrm{x}_k - x_k\|_2}{\|x_k\|_2} \leq \frac{1}{\sigma_k}\left(\Phi_k + \gamma_k\right), \tag{54}$$

where $\Phi_k = \|r_k\|_2/\|x_k\|_2$, *and* γ_k *defined in the previous section.*

For the proof of Theorem 4.1 we require a technical result.

Lemma 4.1. *Let* Ω_k *be the angle between the subspaces spanned by the* k *first right singular vectors of* A *and the Krylov subspace* $\mathcal{K}_k$. *Then* $\sin\Omega_k \leq \gamma_k/\sigma_k$.

Proof: As in the proof of Theorem 2.2, we first assume $m > n$. In this case, after n GKB steps matrix A can be decomposed as

$$A = U_{n+1}B_{n+1}V_n^T, \tag{55}$$

in which, for convenience of the proof, we write $U_{n+1} = [U_{k+1}\ U^{\perp}]$, $\quad V_n = [V_k\ V^{\perp}]$, and

$$B_{n+1} = \begin{array}{cc} \begin{array}{cc} k & n-k \end{array} & \\ \left(\begin{array}{c|c} B_k & C_k \\ \hline D_k & F_k \end{array}\right) & \begin{array}{l} k+1 \\ m-k-1 \end{array} \end{array}$$

with B_k defined in (4). The other matrices, C_k, D_k, and F_k are clear from the context. Now by (55) we have $AV^{\perp} = [U_{k+1}\, U^{\perp}] \begin{bmatrix} C_k \\ F_k \end{bmatrix}$. Multiplying this equality with U_k^T and using the fact that $\mathsf{U}_k^T A = \Sigma_k \mathsf{V}_k^T$, where U_k and V_k have k columns and Σ_k contains the k largest singular values of A, we get $\mathsf{V}_k^T V^{\perp} = \Sigma_k^{-1} \mathsf{U}_k^T [U_{k+1}\, U^{\perp}] \begin{bmatrix} C_k \\ F_k \end{bmatrix}$. Taking norms we obtain $\sin \Omega_k \leq \left\| \begin{bmatrix} C_k \\ F_k \end{bmatrix} \right\|_2 / \sigma_k(A)$, where we have used that $\|\mathsf{V}_k^T V^{\perp}\|_2 = \sin \Omega_k$, see, e.g., [34]. Thus, (4.1) follows because the norm in this expression coincides with γ_k.

If $m = n$ then $A = U_n \widehat{B}_n V_n^T$, with $\widehat{B}_n$ defined in (27), and the proof follows analogously. □

Proof of Theorem 4.1: Note that, based on (5)-(6), the LSQR iterate $x_k = V_k y_k$ and the associate residual $r_k = b - Ax_k$ satisfy

$$x_k = V_k B_k^{\dagger} U_{k+1}^T b, \quad B_k^{\dagger} U_{k+1}^T r_k = 0. \tag{56}$$

Then the error in x_k with respect to x_k can be written as

$$\begin{aligned} \mathsf{x}_k - x_k &= \mathsf{A}_k^{\dagger} b - V_k B_k^{\dagger} U_{k+1}^T b \\ &= (\mathsf{A}_k^{\dagger} - V_k B_k^{\dagger} U_{k+1}^T)(r_k + Ax_k) \\ &= \mathsf{A}_k^{\dagger} r_k + (\mathsf{A}_k^{\dagger} A - V_k B_k^{\dagger} U_{k+1}^T A) x_k. \end{aligned}$$

But

$$\mathsf{A}_k^{\dagger} A = \mathsf{A}_k^{\dagger} \mathsf{A}_k, \quad V_k B_k^{\dagger} U_{k+1}^T A x_k = V_k B_k^{\dagger} U_{k+1}^T A V_k V_k^T x_k = V_k B_k^{\dagger} B_k V_k^T x_k = V_k V_k^T x_k = x_k,$$

therefore

$$\mathsf{x}_k - x_k = \mathsf{A}_k^{\dagger} r_k - (I_n - \mathsf{A}_k^{\dagger} \mathsf{A}_k) x_k = \mathsf{A}_k^{\dagger} r_k - (I_n - \mathsf{P}_k) P_k x_k.$$

Hence

$$\|\mathsf{x}_k - x_k\| \leq \|\mathsf{A}_k^{\dagger}\|_2 \|r_k\|_2 + \|(I_n - \mathsf{P}_k) P_k\|_2 \|x_k\|_2 \leq \|\mathsf{A}_k^{\dagger}\|_2 \|r_k\|_2 + \sin \Omega_k \|x_k\|_2,$$

where we have used the fact that $\|(I_n - \mathsf{P}_k) P_k\|_2 = \sin \Omega_k$ (see, Golub, [34, Theorem 2.6.1, p. 76]). The assertion of the theorem follows on using Lemma (4.1). □

Theorem 4.1 states that if the information contained in $\mathcal{K}_k(A^T A, A^T b)$ is rich enough so that $\gamma_k \ll \sigma_k$ and if the sum $\gamma_k + \Phi_k$ is small compared to σ_k, then the LSQR iterate x_k will be close to the k-th TSVD solution. Theorem 4.1 can also be used to bound the error in x_k with respect to x^{exact} by using the triangular inequality and bounds on $\|\mathsf{x}_k - x^{\text{exact}}\|$ which require smoothness conditions on x^{exact}. This is quite involved and is not considered in the present paper.

We now turn to the stopping rule (52). Note that it is nothing more than a discrete counterpart of the parameter choice rule (8) for $\mu = 1$ which looks for a corner of the continuous Tikhonov L-curve; hence, it should come as no surprise to see the minimizer of Ψ_k closely related to a point on the discrete L-curve

$$(\log \|b - Ax_k\|_2, \log \|x_k\|_2), \quad k = 1, \ldots, q, \tag{57}$$

located near the corner of the L-shaped region, see Fig. 5. Thus, the decision to stop LSQR in connection with discrete ill-posed problems can, in principle, be also managed by locating the corner of discrete L-curves for which several algorithms exist. Corner finding methods include a method based on spline curve fitting by Hansen and O'Leary [3], the triangle method by Castellanos et al. [37], a method by Rodriguez and Theis [38], and an adaptive algorithm referred to as the "pruning algorithm" by Hansen et al. [20]. The pruning algorithm advocates that to overcome difficulties of its predecessors, the corner must be determined by evaluating the overall behavior of the L-curve. The effectiveness of this approach and its capability to determine the "best" corner of discrete L-curves is illustrated numerically in [20]. However, finding the corner using a limited sequence of points is not an easy task, and the existing algorithms are not without difficulties, see, e.g., Hansen [30, p. 190] for discussions on shortcomings, Hansen et al. [20] for a multiple corner case, and Salehi Ravesh et al. [22] for an application of the L-curve criterion to quantification of pulmonary microcirculation.

To illustrate the performance of the pruning algorithm in connection with LSQR (with full reorthogonalization), we use gravity, heat, foxgood and shaw test problems from [32], moler (with $\alpha = 0.5$), lotkin, prolate and hilbert test problems from Matlab's "matrix gallery". In all cases we consider coefficient matrices of size 1024×1024 and distinct right-hand sides defined by $b = Ax^{\text{exact}} + e$, where e is generated by the Matlab function randn with the state value set to 0 and with three distinct noise levels (NL) $\text{NL} = \|e\|_2 / \|Ax^{\text{exact}}\|_2 = 10^{-4},\ 10^{-3}, 10^{-2}$. To ensure that the overall behavior of the L-curve is contained in the data, we take $q = 120$ points. The corner of the L-curve, denoted by k_{LC}, is determined by using the Matlab code corner from [32], and the relative error in $x_{k_{\text{LC}}}$ is denoted by E_{LC}. For comparison, the stopping index $\widehat{k}$ determined by minimizing Ψ_k, the relative error in $x_{\widehat{k}}$ denoted by E_Ψ, optimal parameters and optimal errors, denoted by k_{opt} and E_{opt}, respectively, are also computed.

Average relative errors of 20 realizations as well as the minimum/maximum k determined by the pruning algorithm, the stopping rule (52) and the optimal one along the realizations, are all displayed in Table 1.

Note that, except for the fact that the pruning algorithm failed solving moler and lotkin test problems for $\text{NL} = 10^{-4}$ and $\text{NL} = 10^{-2}$, respectively, the quality of the approximate solutions produced by this algorithm and the stopping rule (52) remains comparable in the other cases and other test problems. In particular, for the noise level $\text{NL} = 10^{-4}$, we note that while the pruning algorithm produced approximate solutions of poor quality, the solutions produced by the stopping rule (52) remained within tolerable bounds. The pruning algorithm failed constructing reasonable approximate solutions because the corner of the L-curve was not correctly identified several times, see Fig. 6 (left). Fig. 6 (center) shows the discrete L-curve of the first realization with the corner determined by the pruning algorithm (corresponding to $k_{\text{LC}} = 107$) marked by □ (in red) and with the "true" corner located at $k = 19$ marked by ○ (in red). A similar observation applies for lotkin test problem.

To learn more about the properties of the pruning algorithm, we investigated the

	$\mathrm{NL} = 10^{-4}$					
	k_{LC}	$\widehat{k}$	k_{opt}	E_{LC}	E_{Ψ}	$\mathrm{E}_{\mathrm{opt}}$
gravity	13(19)	11(14)	11(13)	0.0522	0.0109	0.0043
heat	46(52)	28(42)	28(31)	0.0902	0.0175	0.0125
foxgood	4(5)	5(5)	4(4)	0.0096	0.0119	0.0028
shaw	9(9)	9(9)	9(9)	0.0325	0.0325	0.0325
moler	18(108)	19(21)	9(11)	8.7458	0.1283	0.0107
lotkin	7(7)	7(7)	7(9)	0.4384	0.4384	0.4317
prolate	20(22)	10(12)	9(13)	0.0223	0.0002	0.0002
hilbert	9(10)	9(9)	10(12)	0.4358	0.4382	0.4258

	$\mathrm{NL} = 10^{-3}$					
	k_{LC}	$\widehat{k}$	k_{opt}	E_{LC}	E_{Ψ}	$\mathrm{E}_{\mathrm{opt}}$
gravity	11(15)	10(11)	9(11)	0.0368	0.0224	0.0118
heat	27(31)	28(29)	20(22)	0.0774	0.0691	0.0222
foxgood	3(4)	3(4)	3(3)	0.0201	0.0201	0.0074
shaw	7(8)	7(8)	7(9)	0.0514	0.0515	0.0439
moler	9(10)	9(10)	7(8)	0.0496	0.0654	0.0220
lotkin	3(5)	5(5)	5(7)	0.4478	0.4475	0.4445
prolate	17(17)	12(16)	6(11)	0.0260	0.0145	0.0008
hilbert	7(8)	7(8)	7(9)	0.4396	0.4396	0.4391

	$\mathrm{NL} = 10^{-2}$					
	k_{LC}	$\widehat{k}$	k_{opt}	E_{LC}	E_{Ψ}	$\mathrm{E}_{\mathrm{opt}}$
gravity	7(11)	7(8)	6(8)	0.0454	0.0356	0.0266
heat	15(17)	16(16)	14(16)	0.0674	0.0674	0.0629
foxgood	2(2)	2(2)	2(3)	0.0311	0.0311	0.0217
shaw	5(5)	5(6)	6(8)	0.1094	0.0660	0.0534
moler	4(4)	4(4)	5(6)	0.1885	0.1885	0.0788
lotkin	3(18)	3(3)	3(5)	1.9×10^{5}	0.4522	0.4505
prolate	10(13)	7(12)	1(6)	0.0173	0.0150	0.0071
hilbert	6(6)	6(6)	6(7)	0.4400	0.4400	0.4400

Table 1: Regularization parameters and relative errors of regularized solutions determined by the pruning algorithm, the stopping rule (52) and the optimal one. The exact solution for moler, lotkin, prolate and hilbert test problems were taken to be the solution of shaw test problem from [32].

behavior of k_{LC} as a function of the number of data points q being used. The results for moler test problem with $\mathrm{NL} = 10^{-4}$ are displayed in Fig. 6 (right). We note that for several values of q the corner index k_{LC} stagnates at $k = 21$, which coincides with the maximum minimizer of Ψ_k, and that for larger values of q this corner index falls far away from $k = 21$. Corner indexes for heat and gravity test problems did not behave this way and they are not reported here. The results suggest that the corner index k_{LC} depends on the number of points q, and that unfortunate choices of q might lead to wrong corners. We shall return to this point later in connection with a deblurring problem.

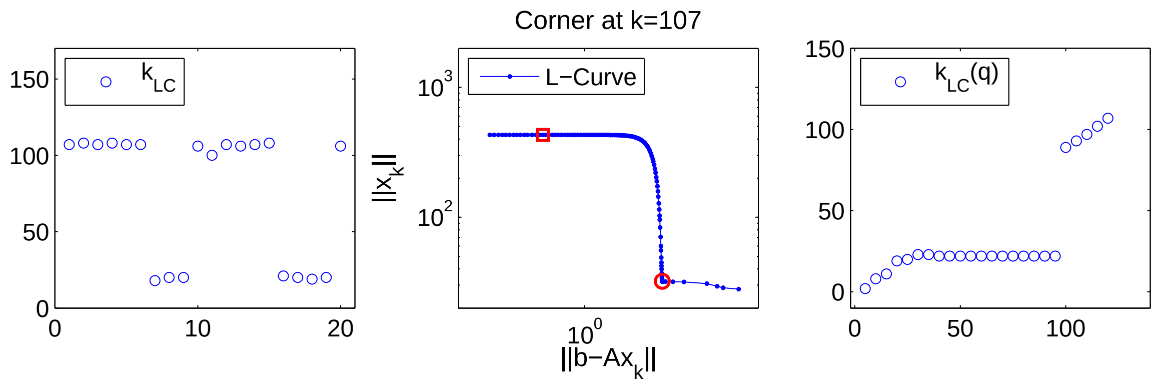

Figure 6: Corner index determined by the pruning algorithm of 20 realizations for moler test problem in connection with LSQR (left). L-curve of first realization (center); in this case, the corner determined by pruning algorithm is marked by □ while the "true" corner is marked by ○. Corner index determined by pruning algorithm using q points with q ranging from 30 until 120 (right).

4.2 P-LSQR

As commented earlier, our intention is to construct approximate solutions using a version of LSQR that incorporates smoothing properties of the regularization matrix L into the computed solution. We shall do this by applying LSQR to the least squares problem $\min \|\bar{A}x - \bar{b}\|$ with $\bar{A}$ and $\bar{b}$ from (30), as suggested in [13], using (52) as stopping rule. Our iterative regularization algorithm, which we denote by P-LSQR, can thus be summarized as follows

- apply LSQR to $\|\bar{A}x - \bar{b}\|_2$ and compute the iterate $\bar{x}_k$ using the stopping rule defined in (52).
- once the stopping index is determined, take as approximate solution

$$x_k = L_A^\dagger \bar{x}_k + x_N,$$

  with x_N and $L_A^\dagger$ as in (31).

Obviously for P-LSQR to be computationally feasible, the dimension of $\mathcal{N}(L)$ must be small and the products with $L^\dagger$ and $L^{T^\dagger}$ must be performed as efficiently as possible, i.e., all possibilities must be explored in order to reduce the computational cost, either through fast matrix-vector products or through the use of preconditioners. However, the latter is not considered in this paper.

5 Numerical examples

We give some examples to illustrate our methods. Two examples involve deblurring problems and a third one involves a Super-resolution problem; as before, the data vector is of the form $b = Ax^{\text{exact}} + e$, where e is generated by the Matlab code randn with the state value set to 0, and NL $= \|e\|_2/\|Ax^{\text{exact}}\|_2$ is referred to as the *the noise level.* All computations were carried out in Matlab. To simplify the notation, the extension of GKB-FP based on the LU factorization of L is denoted by FP-LU and the L_D-based approach is denoted by FP-L_D. Average values of regularization parameters, time and relative error in $x_\lambda^{(k)}$, are denoted by $\bar{\lambda}$, $\bar{t}$ and $\bar{E}$, respectively, while the minimum and maximum number of steps required by the algorithms are denoted by k_m and k_M, respectively.

5.1 Deblurring problems

The goal in this case is to recover an image stored in a vector $x^{\text{exact}} \in \mathbb{R}^{MN}$ from a blurred and noisy image $b = b^{\text{exact}} + e$ so that $Ax^{\text{exact}} = b^{\text{exact}}$, where A plays the role of blurring operator (often referred to as the PSF matrix), and b^{exact} represents the blurred image. For simplicity, we consider $N \times N$ images and use $N^2 \times N^2$ PSF matrices defined by $A = (2\pi\sigma^2)^{-1} T \otimes T$. In this case, σ controls the width of the Gaussian point spread function and T is an $N \times N$ symmetric banded Toeplitz matrix with half-bandwidth band [15]; in what follows we use $\sigma = 2$ and $\mathsf{band} = 16$. The regularization matrix is chosen as

$$L = \begin{bmatrix} I_N \otimes L_1 \\ L_1 \otimes I_N \end{bmatrix}, \quad L_1 = \begin{bmatrix} 1 & \text{-1} & & & \\ & 1 & \text{-1} & & \\ & & \ddots & \ddots & \\ & & & 1 & \text{-1} \end{bmatrix} \in \mathbb{R}^{(N-1)\times N}. \tag{58}$$

5.1.1 Rice test problem

Rice 64: The purpose here is two-fold: to illustrate that the methods proposed in this work are less expensive than the joint bidiagonalization (JBD) algorithm by Kilmer et al. [15] and, to learn more about the capabilities of the pruning algorithm to identify the corner of discrete L-curves. We start with the observation that at the k-th step, the JBD-based approach determines approximate solutions given by

$$x_\lambda^{(k)} = \underset{x \in \mathcal{Z}_k}{\operatorname{argmin}} \|b - Ax\|_2^2 + \lambda^2 \|Lx\|_2^2, \tag{59}$$

where $\mathcal{Z}_k$ is a Krylov subspace generated by the JBD algorithm applied to Q_A and Q_L, with $Q_A \in \mathbb{R}^{m \times n}$, $Q_L \in \mathbb{R}^{p \times n}$ and $Q = [Q_A^T \, Q_L^T]^T$, where Q is from the QR factorization of $\widehat{A} = [A^T \, L^T]^T$. Each step of the JBD approach requires two matrix vector products of the form $\widehat{v} = QQ^T v$, but the QR factorization is never computed; in practice one takes

$$\widehat{v} = \widehat{A} u_{\text{LS}}, \quad \text{with} \quad u_{\text{LS}} = \underset{u \in \mathbb{R}^n}{\operatorname{argmin}} \|v - \widehat{A}u\|_2. \tag{60}$$

This not only explains why the JBD-based approach is expensive but also shows that its efficiency depends on the way u_{LS} is computed. Two distinct ways were considered in this paper: one uses LSQR as reported in [39], and the other one uses LSQR with subspace preconditioning, as done in [12, 16]. We will report results obtained through the former, which turned out to be the most efficient. To achieve our first goal we will use the JBD algorithm with the FP method as parameter choice rule, which we call JBD-FP. JBD-FP proceeds like PROJ-L in that for chosen $p_0 > 1$ and for $k \geq p_0$, the largest convex fixed-point of $\phi^{(k)}(\lambda)$ is computed and the process is repeated until a stopping criterion is satisfied. Two distinct implementations of JBD-FP were considered: one implementation denoted by JBD-FP$_L$, which deals with problem (2) using the matrix L, and other denoted by JBD-FP$_D$, which deals with the equivalent problem (40). In this example we consider a 64×64 subimage of the image *rice* which was used in [15] to illustrate a JBD-based algorithm. Thus $N = 64$, $A \in \mathbb{R}^{4096\times 4096}$ and $L \in \mathbb{R}^{8064\times 4096}$; similarly as in [15], we use data with 1% of white noise. Both GKB and JBD were implemented with complete reorthogonalization; computation of fixed-points started with $p_0 = 10$, and the iterations stopped using a tolerance parameter $\epsilon = 10^{-6}$.

For future comparison, regularization parameters determined by L-curve, a fixed-point method, the stopping rule (52), optimal regularization parameters, and relative errors, are all reported in Table 2. Regularization parameters for L-curve and Fixed-Point methods were computed using the GSVD of the pair (A, R), where $R \in \mathbb{R}^{4095\times 4096}$ is from the QR factorization of L. The optimal Tikhonov regularization parameter, defined as the minimizer of $\|x_\lambda - x^{\text{exact}}\|_2/\|x^{\text{exact}}\|_2$, is computed via exhaustive search. P-LSQR uses $\bar{A}$ and $\bar{b}$ from the explicit transformation approach implemented in std_form in [32]. As it is apparent, see Figure 7, in this case the L-Curve has a well defined corner and $\phi(\lambda)$ (with $\mu = 1$) has a unique fixed-point that minimizes $\Psi(\lambda)$. The true image, the blurred and noisy image, and the images determined by L-Curve and fixed-point methods are all displayed in Fig. 8.

	L-Curve	FP	optimal	P-LSQR	optimal
λ	0.0783	0.0956	0.0409	$k = 48$	$k = 83$
E	8.03%	8.19%	7.75%	8.31%	7.82%

Table 2: Regularization parameters and relative errors.

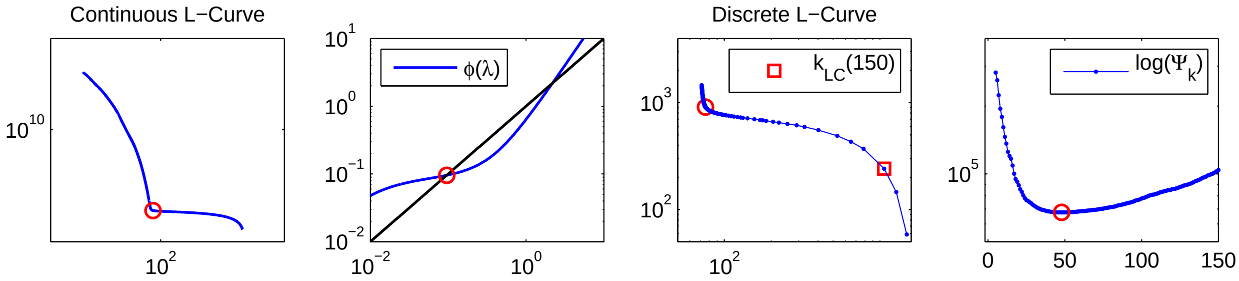

Figure 7: L-Curve, $\phi(\lambda)$, discrete L-curve and $\log(\Psi_k)$ for rice test problem with $N = 64$. The corner of the L-curve, the fixed-point of $\phi(\lambda)$, the “true” corner of the discrete L-curve, and the minimizer of Ψ_k, are all marked by ○. The corner determined by the pruning algorithm of the first of 20 realizations using $q = 150$ points is marked by □.

We now turn to the results obtained through JBD-FP and the methods proposed in this paper. Average time of 20 realizations are Table 3. It becomes apparent that the JBD-FP approaches are in fact more expensive than the methods proposed in this work, and that the fastest one is P-LSQR followed by PROJ-L and FP-L$_D$. Note that FP-LU can be a good option. As far accuracy is concerned, all methods produced solutions with relative error of approximately 8.2%; this is in accordance with the results obtained using the GSVD of the pair (A, R) shown in Table 2.

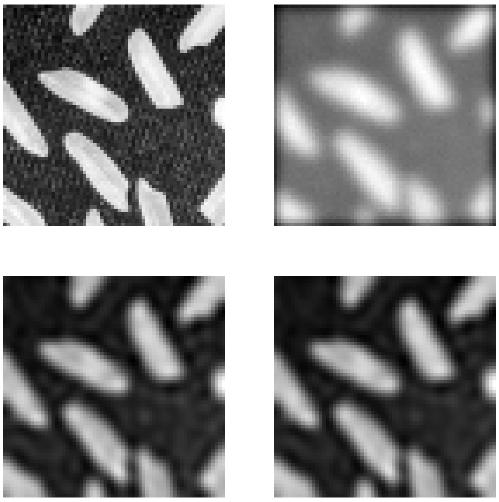

Figure 8: True image (top left), blurred and noisy image (top right), reconstructed image by LC (bottom left) and reconstructed image by FP (bottom right).

	JBD-FP$_L$	JBD-FP$_D$	FP-LU	FP-L$_D$	PROJ-L	P-LSQR
$\bar{t}$	5.2206	4.2688	2.0084	0.8863	0.2576	0.2809

Table 3: Average time (in seconds) of 20 realizations.

Concerning the capability of the pruning algorithm to find the corner of discrete L-curves, we arrive at the same conclusion as before: the corner index k_{LC} can vary with the number of points q of the discrete L-curve. Table 4 shows the corner index k and the relative error in x_k of the first realization for several values of q. The starting value of q was chosen relatively close to the minimizer of Ψ_k, $k = 48$, to evaluate how the corner index behaves in these cases. A false corner corresponding to $q = 150$ is displayed in Fig. 7.

q	60	80	100	120	140	160	180	200
$k_{\text{LC}}(q)$	3	3	43	43	3	3	46	43
Error	20.65%	20.65%	8.46%	8.46%	20.65%	20.65%	8.38%	8.46%

Table 4: Corner index $k_{\text{LC}}(q)$ selected by the pruning algorithm as a function of the number of points q of the discrete L-curve $(\log \|\bar{r}_k\|_2, \log \|\bar{A}x - \bar{b}\|_2)$, where $\bar{r}_k$ and $\bar{x}_k$ are LSQR iterates of $\min \|\bar{A}x - \bar{b}\|_2$ for rice test problem.

The conclusion we can drawn from the numerical experiments so far is that LSQR coupled with the proposed stopping rule is cheaper than the pruning algorithm, that the corner determined by the pruning algorithm depends on the number of points of the discrete L-curve, and that for the tested problems our approach performs similarly as the pruning algorithm when the latter works well. These conclusions explain why the pruning algorithm will not be used in the following examples.

Rice 256: We now consider the 256×256 entire *rice* image. The PSF matrix $A \in \mathbb{R}^{65536 \times 65536}$, it has singular values decaying gradually to zero without any particular gap (not shown here) and condition number $\kappa(A) \approx 3.40 \times 10^{16}$. The regularization matrix

$L \in \mathbb{R}^{130560 \times 65336}$. Only FP-L$_D$, P-LSQR and PROJ-L are used. Average results of 20 realizations each with NL $= 0.01$ with and without complete reorthogonalization (labeled as reorth $= 1$ and reorth $= 0$, respectively) are displayed in Table 5. Again, all methods produced solutions with approximately the same quality regardless of whether reorthogonalization is used or not. However, in terms of speed, PROJ-L is superior. In this example $\mu = 1$ did not work in all runs and adjustments were needed. For details on such an adjustment the reader is referred to [6].

	FP-L$_D$	P-LSQR	PROJ-L	FP-L$_D$	P-LSQR	PROJ-L
$\bar{\lambda}$	0.0706	-	0.0874	0.0746	-	0.0874
$\bar{E}$	0.0820	0.0851	0.0831	0.0812	0.0845	0.0831
$\bar{t}$	7.9603	4.3365	1.0652	54.6983	15.9995	1.8808
k_m (k_M)	433 (512)	242 (268)	28 (30)	280 (290)	140 (141)	28 (30)
$\bar{\mu}$	0.6519	-	1	0.6635	-	1
		reorth $= 0$			reorth $= 1$	

Table 5: Results for entire *rice* test problem for NL $= 0.01$, $p_0 = 15$ and $\epsilon = 10^{-6}$.

5.1.2 Pirate test problem

In this example we consider a large image of size 512 × 512 called *pirate*, see Fig. 9. Thus $N = 512$, the PSF matrix $A \in \mathbb{R}^{262144 \times 262144}$ and the regularization matrix $L \in \mathbb{R}^{523264 \times 262144}$. As in the previous example, we report average results of 20 realizations using FP-L$_D$, P-LSQR and PROJ-L with and without complete reorthogonalization. Numerical results for the noise level NL $= 0.01$ are shown in Table 6. In this case, the relative error in the computed solutions is approximately 14.7% and once more the fastest algorithm is PROJ-L. Visual results of this experiment can be seen in figure 9. For this test problem, the choice $\mu = 1$ works satisfactorily in all runs.

	FP-L$_D$	P-LSQR	PROJ-L	FP-L$_D$	P-LSQR	PROJ-L
$\bar{\lambda}$	0.1563	-	0.1491	0.1577	-	0.1492
$\bar{E}$	0.1479	0.1483	0.1463	0.1472	0.1477	0.1462
$\bar{t}$	64.5863	36.4999	12.5072	426.0727	156.7809	17.6866
$k_m(k_M)$	516 (572)	309 (335)	39 (39)	282 (282)	163 (163)	39 (39)
		reorth $= 0$			reorth $= 1$	

Table 6: Results for Pirate test problem with NL $= 0.01$, $p_0 = 20$ and $\epsilon = 10^{-6}$.

5.2 Super-resolution

High-resolution (HR) images are important in a number of areas such as medical imaging and video surveillance. However, due to hardware limitations and cost of image acquisition systems, Low-resolution (LR) images are often available. We consider the problem of estimating an HR image form observed multiple LR images. Let the original HR image of size $M = M_1 \times M_2$ in vector form be denoted by $x \in \mathbb{R}^M$, and let the k-th LR image of size $N = N_1 \times N_2$ in vector form be denoted by $b_k \in \mathbb{R}^N$, $k = 1, 2, \ldots, q$,

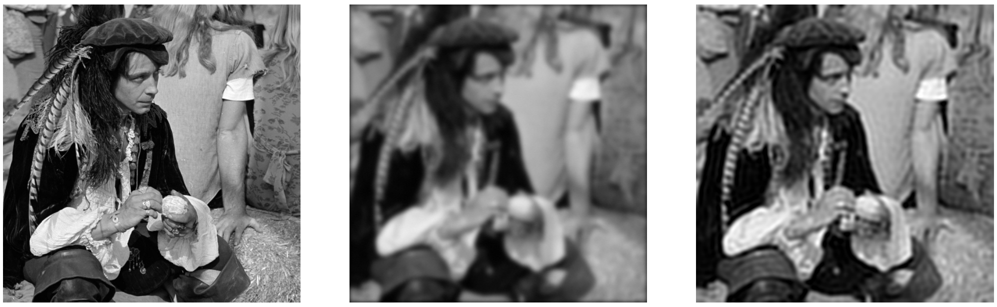

Figure 9: True image, LR noisy image, and restored image by PROJ-L.

with $M_1 = N_1 \times D_1$, $M_2 = N_2 \times D_2$, where D_1 and D_2 represent down-sampling factors for the horizontal and vertical directions, respectively. Assuming that the acquisition process of the LR sequence involves blurring, motion, subsampling and additive noise, an observation model that relates x to b_k is written as [40]

$$b_k = A_k x + \epsilon_k \tag{61}$$

where A_k is $N \times M$, and ϵ_k stands for noise. The goal is to estimate the HR image x from all LR images b_k. In this case Tikhonov regularization takes the form

$$x_\lambda = \arg \min_{x \in \mathbb{R}^M} \{\|b - Ax\|_2^2 + \lambda^2 \|Lx\|_2^2\} \tag{62}$$

where $b = [b_1^T \dots b_q^T]^T$, $A = [A_1^T \dots A_q^T]^T$, and L is a discrete 2D differential operator.

In this example we estimate the 96×96 image *tree* from a sequence of five noisy LR images with $D_1 = D_2 = 2$. Therefore $A \in \mathbb{R}^{11520 \times 9216}$ and $L \in \mathbb{R}^{18240 \times 9216}$. As both A and the regularization matrices are not too large, the JBD-FP approaches are used again and all implemented with complete reorthogonalization. Average results of 20 realizations are shown in Table 7. We note again that PROJ-L is superior. For this example the choice $\mu = 1$ worked satisfactorily in all cases. The original HR image and two noisy LR images are depicted in the first row of Fig. 10. One of the restored images determined by PROJ-L is depicted in the second row of the same figure. Also, to illustrate the performance of the algorithms using distinct regularization matrices, we consider the cases $L = I$ and

	FP-LU	FP-L$_D$	P-LSQR	PROJ-L	JBD-FP	JBD-FP
$\bar{\lambda}$	0.0309	0.0309	-	0.0300	0.0309	0.0309
$\bar{E}$	0.0496	0.0496	0.0502	0.0535	0.0497	0.0497
$\bar{t}$	16.4869	9.1734	3.2403	0.8753	21.6653	25.1221
k_m (k_M)	281 (284)	281 (284)	157 (160)	47 (57)	112 (114)	112 (114)

Table 7: Results for super-resolution problem for NL = 0.01, $\epsilon = 10^{-6}$, and $p_0 = 15$.

$$L_{2,2\text{D}} = \begin{bmatrix} I_N \otimes L_2 \\ L_2 \otimes I_N \end{bmatrix}, \quad L_2 = \begin{bmatrix} \text{-1} & 2 & \text{-1} & & \\ & \ddots & \ddots & \ddots & \\ & & \text{-1} & 2 & \text{-1} \end{bmatrix} \in \mathbb{R}^{(N-2) \times N}.$$

In this case we obtained solutions with relative error of approximately 28.80% and 4.95%, respectively. That is, while the quality of the solutions for the case $L = I$ deteriorate

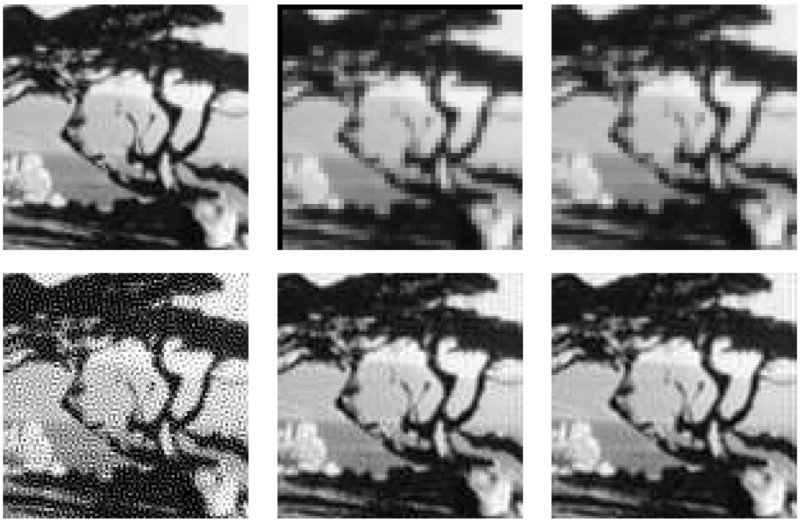

Figure 10: First row: HR image (left) and two LR noisy images. Second row: Restored image with $L = I$ determined by GKB-FP (left), restored image with L from (58) determined by PROJ-L (center), and restored image with $L_{2,2\text{D}}$ determined by PROJ-L (right).

significantly, the quality of the solutions using $L_{2,2\text{D}}$ remain practically the same as that obtained using L from (58). Visual results are depicted in Fig. 10.

We end the numerical results section with the observation that we also performed numerical experiments involving the three above deblurring test problems using data with distinct noise levels. The results showed that the extensions of GKB-FP perform similarly as the original version of the algorithm, and are not show here.

6 Conclusions

We reviewed the GKB-FP algorithm and showed how to extend it to large-scale general form Tikhonov regularization. As a result, three distinct approaches that do not require estimates of the noise level were proposed and numerically illustrated on large-scale deblurring problems. Numerical results for representative test problems using data with realistic noise levels (e.g., 0.1% and 0.01%) showed that in term of accuracy and efficiency, the extended versions of GKB-FP perform satisfactorily in as much the same way as the original version of the algorithm, and are therefore competitive and attractive for large-scale general-form Tikhonov regularization. In addition, to overcome possible difficulties in GKB-FP when selecting the Tikhonov regularization parameter at each iteration, we proposed a stopping rule for LSQR, and showed numerically that the smoothed preconditioned LSQR algorithm (P-LSQR) coupled with the new rule can be a good alternative to large-scale general-form Tikhonov regularization.

Acknowledgments

The authors are grateful to the reviewers for their valuable comments which have significantly improved the presentation of this work.

References

[1] Tikhonov AN. Solution of incorrectly formulated problems and the regularization method. *Soviet Mathematics Doklady* 1963; **4**:1035-1038.

[2] Morozov VA. *Regularization Methods for Solving Incorrectly Posed Problems.* Springer: New York, 1984.

[3] Hansen PC, O'Leary DP. The use of the L-curve in the regularization of discrete ill-posed problems. *SIAM Journal on Scientific Computing* 1993; **14**:1487-1503.

[4] Golub GH, Heath M, Wahba G. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics* 1979; **21**:215-222.

[5] Regińska T. A regularization parameter in discrete ill-posed problems. *SIAM Journal on Scientific Computing* 1996; **3**:740-749.

[6] Bazán FSV. Fixed-point iterations in determining the Tikhonov regularization parameter. *Inverse Problems* 2008; **24**: DOI:10.1088/0266-5611/24/3/035001.

[7] Engl HW, Hanke M, Neubauer A. *Regularization of Inverse Problems.* Kluwer: Dordrecht, 1996.

[8] Bakushinski AB. Remarks on choosing a regularization parameter using quasi-optimality and ratio criterion. *USSR Computational Mathematics and Mathematical Physics* 1984; **24**:181-182.

[9] Kindermann S. Convergence analysis of minimization-based noise level-free parameter choice rules for linear ill-posed problems. *Electronic Transactions on Numerical Analysis* 2011; **38**:233-257.

[10] Bauer F, Lukas MA. Comparing parameter choice methods for regularization of ill-posed problems. *Mathematics and Computers in Simulation* 2011; **81**:1795-1841.

[11] Reichel L, Rodriguez G. Old and new parameter choice rules for discrete ill-posed problems. *Numerical Algorithms* 2012; DOI 10.1007/s11075-012-9612-8.

[12] Bunse-Gerstner A, Guerra-Ones V, Madrid de la Vega H. An improved preconditioned LSQR for discrete ill-posed problem. *Mathematics and Computers in Simulation* 2006; **73**:65-75.

[13] Hanke M, Hansen PC. Regularization methods for large-scale problems. *Surveys on Mathematics for Industry* 1993; **3**:253-315.

[14] Hansen PC, T. K. Jensen TK. Smoothing-Norm Preconditioning for Regularizing Minimum-Residual Methods. *SIAM Journal on Matrix Analysis and Applications* 2006; **29**:1-14.

[15] Kilmer ME, Hansen PC, Español MI. A projection-based approach to general-form Tikhonov regularization. *SIAM Journal on Scientific Computing* 2007; **29**:315-330.

[16] Jacobsen M, Hansen PC, Saunders MA. Subspace preconditioned LSQR for discrete ill-posed problems. *BIT* 2003; **43**:975-989.

[17] Paige CC, Saunders MA. LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software* 1982; **8**:43-71.

[18] Saad Y, Schultz MH. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**:856-869.

[19] Neuman A, Reichel L, Sadok H. Implementations of range restricted iterative methods for linear discrete ill-posed problems. *Linear Algebra and its Applications* 2012; **436**:3974-3990.

[20] Hansen PC, Jensen TK, Rodriguez G. An adaptive pruning algorithm for the discrete L-curve criterion. *Journal of Computational and Applied Mathematics* 2007; **198**:483-492.

[21] Morigi S, Reichel L, Sgallari F, Zama F. Iterative methods for ill-posed problems and semiconvergent sequences. *Journal of Computational and Applied Mathematics* 2006; **193**:157-167.

[22] Salehi Ravesh M, Brix G, Laun FB, Kuder TA, Puderbach M, Ley-Zaporozhan J, Ley S, Fieselmann A, Herrmann MF, Schranz W, Semmler W, Risse F. Quantification of pulmonary microcirculation by dynamic contrast-enhanced magnetic resonance imaging: Comparison of four regularization methods. *Magnetic Resonance in Medicine* 2012; DOI: 10.1002/mrm.24220.

[23] Chung J, Nagy JG, O'Leary DP. A Weighted-GCV Method for Lanczos-Hybrid Regularization. *Electronic Transaction on Numerical Analysis* 2008; **28**:149-167.

[24] Bazán FSV, Borges LS. GKB-FP:an algorithm for large-scale discrete ill-posed problems. *BIT* 2010; **50**:481-507.

[25] Lampe J, Reichel L, Voss H. Large-Scale Tikhonov Regularization via Reduction by Orthogonal Projection. *Linear Algebra and its Applications* 2012; **436**:2845-2865.

[26] Reichel L, Sgallari F, Ye Q. Tikhonov regularization based on generalized Krylov subspace methods. *Applied Numerical Mathematics* 2012; **62**:1215-1228.

[27] Hochstenback ME, Reichel L. An Iterative Method For Tikhonov Regularization With a General Linear Regularization Operator. *Journal of Integral Equations and Applications* 2010; **22**:463-480.

[28] Eldén L. A weighted pseudoinverse, generalized singular values, and constrained least square problems. *BIT* 1982; **22**:487-502.

[29] Golub GH, Kahan W. Calculating the singular values and pseudo-inverse of a matrix. *Journal of the Society for Industrial and Applied Mathematics Series B Numerical Analysis* 1965; **2**:205-224.

[30] Hansen PC. *Rank-deficient and discrete ill-posed problems.* SIAM: Philadelphia, 1998.

[31] Kirsch A. *An Introduction to the Mathematical Theory of Inverse Problems.* Springer: New York, 1996.

[32] Hansen PC. Regularization Tools: A MATLAB package for analysis and solution of discrete ill-posed problems. *Numerical Algorithms* 1994; **6**:1-35.

[33] Bazán FSV, Francisco JB. An improved Fixed-point algorithm for determining the Tikhonov regularization parameter. *Inverse Problems* 2009; **25**: DOI:10.1088/0266-5611/25/4/045007.

[34] Golub GH, Van Loan CF. *Matrix Computations.* The Johns Hopkins University Press: Baltimore, 1996.

[35] Paige CC, Saunders MA. Solution of sparse indefinite systems of linear equations. *SIAM Journal on Numerical Analysis* 1975; **12**:617-629.

[36] Hansen PC. The Discrete Picard Condition For Discrete Ill-Posed Problems. *BIT* 1990; **30**:658-672.

[37] Castellanos JJ, Gómez S, Guerra V. The triangle method for finding the corner of the L-curve. *Applied Numerical Mathematics* 2002; **43**:359-373.

[38] Rodriguez G, Theis D. An algorithm for estimating the optimal regularization parameter by the L-curve. *Rendiconti di Matematica* 2005; **25**:69-84.

[39] Saunders MA. Computing projections with LSQR. *BIT* 1997; **37**:96-104.

[40] Park SC, Park MK, Kang MG. Super-resolution image reconstruction: a technical overview. *IEEE Signal Processing Magazine* 2003; **20**:21-36.