SIAM J. Sci. Comput.
Vol. 37, No. 5, pp. S30–S50

A GENERALIZED KRYLOV SUBSPACE METHOD FOR ℓ_p-ℓ_q MINIMIZATION[*]

A. LANZA[†], S. MORIGI[†], L. REICHEL[‡], AND F. SGALLARI[†]

Abstract. This paper presents a new efficient approach for the solution of the ℓ_p-ℓ_q minimization problem based on the application of successive orthogonal projections onto generalized Krylov subspaces of increasing dimension. The subspaces are generated according to the iteratively reweighted least-squares strategy for the approximation of ℓ_p/ℓ_q-norms by weighted ℓ_2-norms. Computed image restoration examples illustrate that it suffices to carry out only a few iterations to achieve high-quality restorations. The combination of a low iteration count and a modest storage requirement makes the proposed method attractive.

Key words. ℓ_p-ℓ_q minimization, generalized Krylov subspaces, iteratively reweighted least-squares, half-quadratic, image restoration

AMS subject classifications. 68U10, 65K10, 65F22, 65F10

DOI. 10.1137/140967982

1. Introduction.

This paper is concerned with the computation of approximate solutions of ℓ_p-ℓ_q minimization problems of the form

$$\min_{x\in\mathbb{R}^n} \mathcal{J}(x), \qquad \mathcal{J}(x) = \frac{1}{p}\|Ax-b\|_p^p + \frac{\mu}{q}\|\Phi(x)\|_q^q\,, \tag{1.1}$$

where $0 < p, q \le 2$, $\mu > 0$, $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $x \in \mathbb{R}^n$, and $\Phi : \mathbb{R}^n \to \mathbb{R}^s$ is a linear or nonlinear operator of interest in image restoration problems with particular properties. Examples of such operators will be given below. The general minimization problem (1.1) encompasses a wide variety of problems that have been extensively studied in many different research areas, including numerical linear algebra [1, 33], image restoration [7, 28], pattern recognition [9, 21], and compressive sensing [6, 7, 10, 22]. Different choices of the parameters p and q, the matrix A, and the operator Φ yield a variety of popular models that have been successfully used in many application fields. For instance, the model (1.1) with $p = 2$, $m < n$, $0 < q \le 1$, and Φ the identity matrix can be used to compute sparse solutions of undetermined linear systems; when $p = 2$, $0 < q \le 1$, and A is a sampling operator, the model (1.1) can be applied for compressive sensing.

In this paper we focus on the efficient solution of the ℓ_p-ℓ_q problem (1.1) with application to image restoration. However, the proposed method easily can be used for the solution of all the other abovementioned applications that use the model (1.1). In the context of image restoration, the vector x represents the unknown uncorrupted image that is to be estimated, while the available noise- and possibly blur-contaminated image is represented by the vector b. Typically, both x and b are column-major representations of the corresponding two-dimensional images and have

[*] Received by the editors May 5, 2014; accepted for publication (in revised form) November 11, 2014; published electronically October 29, 2015.
http://www.siam.org/journals/sisc/37-5/96798.html

[†] Department of Mathematics, University of Bologna, Bologna, Italy (alessandro.lanza2@unibo.it, serena.morigi@unibo.it, fiorella.sgallari@unibo.it).

[‡] Department of Mathematical Sciences, Kent State University, Kent, OH 44242 (reichel@math.kent.edu). The research of this author was partially supported by NSF grant DMS-1115385.

the same size (that is, $m = n$); however, this is not a requirement of the proposed method.

The first term in the definition of $\mathcal{J}(x)$ is commonly referred to as the *fidelity term* and the second term as the *regularization term.* The operator Φ is chosen to yield a computed solution with some known desired features. The scalar $\mu > 0$ is a regularization parameter. Its purpose is to balance the influence of the fidelity and regularization terms on the computed solution in a suitable manner.

In image restoration applications, the matrix A is the identity operator or a blurring operator. In the former case, the purpose of solving (1.1) is to denoise an available noise-contaminated image b. Thus, the computed solution x is a denoised version of b. When A is a blurring operator, the solution x of (1.1) is a deblurred and denoised version of an available blur- and noise-contaminated image b. Blurring operators generally are severely ill-conditioned and may be singular. Due to the ill-conditioning of A and the presence of noise in b, minimization of only the fidelity term in (1.1) typically yields a meaningless computed solution of very large norm. The purpose of the regularization term is to be able to determine a useful solution of (1.1) of moderate norm.

When $p = q = 2$ and Φ is a linear operator that is represented by a regularization matrix $L \in \mathbb{R}^{s\times n}$, the minimization problem (1.1) turns into a classical Tikhonov-regularized linear least-squares problem of the form

$$\min_{x\in\mathbb{R}^n} \left\{ \|Ax - b\|_2^2 + \mu \|Lx\|_2^2 \right\} . \tag{1.2}$$

Assuming that

$$\operatorname{rank} \begin{bmatrix} A \\ L \end{bmatrix} = n \, ,$$

the problem (1.2) has the unique solution

$$x^* = \left(A^T A + \mu\, L^T L\right)^{-1} A^T b$$

for any $\mu > 0$. Here and below, the superscript ^{T} denotes transposition. When L is the identity matrix, denoted by I, the Tikhonov regularization problem (1.2) is said to be in *standard form* and the solution can be efficiently computed by partial Golub–Kahan bidiagonalization of A; see, e.g., [3, 4, 5]. Golub–Kahan bidiagonalization also can be applied when $L \neq I$, provided that the regularization problem can be transformed to standard form without too much effort by applying the *A-weighted pseudoinverse* of L; see [11] for details. For the situation when the A-weighted pseudoinverse of L is complicated and unattractive to use, other methods have been proposed in the literature. A scheme that projects L into a Krylov subspace

$$\mathcal{K}_l(A^T A, A^T b) = \operatorname{span}\{A^T b, (A^T A)A^T b, \ldots, (A^T A)^{l-1} A^T b\}$$

for some $l \geq 1$ and determines an approximate solution of (1.2) in this subspace is described in [14]. Methods based on reducing both A and L by generalized Golub–Kahan-type or Arnoldi-type methods are discussed in [15, 25], and a scheme that computes a partial generalized singular value decomposition of the matrix pair $\{A, L\}$ is advocated in [16]. An efficient iterative algorithm, based on the nonlinear Arnoldi framework developed by Voss [30], for the solution of large-scale Tikhonov regularization problems (1.2) with automatic selection of the regularization parameter μ is proposed in [18]. The method relies on successive orthogonal projections onto generalized Krylov subspaces of increasing and low dimension. We refer to this method as the generalized Krylov subspace (GKS) method and will present an extension that can be applied to the solution of (1.1).

The popular total variation (TV) regularization method is obtained by setting $q = 1$ in (1.1) and letting $\Phi = \Phi_{\mathrm{TV}} : \mathbb{R}^n \to \mathbb{R}^n$ be the (nonlinear) map from the image x to the ℓ_2-norm of its gradient field. Thus,

$$\|x\|_{\mathrm{TV}} = \|\Phi_{\mathrm{TV}}(x)\|_1 = \sum_{i=1}^{n} \|\big(\nabla x\big)_i\|_2 = \sum_{i=1}^{n} \sqrt{\big(D_1 x\big)_i^2 + \big(D_2 x\big)_i^2}\ . \tag{1.3}$$

Each element of x represents a pixel and $\big(\nabla x\big)_i \in \mathbb{R}^2$ denotes the discrete gradient of x at pixel i. The linear operators $D_1, D_2 \in \mathbb{R}^{n \times n}$ are finite difference approximations of first-order horizontal and vertical partial derivatives, respectively. TV-norm regularization has been applied extensively and successively to image restoration [19, 20, 28, 29] because of its ability to preserve edges. Substituting (1.3) into (1.1) yields the ℓ_p-TV restoration model

$$\min_{x \in \mathbb{R}^n} \left\{ \frac{1}{p} \|Ax - b\|_p^p + \mu \|x\|_{\mathrm{TV}} \right\}, \tag{1.4}$$

where the choice of $0 < p \leq 2$ depends on the type of noise in the available contaminated image b. For additive Gaussian noise one generally chooses $p = 2$ and for impulse noise the standard choice is $p = 1$. These p-values give the popular ℓ_2-TV and ℓ_1-TV models, respectively. The ℓ_1-TV model also has been applied successfully in the contexts of image cartoon-texture decomposition [31] and multiscale image decomposition [32]. Notice that, except in some special cases, such as for the abovementioned TV model, the method proposed in this paper requires that the nonlinear operator Φ be linearized.

Many algorithms have been proposed for the efficient solution of problems of the form (1.4) for certain p values. An ad hoc proposal for image restoration, which uses the coherence map of the image, has been introduced in [2]. However, to the best of our knowledge, the only algorithm that can be applied with no or little modification to the solution of the general ℓ_p-ℓ_q minimization problem (1.1) with $0 < p, q \leq 2$ is the iteratively reweighted norm (IRN) approach introduced in [27]. This scheme is shown to be equivalent to the half-quadratic (HQ) method [7] and to the gradient linearization iterative procedure [23]. As will be illustrated in section 2, the IRN method consists of iteratively solving a sequence of penalized weighted least-squares problems that differ from each other only by the choice of the diagonal weighting matrices.

This paper presents a novel efficient numerical method for the solution of the general ℓ_p-ℓ_q minimization problem (1.1). Our method generalizes the GKS approach described in [18] for the solution of Tikhonov-regularized least-squares problems (1.2). The proposed approach significantly improves the computational efficiency of the IRN method. We will refer to our approach as the GKSpq method.

The organization of this paper is as follows. Sections 2 and 3 introduce the main ingredients of the GKSpq method by reviewing the IRN and GKS approaches. Our new GKSpq method is described in section 4, and its convergence is discussed in section 5. Numerical examples are presented in section 6, and concluding remarks can be found in section 7.

2. The IRN method.

In this section we express the ℓ_p-ℓ_q minimization problem (1.1) as an equivalent sequence of weighted ℓ_2-ℓ_2 minimization problems following the

HQ technique. The latter has been applied successfully to image restoration; see, e.g., [7]. Our formulation of the problem (1.1) gives a solution method that is equivalent to the one proposed in [27].

Our discussion of the problem (1.1) relies on the following expression for the pth power of $t \in \mathbb{R}\backslash\{0\}$, $0 < p < 2$, shown in [7, Lemma 1],

$$|t|^p = \min_{w>0} \left\{ w\, t^2 \,+\, \frac{1}{\beta_p w^{\alpha_p}} \right\}, \tag{2.1}$$

where α_p and β_p are positive scalars defined by

$$\alpha_p = \frac{p}{2-p} \qquad \text{and} \qquad \beta_p = \frac{2^{\frac{2}{2-p}}}{(2-p)\cdot p^{\frac{p}{2-p}}} \,. \tag{2.2}$$

The minimum is achieved at

$$w^* = \frac{p}{2}\,|t|^{p-2}\,. \tag{2.3}$$

Note that the function inside the curly bracket in (2.1) is quadratic in the parameter t, but not in the optimization variable w; hence the name *half-quadratic*. In order to be able to apply (2.1)–(2.3) for $t = 0$, we need to use the smoothed version $|t|_\epsilon := \sqrt{t^2 + \epsilon}$ for any $t \in \mathbb{R}$ and some $\epsilon > 0$. For simplicity of notation, we will suppress the subscript ϵ.

The functional $\mathcal{J}$ in (1.1) can be written componentwise as

$$\mathcal{J}(x) = \frac{1}{p}\sum_{i=1}^{m} |(Ax)_i - b_i|^p + \frac{\mu}{q}\sum_{j=1}^{s} |\big(\Phi(x)\big)_j|^q\,, \tag{2.4}$$

where $(Ax)_i$, b_i, and $(\Phi(x))_j$ denote the ith and jth entries of the vectors $Ax, b \in \mathbb{R}^m$ and $\Phi(x) \in \mathbb{R}^s$, respectively. The HQ approach presented in [7] exploits the relations (2.1)–(2.3) to minimize (2.4). Using (2.1) and (2.4), the problem (1.1) can be expressed as the constrained minimization problem

$$\min_{x, v>0, z>0} \mathcal{L}(x, v, z)\,, \tag{2.5}$$

where

$$\mathcal{L}(x,v,z) := \frac{1}{p}\,\sum_{i=1}^{m}\left(v_i|(Ax)_i - b_i|^2 + \frac{1}{\beta_p v_i^{\alpha_p}}\right) + \frac{\mu}{q}\,\sum_{j=1}^{s}\left(z_j|\big(\Phi(x)\big)_j|^2 + \frac{1}{\beta_q z_j^{\alpha_q}}\right),$$

and the constraints $v > 0$ and $z > 0$ are considered componentwise. We have

$$\begin{aligned}
\min_{x\in\mathbb{R}^n} \mathcal{J}(x) \;=\; \min_{x\in\mathbb{R}^n} \quad & \left\{ \;\frac{1}{p}\,\sum_{i=1}^{m} \min_{v_i>0}\left\{ v_i|(Ax)_i - b_i|^2 + \frac{1}{\beta_p v_i^{\alpha_p}}\right\}\right. \\
& \left. + \,\frac{\mu}{q}\,\sum_{j=1}^{s} \min_{z_j>0}\left\{ z_j|\big(\Phi(x)\big)_j|^2 + \frac{1}{\beta_q z_j^{\alpha_q}}\right\} \;\right\} \\
= \min_{x,v>0,z>0} \quad & \left\{ \;\frac{1}{p}\,\sum_{i=1}^{m}\left(v_i|(Ax)_i - b_i|^2 + \frac{1}{\beta_p v_i^{\alpha_p}}\right)\right. \\
& \left. + \,\frac{\mu}{q}\,\sum_{j=1}^{s}\left(z_j|\big(\Phi(x)\big)_j|^2 + \frac{1}{\beta_q z_j^{\alpha_q}}\right) \;\right\},
\end{aligned} \tag{2.6}$$

which is (2.5).

To solve the constrained nonlinear problem (2.6), we apply an alternating minimization iterative procedure, namely, for $k = 0, 1, \ldots,$ we solve successively

$$v^{(k+1)} = \arg\min_{v>0} \ \mathcal{L}(x^{(k)}, v, z^{(k)}), \tag{2.7}$$

$$z^{(k+1)} = \arg\min_{z>0} \ \mathcal{L}(x^{(k)}, v^{(k+1)}, z), \tag{2.8}$$

$$x^{(k+1)} = \arg\min_{x} \ \mathcal{L}(x, v^{(k+1)}, z^{(k+1)}). \tag{2.9}$$

The minimization problems (2.7) and (2.8) are separable into one-dimensional subproblems for each component. Using (2.1) and (2.3), we see that these subproblems have the explicit solutions

$$v_i^{(k+1)} = \frac{p}{2}\left|\left(Ax^{(k)}\right)_i - b_i\right|^{p-2}, \quad i = 1, 2, \ldots, m, \tag{2.10}$$

$$z_j^{(k+1)} = \frac{q}{2}\left|\left(\Phi(x^{(k)})\right)_j\right|^{q-2} \quad, \quad j = 1, 2, \ldots, s. \tag{2.11}$$

We solve the minimization problem (2.9) by first defining the diagonal weighting matrices $W_F^{(k+1)} \in \mathbb{R}^{m\times m}$ and $W_R^{(k+1)} \in \mathbb{R}^{s\times s}$ according to

$$W_F^{(k+1)}{=}\mathrm{diag}(w_F^{(k+1)}), \quad w_F^{(k+1)} = \frac{2}{p}\, v^{(k+1)} = \left|Ax^{(k)} - b\right|^{p-2}, \tag{2.12}$$

$$W_R^{(k+1)}{=}\mathrm{diag}(w_R^{(k+1)}), \quad w_R^{(k+1)} = \frac{2}{q}\, z^{(k+1)} = \left|\Phi(x^{(k)})\right|^{q-2} \tag{2.13}$$

the right-hand side norm is componentwise. Then we rewrite (2.6) in a more compact form, in which terms that do not depend on x are omitted,

$$x^{(k+1)} = \min_{x\in\mathbb{R}^n} \left\{ \left\| \left(W_F^{(k+1)}\right)^{1/2} (Ax - b) \right\|_2^2 + \mu \left\| \left(W_R^{(k+1)}\right)^{1/2} \Phi(x) \right\|_2^2 \right\}. \tag{2.14}$$

This minimization problem is a weighted regularized least-squares problem. The problem is linear when the operator Φ is linear, and nonlinear otherwise. Note that in all the cases of interest described in section 1, including nonlinear TV-norm regularization, the operator Φ is represented in (2.14) by a fixed matrix; see, e.g., [27]. We therefore from now on assume Φ to be linear, and represent Φ by the matrix $L \in \mathbb{R}^{s\times n}$.

Introduce the $n \times n$ matrix

$$T(W_F, W_R) := A^T W_F A + \mu L^T W_R L\ . \tag{2.15}$$

The normal equations associated with the minimization problem (2.14) read as

$$T\big(W_F^{(k+1)}, W_R^{(k+1)}\big)\, x = A^T W_F^{(k+1)} b \tag{2.16}$$

At each (outer) iteration $k = 0, 1, \ldots,$ the standard IRN method [27] solves the linear system of equations (2.16) with a symmetric positive definite matrix for (an approximation of) $x^{(k+1)}$ with the aid of the conjugate gradient (CG) algorithm, using as initial iterate the approximate solution $x^{(k)}$ obtained from the previous iteration, and terminating the CG (inner) iterations as soon as a specified stopping criterion is satisfied; see section 6 for more details. Algorithm 1 illustrates the main steps of the standard IRN method. Details of the algorithm are commented on below.

ALGORITHM 1. THE IRN METHOD [27] FOR THE SOLUTION OF THE ℓ_p-ℓ_q PROBLEM (1.1).

```
Inputs: $A \in \mathbb{R}^{m\times n}$, $L \in \mathbb{R}^{s\times n}$, $b \in \mathbb{R}^m$, $0 < p, q \leq 2$, $\mu > 0$
Output: Approximate solution $x^*$ of (1.1)
1. Initialize:   $x^{(0)} = b$  or  $x^{(0)} = \left(A^TA + \mu L^TL\right)^{-1} A^Tb$
2. for  $k = 0, 1, \ldots$ until convergence do
3.     Compute weighting matrices $W_F^{(k+1)}$ and $W_R^{(k+1)}$ by (2.12) and (2.13)
4.     Compute approximate solution $x^{(k+1)}$ by solving the linear system (2.16)
5. end for
6. $x^* = x^{(k+1)}$
```

The kth (outer) iteration of Algorithm 1 consists of two main steps. The first step (line 3) computes the diagonal weighting matrices by (2.12) and (2.13). The cost of this computation is dominated by the evaluation of one matrix-vector product (MVP) with A in (2.12) and one with L in (2.13). The second step (line 4) solves the linear system of equations (2.16) by the CG algorithm. First, the right-hand side vector of (2.16) is formed with a dominating cost of one MVP with the matrix A^T. Then the cost of each CG (inner) iteration is dominated by one MVP evaluation with the matrix $T(W_F, W_R)$ defined in (2.15). This matrix is not explicitly formed. Instead, each MVP with $T(W_F, W_R)$ is computed by evaluating one MVP with each one of the matrices L, L^T, A, and A^T; cf. (2.15). The cost of the two MVP evaluations with the diagonal weighting matrices W_F and W_R is negligible.

Summarizing, the computational cost of K iterations with Algorithm 1 is dominated by $3K + 4\,\mathrm{CGits}_K$ MVP evaluations, with CGits_K denoting the total number of CG iterations carried out.

3. The GKS method.

This section reviews the GKS method proposed in [18]. The method is designed for the solution of large-scale Tikhonov regularization problems (1.2) with a general regularization matrix $L \in \mathbb{R}^{s\times n}$. We notice that (1.2) represents a very specific instance of the ℓ_p-ℓ_q problem in (1.1) obtained for $p = q = 2$. The method described in [18] determines the regularization parameter $\mu > 0$ by the discrepancy principle [12]; however, many other approaches to compute this parameter can be used (see below). The GKS method computes a sequence of orthogonal projections of generalized Krylov subspaces onto subspaces of low dimension.

The iterative method starts with a user-chosen subspace $\mathcal{V}_0 \subset \mathbb{R}^n$ of low dimension $l \ll n$. In [18], one lets $\mathcal{V}_0 = \mathcal{K}_l(A^TA, A^Tb)$ for $l \leq 5$. The columns of the matrix $V_0 \in \mathbb{R}^{n\times l}$ form an orthonormal basis for the space $\mathcal{V}_0$.

Assume for the moment that an estimate $\delta > 0$ of the ℓ_2-norm of the noise in b is available. The discrepancy principle then prescribes that the regularization parameter μ in (1.2) be chosen so that the solution x^* satisfies $\|Ax^* - b\|_2 = \eta\delta$ for some user-chosen constant $\eta \geq 1$ that is independent of δ. It is applied in [18] to determine $\mu = \mu_k$ for each iteration $k = 0, 1, \ldots$ and to compute the associated solution $x^{(k)}$ of the Tikhonov minimization problem (1.2) restricted to the subspace $\mathcal{V}_k$. This subspace is spanned by the orthonormal columns of the matrix V_k. Thus, $x^{(k)}$ is determined as follows:

$$y^{(k)} = \arg\min_{y\in\mathbb{R}^{l+k}} \left\{ \|AV_ky - b\|_2^2 + \mu_k\|LV_ky\|_2^2 \right\}, \quad x^{(k)} = V_ky^{(k)}\,. \tag{3.1}$$

Define the $n \times n$ matrix

$$T(\mu) := A^T A + \mu L^T L \,.$$

The solution $y^{(k)}$ of the reduced minimization problem (3.1) can be determined by solving the associated normal equations,

$$V_k^T T(\mu_k) V_k \, y = V_k^T A^T b \,, \tag{3.2}$$

and the corresponding approximate solution $x^{(k)}$ of the original unreduced problem (1.2) is given by $x^{(k)} = V_k y^{(k)}$. Note that QR factorizations of the matrices AV_k and LV_k with small $(l+k) \times (l+k)$ upper triangular matrices are available and can be used in the solution process; see [18] for details.

Following the approach of the nonlinear Arnoldi method proposed by Voss [30], the subspace $\mathcal{V}_k$ is expanded to $\mathcal{V}_{k+1}$ by adding a new basis vector v_{new} to $\mathcal{V}_k$. This basis vector is determined by normalizing the residual $r^{(k)}$ of the unreduced problem (1.2). Thus,

$$v_{\text{new}} = \frac{r^{(k)}}{\|r^{(k)}\|_2} \,, \qquad r^{(k)} = T(\mu_k) x^{(k)} - A^T b \,. \tag{3.3}$$

Note that the residual vector $r^{(k)}$ is parallel to the gradient of the functional minimized in the original unreduced problem (1.2) evaluated at $x^{(k)}$. In the absence of round-off errors, $r^{(k)}$ is orthogonal to the search space $\mathcal{V}_k$. To enforce orthogonality in the presence of round-off errors, the residual $r^{(k)}$ can be reorthogonalized against $\mathcal{V}_k$ before normalization.

We remark that the space $\mathcal{V}_k$ is a Krylov subspace only in very special situations. Since the μ_k are updated during the iterations, so is the matrix $T(\mu_k)$. Therefore, the space $\mathcal{V}_k$, in general, is not a Krylov subspace when $L \neq I$. In particular, the new basis vector v_{new} cannot be computed with a short recurrence relation. For this reason, we refer to the search space $\mathcal{V}_k$ as a generalized Krylov subspace. Algorithm 2 describes the main steps of the GKS method.

Algorithm 2. The GKS method [18] for the solution of the ℓ_2-ℓ_2 problem (1.2).

```
Inputs: $A \in \mathbb{R}^{m\times n}$, $L \in \mathbb{R}^{s\times n}$, $b \in \mathbb{R}^m$, $\delta > 0$
Output: Approximate solution $x^*$ of (1.2)
1. Initialize:   $V_0 \in \mathbb{R}^{n\times l}$ such that $V_0^T V_0 = I_l$
2. for  $k = 0, 1, \dots$ until convergence do
3.     Compute regularization parameter $\mu_k$ by the discrepancy principle (see [18])
4.     Compute solution $y^{(k)}$ of reduced problem in (3.1) by solving (3.2)
5.     Compute residual $r^{(k)}$ of unreduced problem (1.2) by (3.3)
6.     Compute new basis vector $v_{\text{new}}$ by normalizing residual $r^{(k)}$
7.     Enlarge subspace: $V_{k+1} = [V_k, v_{\text{new}}]$
8. end for
9. Determine approximate solution $x^* = V_k y^{(k)}$
```

We note that Algorithm 2 also can be applied when no estimate of the norm of the noise in b is available. In this situation, the regularization parameter μ_k has to be determined by a so-called heuristic parameter choice rule. Recent discussions on heuristic parameter choice rules and computed examples that illustrate their performance can be found, e.g., in [17, 24] and references therein.

4. The GKSpq method.

It is our aim to improve the efficiency of the standard IRN approach [27] described by Algorithm 1 for the solution of the ℓ_p-ℓ_q minimization problem (1.1). By using the HQ formulation, the ℓ_p-ℓ_q minimization problem can be expressed in the form (2.14) and then be solved by applying ideas that form the basis for the GKS approach. Thus, our solution subspaces are chosen to be nested generalized Krylov subspaces of increasing dimension. They are constructed using the solutions of (2.14) for increasing values of k. This defines the GKSpq method. The initial subspace $\mathcal{V}_0$ is generated in the same manner as for the GKS method, and at each iteration $k = 0, 1, \ldots,$ we first determine the new weighting matrices $W_F^{(k+1)}$ and $W_R^{(k+1)}$ in the same way as in the IRN algorithm. Having at our disposal the matrix V_k, whose orthonormal columns form a basis for the generalized Krylov subspace $\mathcal{V}_k$, the solution $y^{(k+1)}$ of the current weighted least-squares problem (2.14) restricted to the subspace $\mathcal{V}_k$ satisfies

$$\min_{y\in\mathbb{R}^{k+l}} \left\{ \left\| \left(W_F^{(k+1)}\right)^{1/2} (AV_k y - b) \right\|_2^2 + \mu \left\| \left(W_R^{(k+1)}\right)^{1/2} LV_k y \right\|_2^2 \right\}. \tag{4.1}$$

The normal equations associated with this reduced least-squares problem are given by

$$V_k^T T\big(W_F^{(k+1)}, W_R^{(k+1)}\big) V_k\, y = V_k^T A^T W_F^{(k+1)} b \tag{4.2}$$

with the matrix $T(W_F^{(k+1)}, W_R^{(k+1)})$ defined by (2.15). We note that the matrix $V_k^T T(W_F^{(k+1)}, W_R^{(k+1)}) V_k$ is of small order $k + l \ll n$.

The solution $y^{(k+1)}$ of (4.1) is computed by solving the least-squares problem

$$\min_{y\in\mathbb{R}^{k+l}} \left\| \begin{bmatrix} W_F^{1/2} AV \\ \mu^{1/2} W_R^{1/2} LV \end{bmatrix} y - \begin{bmatrix} W_F^{1/2} b \\ 0 \end{bmatrix} \right\|_2^2, \tag{4.3}$$

where we, for notational simplicity, suppress the subscript k of the subspace matrix V and the superscripts $k+1$ of the weighting matrices. Let $V \in \mathbb{R}^{n\times d}$, $d = k+l \ll n$, and introduce the QR factorizations

$$W_F^{1/2} AV = Q_A R_A \quad \text{with} \quad Q_A \in \mathbb{R}^{m\times d}, \quad R_A \in \mathbb{R}^{d\times d}\,, \tag{4.4}$$

$$W_R^{1/2} LV = Q_L R_L \quad \text{with} \quad Q_L \in \mathbb{R}^{s\times d}\,, \quad R_L \in \mathbb{R}^{d\times d}\,. \tag{4.5}$$

Thus, the matrices Q_A and Q_L have orthonormal columns and the matrices R_A and R_L are upper triangular. Substituting these factorizations into (4.3) yields the minimization problem

$$\min_{y\in\mathbb{R}^{k+l}} \left\| \begin{bmatrix} R_A \\ \mu^{1/2} R_L \end{bmatrix} y - \begin{bmatrix} Q_A^T W_F^{1/2} b \\ 0 \end{bmatrix} \right\|_2^2$$

with the associated normal equations

$$\big(R_A^T R_A + \mu R_L^T R_L\big) y = R_A^T Q_A^T W_F b\,. \tag{4.6}$$

Recalling the definition (2.15) of the matrix $T(W_R, W_F)$ and using the fact that $x^{(k+1)} = V_k y^{(k+1)}$, the residual $r^{(k+1)}$ of (2.16) can be computed according to

$$\begin{aligned} r^{(k+1)} &= T\big(W_F, W_R\big)\, x - A^T W_F b \\ &= \big(A^T W_F A + \mu L^T W_R L\big)\, x - A^T W_F b \\ &= A^T W_F \big(AVy - b\big) + \mu L^T W_R \big(LVy\big)\,. \end{aligned} \tag{4.7}$$

The subspace $\mathcal{V}_k$ is then expanded to $\mathcal{V}_{k+1}$ in the same fashion as in the GKS method, that is, by adding the new unit vector v_{new} obtained by normalizing the residual $r^{(k+1)}$ in (4.7). The main steps of the GKSpq algorithm are described by Algorithm 3.

Algorithm 3. The GKSpq method for the solution of the ℓ_p-ℓ_q problem (1.1).

```
Inputs: $A \in \mathbb{R}^{m\times n}$, $L \in \mathbb{R}^{s\times n}$, $b \in \mathbb{R}^m$, $0 < p, q \le 2$, $\mu > 0$
Output: Approximate solution $x^*$ of (1.1)
1. Initialize:  $x^{(0)} = b$ or $x^{(0)} = \left(A^T A + \mu L^T L\right)^{-1} A^T b$
                $V_0 \in \mathbb{R}^{n\times l}$ such that $V_0^T V_0 = I_l$
2. for  $k = 0, 1, \ldots$ until convergence do
3.     Compute weighting matrices $W_F^{(k+1)}$ and $W_R^{(k+1)}$ by (2.12) and (2.13)
4.     Compute solution $y^{(k+1)}$ of reduced problem (4.6)
5.     Compute residual $r^{(k+1)}$ of unreduced problem (2.14) by (4.7)
6.     Compute new basis vector $v_{\text{new}}$ by normalizing residual $r^{(k+1)}$
7.     Enlarge subspace and update matrices:
           $V_{k+1} = [V_k, v_{\text{new}}]$,   $AV_{k+1} = [AV_k, Av_{\text{new}}]$,   $LV_{k+1} = [LV_k, Lv_{\text{new}}]$
8. end for
9. Determine approximate solution $x^* = V_k y^{(k+1)}$
```

We briefly comment on the computational cost of Algorithm 3. First, we note that the computation of an orthonormal basis for the initial subspace $\mathcal{V}_0 := \mathcal{K}_l(A^T A, A^T b)$ does not require the matrix $A^T A$ to be explicitly formed; it suffices to carry out $l-1$ steps of Golub–Kahan bidiagonalization of the matrix A with initial vector b. Moreover, for efficiency reasons, the matrices AV and LV are stored and updated during the iterations. The cost for computing the diagonal weighting matrices in step 3 of Algorithm 3 is negligible. In fact, the vectors $AV_k y^{(k+1)} - b$ and $LV_k y^{(k+1)}$ have already been computed in (4.7) when evaluating the residual $r^{(k+1)}$ in the previous iteration and are reused. The computation of the QR factorizations (4.4) and (4.5) required in step 4 of Algorithm 3 demands about $2m(k+l)^2$ and $2s(k+l)^2$ arithmetic floating point operations (flops), respectively. Since $k + l \ll n$, both flop counts are negligible in comparison with the cost for evaluating MVPs with the large matrices A, L, and their transposes. The computational effort required in step 5 to determine the residual by (4.7) is dominated by two MVP evaluations, one with A^T and one with L^T. Finally, in step 7 of the algorithm, the matrices AV and LV are updated for use in the next iteration with a computational cost of two MVPs, one with A and one with L. In summary, the overall computational cost for K iterations with the GKSpq algorithm is dominated by the work required to evaluate $4K$ MVPs.

5. Convergence analysis.

In this section we discuss the convergence of the GKSpq method. Its main steps are described by Algorithm 3. Our discussion relies on convergence results reported in [7] for the IRN/HQ method.

To secure that the GKSpq algorithm does not break down, we require that the solution $y^{(k+1)}$ of the reduced least-squares problem (4.1) exists and is unique or, equivalently, that the coefficient matrix of the normal equations (4.2) is of full rank. We recall that the diagonal weighting matrices $W_F^{(k+1)}$ and $W_R^{(k+1)}$, defined by (2.12) and (2.13), have strictly positive diagonal entries and that the rectangular subspace matrix V_k has full rank. Therefore, in order for (4.2) to be solvable for any such matrices $W_F^{(k+1)}$, $W_R^{(k+1)}$, and V_k, we have to assume that

$$\mathrm{Ker}\left(A^TA\right) \cap \mathrm{Ker}\left(L^TL\right) = \{0\}\,, \tag{5.1}$$

where Ker(M) denotes the null space of the matrix M. This requirement typically is satisfied. For instance, in image restoration problems A represents a blurring operator, which is a low-pass filter, whereas the regularization matrix L usually is a difference operator and, hence, is a high-pass filter. Chan and Liang [7] point out that the property (5.1) also guarantees that the IRN method does not break down.

Both the IRN and GKSpq methods rely on the three main computational steps (2.7)–(2.9) but differ in how the solution of the step (2.9) is computed. For the purpose of discussing convergence, it is convenient to consider the computations of (2.7)–(2.9) as one step. Toward this end, we substitute the expressions for $v^{(k+1)}$ and $z^{(k+1)}$, given by (2.10)–(2.11), into (2.9). Then the minimization problem (2.5) can be expressed by substituting the solutions of the problems (2.7) and (2.8). We obtain that at iteration k, both the IRN and GKSpq algorithms compute the next iterate, $x^{(k+1)}$, by minimizing the following functional over $x \in \mathbb{R}^n$, with $x^{(k)} \in \mathbb{R}^n$ a constant parameter vector:

$$\begin{aligned} Q(x;x^{(k)}) = \frac{1}{p}\sum_{i=1}^{m}\left(\frac{p}{2}|(Ax^{(k)})_i - b_i|^{p-2}|(Ax)_i - b_i|^2 + \frac{2-p}{2}|(Ax^{(k)})_i - b_i|^p\right)\\ + \frac{\mu}{q}\sum_{j=1}^{s}\left(\frac{q}{2}|(Lx^{(k)})_j|^{q-2}|(Lx)_j|^2 + \frac{2-q}{2}|(Lx^{(k)})_j|^q\right). \end{aligned} \tag{5.2}$$

This functional has some important properties, which are summarized in the following proposition.

PROPOSITION 5.1. *Let $0 < p, q \leq 2$ and assume that* (5.1) *holds. Then, for any $x^{(k)} \in \mathbb{R}^n$, the functional $Q(x;x^{(k)})$ defined in* (5.2) *is a strictly convex tangent majorant function at $x^{(k)}$ of the ℓ_p-ℓ_q functional $\mathcal{J}(x)$ defined in* (2.4) *with $\Phi = L$, that is,*

- $Q(x;x^{(k)}) \geq \mathcal{J}(x) \quad \forall\, x \in \mathbb{R}^n$,
- $Q(x;x^{(k)}) = \mathcal{J}(x)$ *at* $x = x^{(k)}$,
- $\nabla_x Q(x;x^{(k)}) = \nabla\mathcal{J}(x)$ *at* $x = x^{(k)}$,
- $\nabla_x^2 Q(x;x^{(k)})$ *is positive definite* $\forall\, x \in \mathbb{R}^n$.

Proof. The stated properties can be shown by using results from [7, Lemma 2]. □

Thus, in iteration k, the IRN and GKSpq methods replace the minimization of the original convex or nonconvex ℓ_p-ℓ_q functional $\mathcal{J}(x)$ by minimizing the simpler *surrogate* functional $Q(x;x^{(k)})$, which is convex and majorizes $\mathcal{J}(x)$.

Using the definition (5.2) of $Q(x; x^{(k)})$, the IRN algorithm can be written as

$$x^{(k+1)} = \arg\min_{x \in \mathbb{R}^n} Q(x; x^{(k)}) \; , \qquad k = 0, 1, \dots ,$$

while the GKSpq algorithm can be expressed as

$$x^{(k+1)} = \begin{cases} \arg\min\limits_{x \in \mathcal{V}_k} Q(x; x^{(k)}) & \text{for } \; k = 0, 1, \dots, n-l-1, \\ \arg\min\limits_{x \in \mathbb{R}^n} Q(x; x^{(k)}) & \text{for } \; k = n-l, n-l+1, \dots . \end{cases} \tag{5.3}$$

Here $l \geq 1$ is the dimension of the user-specified initial subspace $\mathcal{V}_0$, and $\mathcal{V}_k$ is the generalized Krylov subspace generated at iteration k. To justify (5.3), we recall that the subspace $\mathcal{V}_k$, in which in the kth iteration the GKSpq algorithm searches for the new approximate solution $x^{(k+1)}$, is of dimension $l + k$. Hence, from iteration $n - l$ and onward, the subspace $\mathcal{V}_k$ is equivalent to the whole space $\mathbb{R}^n$ and cannot be enlarged further. Therefore, for iterations $k \geq n - l$, the GKSpq method is equivalent to the IRN method with initial approximate solution $x^{(n-l-1)}$. This implies that the convergence behavior of the GKSpq and IRN algorithms is the same as $k \to \infty$.

In the following two subsections, we analyze the convergence of the GKSpq algorithm and first show that the sequence $\{\mathcal{J}(x^{(k)})\}_{k=1}^{\infty}$ of values achieved by the functional (1.1) is monotonically decreasing and convergent for any $0 < p, q \leq 2$. Subsequently, we discuss the convergence of the sequence of computed approximate solutions $\{x^{(k)}\}_{k=1}^{\infty}$ for $1 \leq p, q \leq 2$.

5.1. Monotonicity and convergence of $\mathcal{J}(x^{(k)})$ for $0 < p, q \leq 2$.

We first show the following result.

Lemma 5.2. *Let $0 < p, q \leq 2$ and assume that* (5.1) *holds. Let $\{x^{(k)}\}_{k=1}^{\infty}$ denote the sequence of approximate solutions generated by the GKSpq method using* (5.3). *Then, for any initial approximate solution $x^{(0)} \in \mathbb{R}^n$ and any $k \geq 1$, we have*

$$Q(x^{(k+1)}; x^{(k)}) \leq Q(x^{(k)}; x^{(k)}) \; . \tag{5.4}$$

Proof. We show (5.4) by considering the cases $1 \leq k < n - l$ and $k \geq n - l$ separately, starting with the former. Recall that generically at iteration $k-1$, the approximate solution $x^{(k)}$ computed by the GKSpq algorithm (5.3) lives in the subspace $\mathcal{V}_{k-1}$. Moreover, by construction $\mathcal{V}_{k-1} \subseteq \mathcal{V}_k \subseteq \mathbb{R}^n$. Given $x^{(k)}$, the new approximate solution $x^{(k+1)}$ is determined by minimizing $Q(x; x^{(k)})$ with the constraint that x belongs to the subspace $\mathcal{V}_k$; cf. (5.3). Since $x^{(k)} \in \mathcal{V}_k$ and $x^{(k+1)}$ is the unique minimizer of $Q(x; x^{(k)})$ for $x \in \mathcal{V}_k$, it follows that (5.4) holds.

We turn to the case when $k \geq n - l$. Then, given $x^{(k)}$, the GKSpq algorithm (5.3) computes the new approximate solution $x^{(k+1)}$ as the unique unconstrained global minimizer of $Q(x; x^{(k)})$, and (5.4) follows. □

Theorem 5.3. *Let $0 < p, q \leq 2$, assume that* (5.1) *holds, and let $\{x^{(k)}\}_{k=1}^{\infty}$ denote the sequence of approximate solutions generated by GKSpq method* (5.3). *Then, for any initial approximate solution $x^{(0)} \in \mathbb{R}^n$, the sequence $\{\mathcal{J}(x^{(k)})\}_{k=1}^{\infty}$ of values is monotonically decreasing and convergent where $\mathcal{J}$ is defined by* (2.4).

Proof. The sequence $\{\mathcal{J}(x^{(k)})\}_{k=1}^{\infty}$ is bounded from below by zero and is monotonically decreasing, since we can write

$$\mathcal{J}(x^{(k+1)}) \leq Q(x^{(k+1)}; x^{(k)}) \leq Q(x^{(k)}; x^{(k)}) = \mathcal{J}(x^{(k)}) \; .$$

Here the first inequality and last equality follow from the fact that $Q(x; x^{(k)})$ is a tangent majorant of $\mathcal{J}(x)$, i.e., they follow from the first two properties of Lemma 5.1.

The second inequality is a consequence of Lemma 5.2. Finally, since the sequence $\{\mathcal{J}(x^{(k)})\}_{k=1}^{\infty}$ is monotonically decreasing and bounded from below, it is convergent. □

5.2. Convergence of $x^{(k)}$ for $1 \leq p, q \leq 2$. When $0 < p, q < 1$, the ℓ_p-ℓ_q minimization problem (1.1) is not guaranteed to be convex. We therefore analyze the convergence of the sequence $x^{(k)}$, $k = 1, 2, \ldots$, generated by the GKSpq method for $1 \leq p, q \leq 2$ only. Since after $n - l$ iterations the GKSpq method turns into the IRN method, the convergence behavior of the sequence $\{x^{(k)}\}_{k=1}^{\infty}$ is the same as for sequences generated by the IRN method. Therefore, the convergence results shown in [7] for the IRN method also hold for the GKSpq method. We state the following convergence result from [7].

THEOREM 5.4. *Let $1 \leq p, q \leq 2$, assume that* (5.1) *holds, and let $\{x^{(k)}\}_{k=1}^{\infty}$ denote the sequence of approximate solutions determined by the GKSpq method* (5.3). *Then, for any initial approximate solution $x^{(0)} \in \mathbb{R}^n$, we have the following:*

(a) $\lim_{k\to\infty} \|x^{(k)} - x^{(k-1)}\|_2 = 0$.

(b) *The sequence $\{x^{(k)}\}_{k=1}^{\infty}$ converges to the unique minimizer of the ℓ_p-ℓ_q functional* (2.4).

An analysis of the behavior of $x^{(k)}$ for $k \leq n - l$ is beyond the scope of this paper and will be considered in future work. The following section provides some illustrations of this behavior.

6. Numerical examples. We compare the performances of the GKSpq method described by Algorithm 3 and of the IRN method outlined by Algorithm 1 when applied to the restoration of two gray-scale test images `cameraman` and `brain` depicted in Figures 1(b) and 2(b). These images are synthetically corrupted by blur and noise. They are represented by arrays of 256×256 pixels stored columnwise in vectors in $\mathbb{R}^n$ with $n = 65536$. Let $\bar{x} \in \mathbb{R}^n$ represent the original blur- and noise-free image. This image is assumed not to be available. A block Toeplitz with Toeplitz blocks blurring matrix $A \in \mathbb{R}^{n\times n}$ is generated with the function `blur` from [13]. This function has the parameters `band` and `sigma`, which determine the half-bandwidth of each Toeplitz block in A and the standard deviation of the underlying Gaussian point spread function, respectively. The blurred image $A\bar{x}$ is corrupted by different types and amounts of noise to obtain the image $b \in \mathbb{R}^n$. This image is assumed to be known. It is our aim to determine an accurate approximation x^* of $\bar{x}$, given A and b, by using the ℓ_p-ℓ_q model (1.1).

We compare the GKSpq and IRN methods both in terms of accuracy and efficiency. The former is measured by the relative error e and by the signal-to-noise ratio (SNR) defined by

$$e(x^*, \bar{x}) \, := \, \frac{\|x^* - \bar{x}\|_2}{\|\bar{x}\|_2} \, , \quad \mathrm{SNR}(x^*, \bar{x}) \, := \, 10 \log_{10} \frac{\|\bar{x} - E(\bar{x})\|_2^2}{\|x^* - \bar{x}\|_2^2} \ (\mathrm{dB}) \, , \tag{6.1}$$

where $E(\bar{x})$ denotes the mean gray-level of the uncontaminated image $\bar{x}$. These quantities provide quantitative measures of the quality of the restored image: a large SNR-value and a small e-value indicate that x^* is an accurate approximation of $\bar{x}$.

Computational efficiency is measured in terms of the total number of MVP evaluations with the matrices A, A^T, L, L^T required by the algorithms to satisfy the following stopping criterion. The (outer) iterations of the IRN and GKSpq algorithms are terminated as soon as the relative error of the computed approximate solution $x^{(k)}$ drops below a user-specified threshold $\tau > 0$, i.e., when

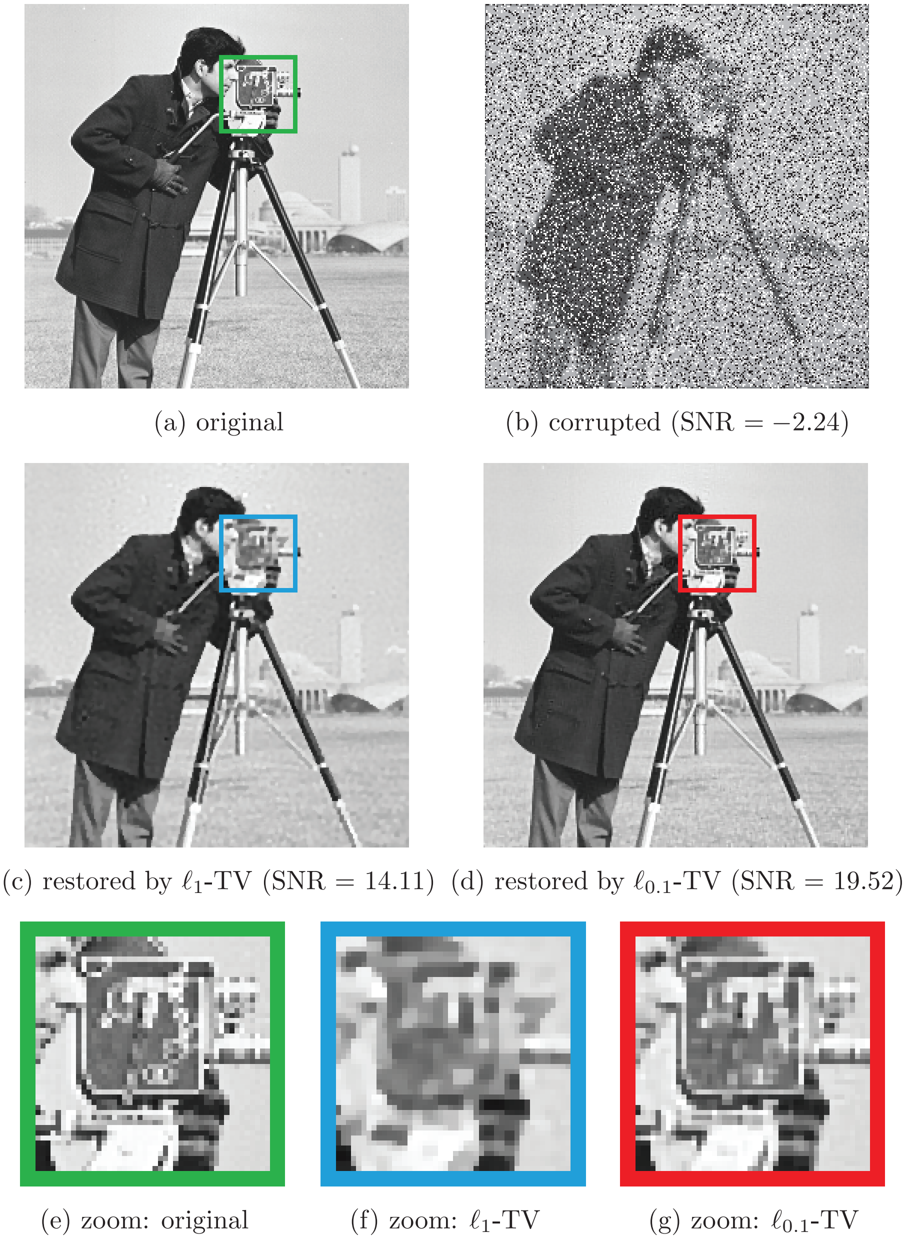
$$\mathrm{e}(x^{(k)}, \bar{x}) < \tau \; . \tag{6.2}$$

Results for the IRN algorithm are computed by using the freely available implementation in the NUMIPAD library [26]. This code implements the variable auto-adapted CG tolerance policy that yields the best timings [27]. The IRN implementation requires the user to specify the regularization parameter. We therefore choose the regularization parameter empirically to give the most accurate restorations, i.e., restorations with the highest SNR-values. To be able to compare the restorations determined by the IRN and GKSpq methods, we choose the regularization parameter in the same manner for the latter method. The GKSpq algorithm uses the initial search space $\mathcal{V}_0 = \mathcal{K}_1(A^T A, A^T b)$. All computations are carried out in MATLAB with about 15 significant decimal digits.

Example 1. We consider the restoration of contaminated test images `cameraman` and `brain` that have been degraded by Gaussian blur with different parameters `band` and `sigma`, and by salt-and-pepper noise of different intensity. Recall that salt-and-pepper noise corrupts images by changing a given percentage of pixels into either the minimum or maximum possible gray-level-value with equal probability. The other pixels are left unchanged. We consider the popular TV-regularized ℓ_p-TV restoration model (1.4), which corresponds to letting $q = 1$, $\Phi = \Phi_{\mathrm{TV}}$, and $0 < p \leq 2$ in (1.1). Specifically, we consider the two models ℓ_1-TV and $\ell_{0.1}$-TV. The ℓ_1-TV model represents the standard choice for the restoration of images corrupted by salt-and-pepper noise, see [7], since it forces sparsity of the residual while keeping the functional convex. The $\ell_{0.1}$-TV model is not convex but holds the potential to better induce sparsity. We are interested in comparing the two models in terms of restoration quality and computational effort required by the IRN and GKSpq minimization algorithms.

Figures 1 and 2 show restorations obtained by the GKSpq algorithm when applied to the test images. The degraded images in Figures 1(b) and 2(b) were obtained from the original blur- and noise-free images in Figures 1(a) and 2(a), respectively, by first applying Gaussian blur with parameters `band` $= 5$ and `sigma` $= 1.5$, and then corrupting 30% of the pixels by salt-and-pepper noise. Figures 1(c), 2(c) and Figures 1(d), 2(d) depict restorations obtained by using the GKSpq method to solve the ℓ_1-TV and $\ell_{0.1}$-TV models, respectively. For both test images, the latter model yields a significantly more accurate restoration than the former both in terms of visual quality and SNR-value. The SNR-values are displayed in Tables 1 and 2.

Tables 1 and 2 report quantitative results of the performances of the IRN and GKSpq methods when applied to the ℓ_1-TV and $\ell_{0.1}$-TV models for different choices of the parameters `band` and `sigma` that define the Gaussian blur and for different percentages of pixels corrupted by salt-and-pepper noise. Columns 4 to 7 show, from left to right, the value of the regularization parameter μ used for each example, the value of the threshold τ for the relative error used to terminate the iterations according to (6.2), and the total number of MVP evaluations as well as the number of outer iterations (in parentheses) required by the algorithms to satisfy the stopping criterion or to yield a relative change smaller than 10^{-4} in the successive computed restorations. Thus, the iterations are terminated when (6.2) is satisfied or when $\|x^{(k+1)} - x^{(k)}\|_2 / \|x^{(k)}\|_2 < 10^{-4}$. The SNR-values of the so-obtained computed restorations are reported in the last two columns of the tables. The dimension of the solution subspace generated by the GKSpq algorithm at convergence is not reported explicitly but can be easily obtained as the sum of the dimension of the initial subspace and the number of iterations carried out. The latter are reported in parentheses



(a) original

(b) corrupted (SNR $= -2.24$)

(c) restored by ℓ_1-TV (SNR $= 14.11$)

(d) restored by $\ell_{0.1}$-TV (SNR $= 19.52$)

(e) zoom: original

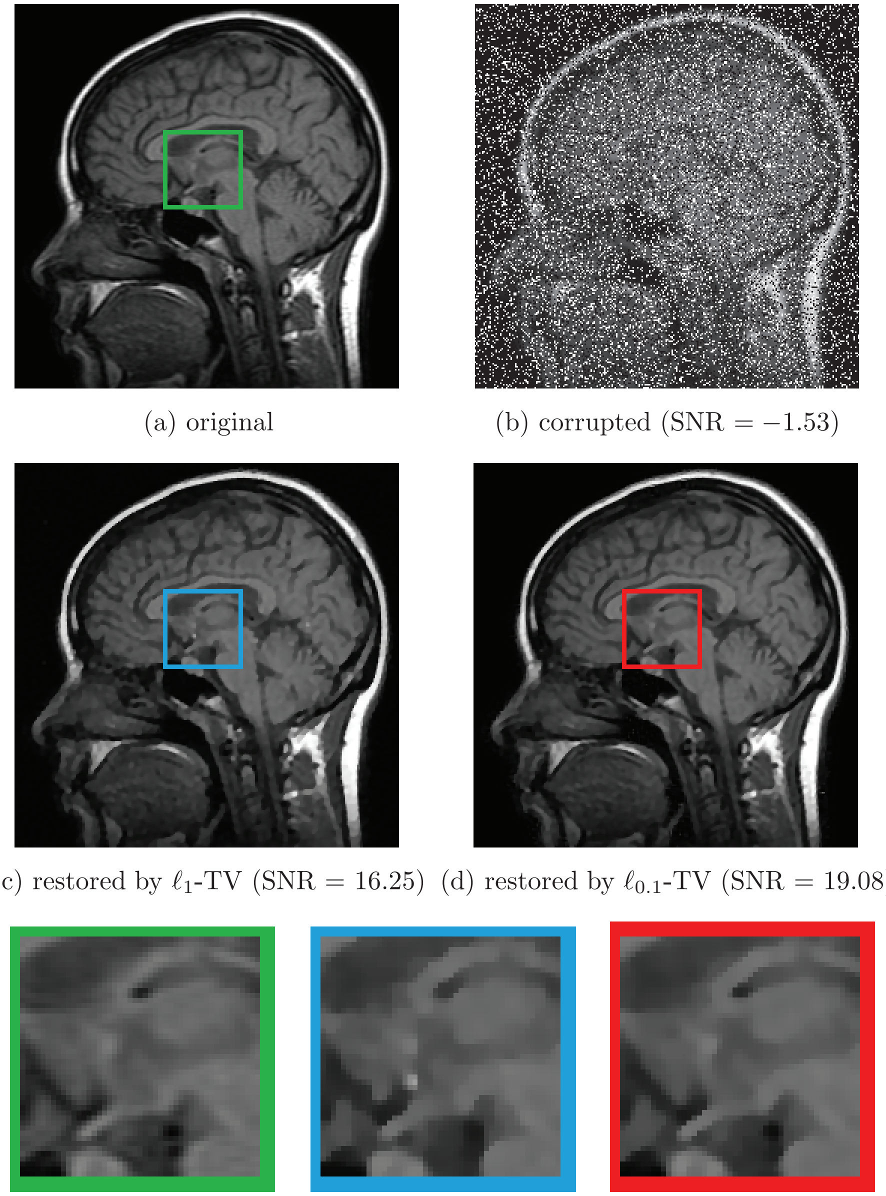
(f) zoom: ℓ_1-TV

(g) zoom: $\ell_{0.1}$-TV

FIG. 1. *Example* 1. *Restoration results obtained by the GKSpq algorithm applied to the* `cameraman` *image that has been contaminated by Gaussian blur with* `band` $= 5$ *and* `sigma` $= 1.5$, *and salt-and-pepper noise corrupting* 30% *of the pixels.*

in the seventh column of Tables 1 and 2. We recall that the GKSpq method increases the dimension of the solution subspace by one in each iteration. The dimension of the initial solution subspace is $l = 1$. The tables illustrate both the superiority of the $\ell_{0.1}$-TV model in terms of accuracy of the restoration and the faster convergence of the GKSpq method.

Figure 3 shows the computational efficiency of the IRN and GKSpq methods. We plot efficiency curves for some of the parameter combinations of Table 1 for the



(a) original

(b) corrupted (SNR $= -1.53$)

(c) restored by ℓ_1-TV (SNR $= 16.25$)

(d) restored by $\ell_{0.1}$-TV (SNR $= 19.08$)

(e) zoom: original

(f) zoom: ℓ_1-TV

(g) zoom: $\ell_{0.1}$-TV

FIG. 2. *Example* 1. *Restoration results obtained by the GKSpq algorithm applied to a test MRI* `brain` *image that has been contaminated by Gaussian blur with* `band` $= 5$ *and* `sigma` $= 1.5$, *and salt-and-pepper noise corrupting* 30% *of the pixels.*

IRN and GKSpq methods. Each sub-plot of Figure 3 displays for both methods $\log_{10}(e^{(k)})$ as a function of the number of MVP evaluations, where $e^{(k)}$ denotes the relative error (6.1) in the restored image available after k steps. The graphs of Figure 3 show the GKSpq method to require fewer MVP evaluations than the IRN method to reduce the error in all contaminated images. In fact, the GKSpq method requires about one third of the MVP evaluations needed by the IRN method to reduce the

Table 1

Example 1. *Comparison of the IRN and GKSpq algorithms applied to the minimization of the ℓ_1-TV and $\ell_{0.1}$-TV functionals for the restoration of* `cameraman` *test images corrupted by different kinds of Gaussian blur and salt-and-pepper noise.*

Blur		Noise	Parameters		MVPs (outer its)		SNR	
`Band`	`sigma`	%	μ	τ	IRN	GKSpq	IRN	GKSpq
ℓ_1-TV								
3	1	10	0.050	0.0450	206 (7)	76 (19)	18.05	18.06
		20	0.130	0.0641	264 (8)	92 (23)	15.88	15.89
		30	0.190	0.0776	380 (10)	128 (32)	14.22	14.22
5	1.5	10	0.013	0.0647	308 (6)	136 (34)	15.80	15.84
		20	0.025	0.0715	294 (7)	112 (28)	14.92	14.93
		30	0.050	0.0787	364 (10)	108 (27)	14.10	14.11
7	2	10	0.005	0.0779	516 (6)	200 (50)	14.18	14.19
		20	0.010	0.0851	462 (7)	156 (39)	13.42	13.43
		30	0.015	0.0911	436 (8)	124 (31)	12.83	12.83
$\ell_{0.1}$-TV								
3	1	10	0.5	0.0219	1154 (9)	420 (105)	25.23	25.12
		20	3.2	0.0372	498 (7)	176 (44)	20.60	22.16
		30	10.1	0.0436	580 (10)	192 (48)	19.23	19.95
5	1.5	10	0.1	0.0321	2484 (8)	928 (232)	21.89	21.61
		20	0.3	0.0374	1728 (8)	632 (158)	20.57	20.46
		30	0.4	0.0416	1778 (9)	588 (147)	19.64	19.52
7	2	10	0.06	0.0415	4418 (9)	1460 (365)	19.66	19.66
		20	0.11	0.0469	2936 (8)	1032 (258)	18.60	18.35
		30	0.17	0.0508	2994 (9)	848 (212)	17.89	17.76

error in the available contaminated images in all examples with ℓ_1-TV and $\ell_{0.1}$-TV models.

Finally, in Figure 4 we provide empirical evidence of the numerical convergence of the proposed GKSpq algorithm applied to the ℓ_1-TV and $\ell_{0.1}$-TV models for the restoration of the `cameraman` test image perturbed by Gaussian blur with `band` $= 5$ and `sigma` $= 1.5$, and salt-and-pepper noise corrupting 20% of the pixels. (See the fifth rows in Table 1.) In particular, the graphs reported in the first and second rows of Figure 4 display, respectively, the quantities $\log_{10}(-(J^{(k)} - J^{(k-1)})/J^{(k-1)})$ and $\log_{10}(\|x^{(k)} - x^{(k-1)}\|_2/\|x^{(k-1)}\|_2)$ as functions of the iteration count k, where $J^{(k)}$ denotes the value $J(x^{(k)})$ achieved after k iterations by the ℓ_p-ℓ_q functional defined in (2.4). These graphs show that numerical convergence of the GKSpq algorithm is fast both in terms of the computed approximate solutions $x^{(k)}$ and in terms of the corresponding functional values $J(x^{(k)})$. Moreover, the plots in the first row of Figure 4 show that the sequence of function values $J(x^{(k)})$, $k = 1, 2, \ldots$, is monotonically decreasing, thus confirming experimentally the theoretical result in Theorem 5.3. Similar convergence behavior can be observed for all the other examples.

Example 2. We consider the restoration of the test image `cameraman` corrupted by Gaussian blur with `band` $= 5$ and `sigma` $= 1.5$, and by additive zero-mean white Gaussian noise with different standard deviations σ. It is well known that for this kind of noise, accurate restorations can be achieved with the ℓ_p-ℓ_q restoration model (1.1) for $p = 2$. We refer to this model as the ℓ_2-ℓ_q model and compare the IRN and the GKSpq methods when applied to the minimization of the ℓ_2-ℓ_q functional for

Table 2

Example 1. *Comparison of the IRN and GKSpq algorithms applied to the minimization of the ℓ_1-TV and $\ell_{0.1}$-TV functionals for the restoration of* `brain` *test images corrupted by different kinds of Gaussian blur and salt-and-pepper noise.*

Blur		Noise	Parameters		MVPs (outer its)		SNR	
`Band`	`sigma`	%	μ	τ	IRN	GKSpq	IRN	GKSpq
ℓ_1-TV								
3	1	10	0.150	0.0594	200 (6)	60 (15)	21.59	21.60
		20	0.250	0.0774	354 (9)	112 (28)	19.51	19.52
		30	0.300	0.0975	398 (9)	168 (42)	17.52	17.52
5	1.5	10	0.015	0.0901	208 (4)	36 (9)	17.76	17.79
		20	0.035	0.0977	206 (5)	44 (11)	17.04	17.05
		30	0.060	0.1092	212 (6)	52 (13)	16.25	16.25
7	2	10	0.010	0.1104	280 (4)	52 (13)	16.35	16.36
		20	0.020	0.1178	216 (4)	52 (13)	15.62	15.63
		30	0.040	0.1265	206 (5)	56 (14)	15.00	15.00
$\ell_{0.1}$-TV								
3	1	10	4	0.0347	392 (6)	128 (32)	26.16	26.38
		20	8	0.0440	356 (6)	188 (47)	24.27	24.63
		30	11	0.0559	402 (7)	228 (57)	21.98	23.11
5	1.5	10	1.0	0.0568	558 (5)	108 (27)	21.99	22.40
		20	1.2	0.0702	610 (5)	124 (31)	20.17	20.24
		30	1.9	0.0813	566 (5)	156 (39)	18.74	19.08
7	2	10	1.0	0.0736	662 (5)	176 (44)	19.88	20.01
		20	1.2	0.0828	686 (5)	168 (42)	18.90	18.97
		30	2.0	0.0908	614 (5)	164 (41)	18.12	18.18

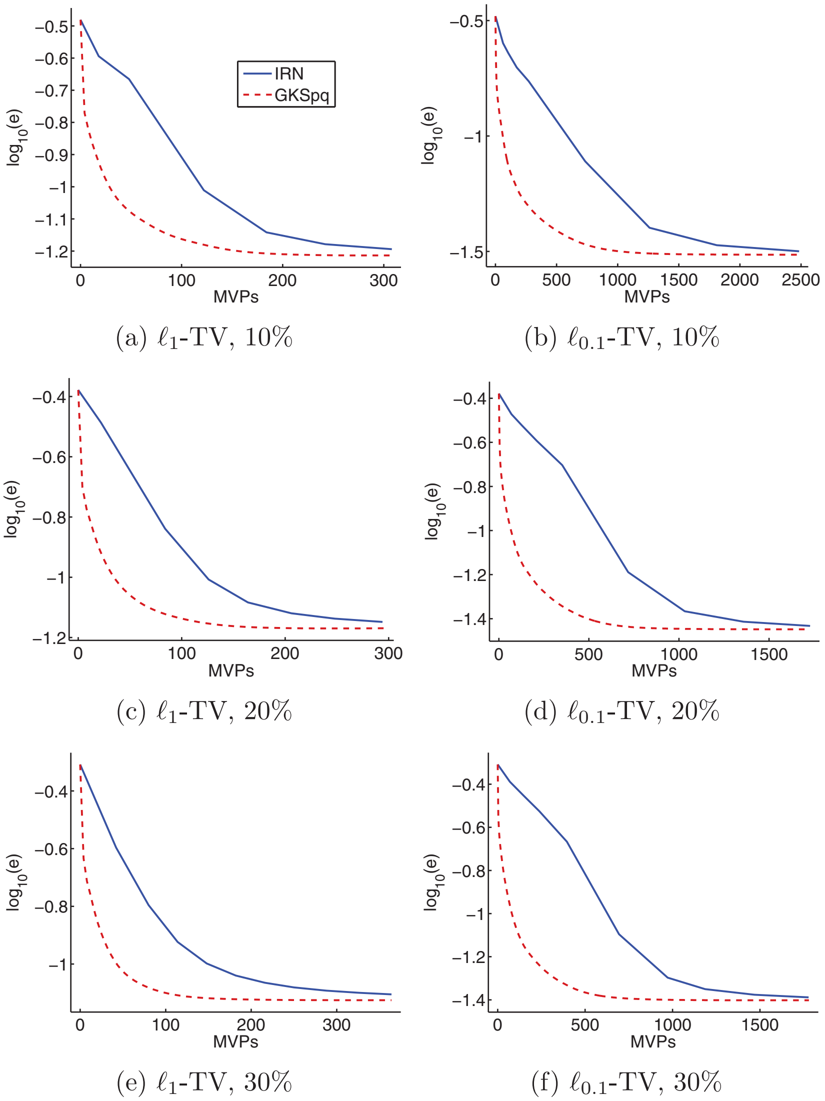
several values of q. Specifically, we let $q \in \{\frac{1}{2}, \frac{3}{2}, 2\}$. The regularization matrix L is chosen to be the first-order discrete derivative operator for two space dimensions,

$$L = \begin{bmatrix} L_1 & \otimes & I_d \\ I_d & \otimes & L_1 \end{bmatrix} \quad \text{with} \quad L_1 = \begin{bmatrix} -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix} \in \mathbb{R}^{(d-1)\times d} \,,$$

where I_d denotes the identity matrix of order d, d is the number of pixels in each row and column of the image considered, the matrix L_1 is bidiagonal, and $\otimes$ denotes the Kronecker product.

Table 3 reports results for the IRN and GKSpq methods for several q-values (first column), and for additive white Gaussian noise of different standard deviations σ (second column). The other columns display the value of the regularization parameter μ used for each test, the value of the threshold τ for the relative error used in the stopping criterion (6.2), and the total number of MVP evaluations needed to satisfy the stopping criterion. The last column of Table 3 reports the SNR-values for the restorations obtained by the IRN and GKSpq methods. Analogously to Example 1, the dimension of the Krylov subspace generated by the GKSpq algorithm at convergence is the sum of the number iterations carried out and the dimension of the initial solution subspace. The latter is equal to one.

The GKSpq method is seen to require fewer MVP evaluations than the IRN method. The quality of the restoration is higher for $q < 2$ than for $q = 2$. We remark that $q = 0.5$ corresponds to a nonconvex functional [8]. Therefore, the IRN



(a) ℓ_1-TV, 10%

(b) $\ell_{0.1}$-TV, 10%

(c) ℓ_1-TV, 20%

(d) $\ell_{0.1}$-TV, 20%

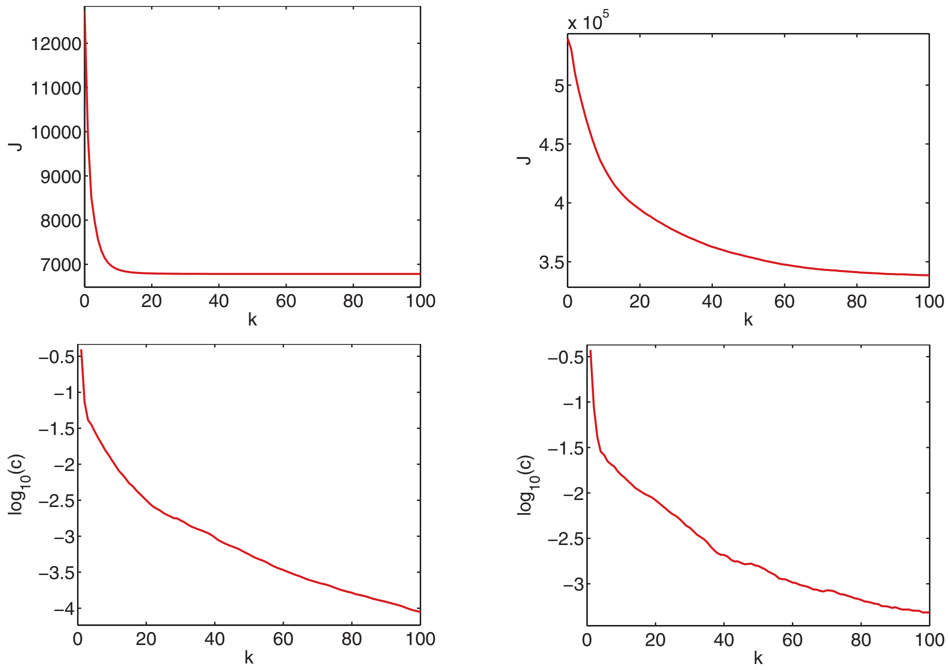
(e) ℓ_1-TV, 30%

(f) $\ell_{0.1}$-TV, 30%

FIG. 3. *Example* 1. *Logarithm of the relative error versus the number of matrix-vector product evaluations performed by the IRN and GKSpq algorithms when applied to the minimization of the* ℓ_1*-TV (left-hand side column) and* $\ell_{0.1}$*-TV (right-hand side column) functionals for the restoration of* `cameraman` *test images that have been contaminated by Gaussian blur with* `band` $= 5$ *and* `sigma` $= 1.5$*, and salt-and-pepper noise corrupting* 10% *(first row),* 20% *(second row), and* 30% *(third row) of the pixels.*

and GKSpq methods are not guaranteed to converge to the same restoration in this situation.

7. Conclusions. We presented a new method for the efficient solution of the ℓ_p-ℓ_q minimization problem based on successive orthogonal projections onto generalized Krylov subspaces of increasing dimensions. The subspaces are generated according to the iteratively reweighted least-squares strategy for the approximation of ℓ_p/ℓ_q-norms by weighted ℓ_2-norms. Computed examples with application to image



(a) ℓ_1-TV, 20%

(b) $\ell_{0.1}$-TV, 20%

Fig. 4. *Example* 1. *Convergence results for the GKSpq method applied to ℓ_1-TV (left-hand side column) and $\ell_{0.1}$-TV (right-hand side column) functionals for the restoration of the* `cameraman` *test image that has been contaminated by Gaussian blur with* `band` $= 5$ *and* `sigma` $= 1.5$*, and salt-and-pepper noise corrupting* 20% *of the pixels.*

Table 3

Example 2. *Comparison of the IRN and GKSpq algorithms applied to the minimization of the ℓ_2-ℓ_q functional for the restoration of* `cameraman` *test images contaminated by Gaussian blur with* `band` $= 5$ *and* `sigma` $= 1.5$*, and additive zero-mean white Gaussian noise with standard deviation σ.*

Model	Noise	Parameters		MVPs (outer its)		SNR	
q	σ	μ	τ	IRN	GKSpq	IRN	GKSpq
2	10	0.050	0.1293	12 (1)	12 (3)	11.60	11.60
	20	0.190	0.1191	8 (1)	8 (2)	10.50	10.50
	30	0.410	0.1049	14 (1)	8 (2)	9.79	9.79
1.5	10	0.012	0.0990	40 (2)	24 (6)	12.11	12.11
	20	0.038	0.1128	64 (2)	24 (6)	10.97	10.97
	30	0.080	0.1225	80 (2)	20 (5)	10.26	10.26
0.5	10	0.00025	0.1025	272 (2)	148 (37)	11.87	12.03
	20	0.00100	0.1171	372 (2)	180 (45)	10.65	10.93
	30	0.00250	0.1259	560 (2)	312 (78)	10.02	10.37

restoration indicate that subspaces of fairly small dimension suffice for the determination of high-quality restorations. The numerical experiments demonstrate that the proposed method requires a smaller number of matrix-vector product evaluations to yield restorations of required accuracy than the standard implementation of the iteratively reweighted least-squares method.

Acknowledgment. We would like to thank the referees for comments that led to improvements of the presentation.

REFERENCES

[1] A. Beck and M. Teboulle, *A fast iterative shrinkage-thresholding algorithm for linear inverse problems*, SIAM J. Imaging Sci., 2 (2009), pp. 183–202.

[2] D. Bertaccini, R. H. Chan, S. Morigi, and F. Sgallari, *An adaptive norm algorithm for image restoration*, in Proceedings of Scale Space and Variational Methods in Computer Vision, Lecture Notes in Comput. Sci. 6667, Springer, Berlin, 2012, pp. 194–205.

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

[4] D. Calvetti and L. Reichel, *Tikhonov regularization of large linear problems*, BIT, 43 (2003), pp. 263–283.

[5] D. Calvetti, S. Morigi, L. Reichel, and F. Sgallari, *Tikhonov regularization and the L-curve for large, discrete ill-posed problems*, J. Comput. Appl. Math., 123 (2000), pp. 423–446.

[6] E. J. Candes, J. Romberg, and T. Tao, *Robust uncertainty principle: Exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inform. Theory, 52 (2006), pp. 489–509.

[7] R. H. Chan and H. X. Liang, *Half-quadratic algorithm for ℓ_p-ℓ_q problems with applications to TV-ℓ_1 image restoration and compressive sensing*, in Proceedings of Efficient Algorithms for Global Optimization Methods in Computer Vision, Lecture Notes in Comput. Sci. 8293, Springer, Berlin, 2014, pp. 78–103.

[8] X. Chen, D. Ge, Z. Wang, and Y. Ye, *Complexity of unconstrained L_2-L_p minimization*, Math. Program., 143 (2014), pp. 371–383.

[9] T.-M.-T. Do and T. Artières, *Regularized bundle methods for convex and non-convex risks*, J. Mach. Learn. Res., 13 (2012), pp. 3539–3583.

[10] D. Donoho, *Compressed sensing*, IEEE Trans. Inform. Theory, 52 (2006), pp. 1289–1306.

[11] L. Eldén, *A weighted pseudoinverse, generalized singular values, and constrained least squares problems*, BIT, 22 (1982), pp. 487–502.

[12] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*, Kluwer, Dordrecht, The Netherlands, 1996.

[13] P. C. Hansen, *Regularization tools version* 4.0 *for MATLAB* 7.3, Numer. Algorithms, 46 (2007), pp. 189–194.

[14] M. E. Hochstenbach and L. Reichel, *An iterative method for Tikhonov regularization with a general linear regularization operator*, J. Integral Equations Appl., 22 (2010), pp. 463–480.

[15] M. E. Hochstenbach, L. Reichel, and X. Yu, *A Golub–Kahan-type reduction method for matrix pairs*, J. Sci. Comput., to appear.

[16] M. E. Kilmer, P. C. Hansen, and M. I. Español, *A projection-based approach to general-form Tikhonov regularization*, SIAM J. Sci. Comput., 29 (2007), pp. 315–330.

[17] S. Kindermann, *Convergence analysis of minimization-based noise level-free parameter choice rules for linear ill-posed problems*, Electron. Trans. Numer. Anal., 38 (2011), pp. 233–257.

[18] J. Lampe, L. Reichel, and H. Voss, *Large-scale Tikhonov regularization via reduction by orthogonal projection*, Linear Algebra Appl., 436 (2012), pp. 2845–2865.

[19] A. Lanza, S. Morigi, F. Sgallari, and A. J. Yezzi, *Variational image denoising based on autocorrelation whiteness*, SIAM J. Imaging Sci., 6 (2013), pp. 1931–1955.

[20] A. Lanza, S. Morigi, F. Sgallari, and A. J. Yezzi, *Variational image denoising while constraining the distribution of the residual*, Electron. Trans. Numer. Anal., 42 (2014), pp. 64–84.

[21] L. Laporte, R. Flamary, S. Canu, S. Déjean, and J. Mothe, *Nonconvex regularizations for features selection in ranking with sparse SVM*, IEEE Trans. Neural Netw. Learn. Syst., 25 (2014), pp. 1118–1130.

[22] Z. Liu, Z. Wei, and W. Sun, *An iteratively approximated gradient projection algorithm for sparse signal reconstruction*, Appl. Math. Comput., 228 (2014), pp. 454–462.

[23] M. Nikolova and R. H. Chan, *The equivalence of the half-quadratic minimization and the gradient linearization iteration*, IEEE Trans. Image Process., 16 (2007), pp. 5–18.

[24] L. Reichel and G. Rodriguez, *Old and new parameter choice rules for discrete ill-posed problems*, Numer. Algorithms, 63 (2013), pp. 65–87.

[25] L. Reichel, F. Sgallari, and Q. Ye, *Tikhonov regularization based on generalized Krylov subspace methods*, Appl. Numer. Math., 62 (2012), pp. 1215–1228.

[26] P. Rodríguez and B. Wohlberg, *Numerical Methods for Inverse Problems and Adaptive Decomposition (NUMIPAD)*, http://numipad.sourceforge.net/ (2009).

[27] P. Rodríguez and B. Wohlberg, *Efficient minimization method for a generalized total variation functional*, IEEE Trans. Image Process., 18 (2009), pp. 322–332.

[28] L. Rudin, S. Osher, and E. Fatemi, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.

[29] C. R. Vogel and M. E. Oman, *Fast, robust total variation-based reconstruction of noisy blurred images*, IEEE Trans. Image Process., 7 (1998), pp. 813–824.

[30] H. Voss, *An Arnoldi method for nonlinear eigenvalue problems*, BIT, 44 (2004), pp. 387–401.

[31] W. T. Yin, D. Goldfarb, and S. Osher, *Image cartoon-texture decomposition and feature selection using total variation regularized $L1$ functional*, in Proceedings of Variational, Geometric, and Level Set Methods in Computer Vision 2005, Lecture Notes in Comput. Sci. 3752, Springer, Berlin, 2005, pp. 73–84.

[32] W. T. Yin, D. Goldfarb, and S. Osher, *The total variation regularized $L1$ model for multiscale decomposition*, Multiscale Model. Simul., 6 (2007), pp. 190–211.

[33] Y. Zhao and D. Li, *Reweighted ℓ_1-minimization for sparse solutions to undetermined linear systems*, SIAM J. Optim., 22 (2012), pp. 1065–1088.