NOISE-REDUCING CASCADIC MULTILEVEL METHODS FOR LINEAR DISCRETE ILL-POSED PROBLEMS

S. MORIGI*, L. REICHEL†, AND F. SGALLARI‡

Abstract. Cascadic multilevel methods for the solution of linear discrete ill-posed problems with noise-reducing restriction and prolongation operators recently have been developed for the restoration of blur- and noise-contaminated images. This is a particular ill-posed problem. The multilevel methods were found to determine accurate restorations with fairly little computational work. This paper describes noise-reducing multilevel methods for the solution of general linear discrete ill-posed problems.

Key words. ill-posed problem, multilevel method, noise-reduction

1. Introduction. Many problems in science and engineering require the determination of the cause of an observed effect. These problems often can be formulated as Fredholm integral equations of the first kind with a smooth kernel h,

$$\int_{\mathbb{S}} h(s,t)x(t)dt = b(s), \qquad s \in \mathbb{S}. \tag{1.1}$$

This paper discusses solution methods for such integral equations. The methods discussed are tailored for the situation when $\mathbb{S}$ is a bounded interval; however, the methods also are applicable, mutatis mutandis, for more general compact sets $\mathbb{S} \in \mathbb{R}^j$.

The right-hand side function b represents the "observed effect" and is assumed to be such that (1.1) has a solution, which represents the "cause." The solution might not be unique. Let $\hat{x}$ denote the solution of minimal norm. The computation of $\hat{x}$ is an ill-posed problems, because $\hat{x}$ does not depend continuously on b. When $\mathbb{S}$ is a bounded interval, this is a consequence of the Riemann-Lebesgue lemma. Nice treatments of ill-posed problems and numerical methods for their solution are provided by Engl et al. [14] and Hansen [18].

Discretization of (1.1) gives a linear system of equations

$$A\boldsymbol{x} = \boldsymbol{b}, \qquad A \in \mathbb{R}^{n\times n}, \qquad \boldsymbol{b} \in \mathbb{R}^n, \tag{1.2}$$

with a matrix of ill-determined rank. In particular, the singular values of A "cluster" at the origin. Therefore, the matrix A is severely ill-conditioned. Linear systems of equations with a matrix of ill-determined rank commonly are referred to as *linear discrete ill-posed problems*. We assume the linear system (1.2) to be consistent and large enough to make the use of iterative solution methods attractive.

Let $A^\dagger$ denote the Moore-Penrose pseudoinverse of A. We are interested in determining the solution $\hat{\boldsymbol{x}}$ of (1.2) of minimal Euclidean norm. It can be expressed as $\hat{\boldsymbol{x}} = A^\dagger\boldsymbol{b}$. Due to the severe ill-conditioning of A, the vector $\hat{\boldsymbol{x}}$ is very sensitive to perturbations in $\boldsymbol{b}$.

The right-hand side $\boldsymbol{b}$ of (1.2) is obtained by discretizing the integral equation (1.1), e.g., by a Nyström method. However, since in applications the available right-

*Dept. of Math.-CIRAM, University of Bologna, Piazza Porta S. Donato 5, 40126 Bologna, Italy. E-mail: `morigi@dm.unibo.it`

†Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA. E-mail: `reichel@math.kent.edu`

‡Dept. of Math.-CIRAM, University of Bologna, Piazza Porta S. Donato 5, 40126 Bologna, Italy. E-mail: `sgallari@dm.unibo.it`

hand side vector represents observations, it typically is contaminated by a measurement error $\boldsymbol{e}^\delta \in \mathbb{R}^n$. We will refer to $\boldsymbol{e}^\delta$ as *noise*. Let $\boldsymbol{b}^\delta$ denote the available noise-contaminated approximation of $\boldsymbol{b}$. Then

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

Thus, our computational task is to determine an accurate approximation of the desired solution $\hat{\boldsymbol{x}}$ of (1.2) by computing an approximate solution of the available linear discrete ill-posed problem with the contaminated right-hand side $\boldsymbol{b}^\delta$,

$$A\boldsymbol{x} = \boldsymbol{b}^\delta. \tag{1.4}$$

This system might not be consistent. We remark that in view of the ill-conditioning of A and the error $\boldsymbol{e}^\delta$ in $\boldsymbol{b}^\delta$, the vector $A^\dagger \boldsymbol{b}^\delta$ typically is not a meaningful approximation of $\hat{\boldsymbol{x}}$.

Throughout this paper we use weighted Euclidean vector norms of the form

$$\|\boldsymbol{x}\| := \left(\frac{1}{n}\sum_{i=1}^{n}\left(x^{(i)}\right)^2\right)^{1/2}, \qquad \boldsymbol{x} = [x^{(1)}, x^{(2)}, \ldots, x^{(n)}]^T \in \mathbb{R}^n. \tag{1.5}$$

We assume that a fairly accurate estimate of the norm of the noise,

$$\delta := \|\boldsymbol{e}^\delta\|, \tag{1.6}$$

is available. This makes it possible to apply the discrepancy principle during the solution process. We illustrate in Section 5 how, in situations when no estimate of δ is known, a fairly accurate estimate can be determined by integrating a discrete version of the nonlinear Perona-Malik diffusion equation for a few time-steps [25].

Tikhonov regularization and truncated iteration are the most popular approaches to determining a meaningful approximate solution of large-scale linear discrete ill-posed problems (1.4) with a noise-contaminated right-hand side; see, e.g., [14, 16, 18] for discussions on these approaches. This paper describes multilevel methods, which apply truncated iteration with a Krylov subspace method to compute an approximate solution on each discretization level. For definiteness, consider the LSQR iterative method discussed in, e.g., [24, 31]. This method is an implementation of the conjugate gradient method applied to the normal equations associated with the linear system of equations (1.4). Let the initial approximate solution be $\boldsymbol{x}_0 := \boldsymbol{0}$. This choice of initial vector is common and will be used for all Krylov subspace methods. Then the ℓth approximate solution, $\boldsymbol{x}_\ell$, determined by LSQR satisfies

$$\|A\boldsymbol{x}_\ell - \boldsymbol{b}^\delta\| = \min_{\boldsymbol{x}\in\mathbb{K}_\ell(A^TA, A^T\boldsymbol{b}^\delta)} \|A\boldsymbol{x} - \boldsymbol{b}^\delta\|, \qquad \boldsymbol{x}_\ell \in \mathbb{K}_\ell(A^TA, A^T\boldsymbol{b}^\delta),$$

where $\mathbb{K}_\ell(A^TA, A^T\boldsymbol{b}^\delta) = \text{span}\{A^T\boldsymbol{b}^\delta, (A^TA)A^T\boldsymbol{b}^\delta, \ldots, (A^TA)^{\ell-1}A^T\boldsymbol{b}^\delta\}$ is a Krylov subspace. Here and below, we assume for ease of exposition that the dimension of a Krylov subspace $\mathbb{K}_\ell(\cdot,\cdot)$ equals its index ℓ. This requirement is satisfied in practice. The computation of $\boldsymbol{x}_\ell$ requires the evaluation of 2ℓ matrix-vector products, ℓ of which with A and ℓ with A^T. We refer to the iterative solution of (1.4) by the LSQR method as the *one-level LSQR method* in order to distinguish this method from the LSQR-based multilevel method introduced below. Similar terminology will be used for the other Krylov subspace methods considered.

The *discrepancy principle* prescribes that the iterations with LSQR be terminated as soon as an approximate solution, $\boldsymbol{x}_{\ell_\delta}$, such that

$$\|A\boldsymbol{x}_{\ell_\delta} - \boldsymbol{b}^\delta\| \leq c\delta \tag{1.7}$$

has been determined, where $c > 1$ is a user-specified constant independent of δ. Note that ℓ_δ increases as δ decreases. Nemirovskii [23] and Hanke [16] have shown that in an infinite-dimensional Hilbert space setting

$$\lim_{\delta \searrow 0} \|\boldsymbol{x}_{\ell_\delta} - \hat{\boldsymbol{x}}\| = 0. \tag{1.8}$$

LSQR therefore is said to be a *regularization method*. It is straightforward to show that (1.8) holds in finite dimensions.

When A is symmetric, it is attractive to apply the MR-II iterative method instead of LSQR. MR-II is a modification, described in [9, 16], of the standard conjugate gradient method. The ℓth approximate solution, $\boldsymbol{x}_\ell$, determined by MR-II when applied to (1.4) is characterized by

$$\|A\boldsymbol{x}_\ell - \boldsymbol{b}^\delta\| = \min_{\boldsymbol{x} \in \mathbb{K}_\ell(A, A\boldsymbol{b}^\delta)} \|A\boldsymbol{x} - \boldsymbol{b}^\delta\|, \qquad \boldsymbol{x}_\ell \in \mathbb{K}_\ell(A, A\boldsymbol{b}^\delta),$$

where $\mathbb{K}_\ell(A, A\boldsymbol{b}^\delta) = \text{span}\{A\boldsymbol{b}^\delta, A^2\boldsymbol{b}^\delta, \ldots, A^\ell\boldsymbol{b}^\delta\}$. The determination of $\boldsymbol{x}_\ell$ only requires the evaluation of $\ell + 1$ matrix vector products with A. Hanke [16] shows that the iterates determined by MR-II satisfy (1.8) in a Hilbert space.

GMRES is a popular iterative method for the solution of linear systems of equations with a not too ill-conditioned nonsymmetric matrix A; see, e.g., [31]. The ℓth iterate, $\boldsymbol{x}_\ell$, determined by GMRES satisfies

$$\|A\boldsymbol{x}_\ell - \boldsymbol{b}^\delta\| = \min_{\boldsymbol{x} \in \mathbb{K}_\ell(A, \boldsymbol{b}^\delta)} \|A\boldsymbol{x} - \boldsymbol{b}^\delta\|, \qquad \boldsymbol{x}_\ell \in \mathbb{K}_\ell(A, \boldsymbol{b}^\delta),$$

where $\mathbb{K}_\ell(A, \boldsymbol{b}^\delta) = \text{span}\{\boldsymbol{b}^\delta, A\boldsymbol{b}^\delta, \ldots, A^{\ell-1}\boldsymbol{b}^\delta\}$. The computation of $\boldsymbol{x}_\ell$ requires the evaluation of ℓ matrix-vector products with A. The property (1.8) is shown in [7] in a Hilbert space setting under certain conditions.

We also will consider the Range Restricted GMRES (RRGMRES) method. The ℓth iterate, $\boldsymbol{x}_\ell$, determined by RRGMRES is characterized by

$$\|A\boldsymbol{x}_\ell - \boldsymbol{b}^\delta\| = \min_{\boldsymbol{x} \in \mathbb{K}_\ell(A, A\boldsymbol{b}^\delta)} \|A\boldsymbol{x} - \boldsymbol{b}^\delta\|, \qquad \boldsymbol{x}_\ell \in \mathbb{K}_\ell(A, A\boldsymbol{b}^\delta).$$

Its computation demands $\ell + 1$ matrix-vector product evaluations with A. The limit (1.8) can be established in Hilbert space similarly as for GMRES. RRGMRES does not propagate the error $\boldsymbol{e}^\delta$ in $\boldsymbol{b}^\delta$ into the iterates as quickly as GMRES. This is discussed in [8] and it is also illustrated in the computed examples of Section 5.

Krylov subspace methods applied with the stopping criterion (1.7) seek to curb the propagation of the noise $\boldsymbol{e}^\delta$ into the the computed approximate solutions $\boldsymbol{x}_\ell$ by restricting the number of iterations. When there is much noise in $\boldsymbol{b}^\delta$, only few iterations are carried out. This limits the resolution of the computed approximate solution.

The present paper describes cascadic multilevel methods that use one of the Krylov subspace methods discussed above as basic iterative method and apply noise-reducing prolongation and restriction operators. The computations proceed from coarser to finer levels. An approximate solution is computed on each level with the Krylov subspace method chosen. The iterations are terminated on each level by a stopping criterion analogous (1.7). The multilevel methods so defined are regularization methods in a well-defined sense; see Section 2 for details. The computations are terminated as soon as an iterate in $\mathbb{R}^n$ that satisfies (1.7) has been determined. The advantages of cascadic multilevel methods, when compared with one-level Krylov subspace methods, include:

- Cascadic multilevel methods require fewer iterations in $\mathbb{R}^n$ than the corresponding one-level Krylov subspace methods. This depends on that cascadic multilevel methods determine a fairly accurate initial approximation of the solution in $\mathbb{R}^n$ by iterations on coarser levels, while the initial approximate solution for all one-level Krylov subspace methods is $\boldsymbol{x}_0 = \mathbf{0}$. For large-scale problems, the dominant computational effort for the multilevel methods are the iterations in $\mathbb{R}^n$, and specifically the evaluation of matrix-vector products with A and possibly A^T. The computations required for coarse-level iterations are negligible. Therefore the smaller number of iterations in $\mathbb{R}^n$ required by cascadic multilevel methods, when compared with one-level Krylov subspace methods, translates into a smaller arithmetic effort for the former methods.
- Multilevel methods allow the application of noise-reducing restriction and prolongation operators. This limits the propagation of the error $\boldsymbol{e}^\delta$ in the right-hand side $\boldsymbol{b}^\delta$ into the computed approximate solutions. The noise-reducing restriction and prolongation operators help cascadic multilevel methods determine better approximations of the desired solution $\hat{\boldsymbol{x}}$ of the noise-free linear system (1.2) than the corresponding one-level Krylov subspace iterative methods.

Restriction of the right-hand side $\boldsymbol{b}^\delta$ from finer to coarser levels is carried out by noise-reducing local weighted least-squares approximation. We compare two restriction operators, one of which is an adaption of the restriction operator used in [21], which, in turn, is inspired by a local least-squares method for noise-removal described by Buades et al. [5]. The other restriction operator, which is believed to be new, is well suited for the situation when the desired solution $\hat{\boldsymbol{x}}$ represents a smooth function. Details of these restriction operators are provided in Section 3.

Prolongation of the computed solution from coarser to finer levels is performed by local piecewise linear interpolation followed by nonlinear smoothing. The latter is achieved by time-integration of a discretized nonlinear Perona-Malik diffusion equation for a few time-steps [25]. The smoother is designed to remove noise, but preserve edges. Here and throughout this paper "edge" refers to any rapid spacial transition. Time-integration is carried out by an explicit method. This makes application of the smoother inexpensive. Further details are described in Section 4. Numerical examples, presented in Section 5, show our cascadic multilevel methods to require less computational work and yield more accurate approximations of $\hat{\boldsymbol{x}}$ than the corresponding one-level Krylov subspace iterative methods. Concluding remarks can be found in Section 6.

An important feature of noise-reducing cascadic multilevel methods is that they interlace approximate solution of (1.4) and noise removal. This reduces the propagation of the error $\boldsymbol{e}^\delta$ in $\boldsymbol{b}^\delta$, when compared with the solution by one-level Krylov subspace methods or by cascadic multilevel methods without noise reduction [29]. Noise-reducing cascadic multilevel methods were first developed in [21, 22] for image deblurring and denoising and many illustrations of the performance of different noise-reducing restriction and prolongation operators can be found in these references, as well as in Section 5. We compared noise-reducing prolongation operators based on integrating the Perona-Malik diffusion equation or with prolongation operators based on integrating a TV-norm-type operator in [22] and found the former to yield somewhat higher accuracy. We therefore use Perona-Malik-based prolongation operators in the present paper. However, the difference in performance was not very large and other noise-reducing schemes may perform as well as the one used in this paper. Similarly, other noise-reducing restriction operators also may yield good results; see Buades et al. [6] for a review of denoising algorithms.

Several multilevel methods for ill-posed problems have been described in the literature; see, e.g., [22, 29] for references. However, only few of them are applied to the solution of (1.4) without Tikhonov regularization. Recently, Donatelli and Serra-Capizzano [12, 13] described such multigrid methods, which are not cascadic; they use full V- or W-cycles. These methods are shown in [13] not to be regularization methods, though the computed results are nice. Scherzer [32] analyzed a cascadic Landweber iteration-based multilevel method for nonlinear ill-posed problems. However, Landweber iteration-based methods tend to converge slowly. Cascadic multilevel methods for image restoration, related to the methods of the present paper are described in [21, 22]. The solution of general linear ill-posed problems by cascadic multilevel methods is discussed in [29], where restriction is carried out by downsampling and prolongation by piecewise linear interpolation. These multilevel are not noise-reducing. They determine approximate solutions of about the same quality, but with less computational effort, than the corresponding one-level Krylov subspace methods.

2. Cascadic multilevel methods.

Let

$$\mathbb{W}_1 \subset \mathbb{W}_2 \subset \cdots \subset \mathbb{W}_\ell$$

be a sequence of nested subspaces of $\mathbb{R}^n$ with $\dim(\mathbb{W}_i) = n_i$ and $n_1 < n_2 < \ldots < n_\ell = n$. We refer to the subspaces $\mathbb{W}_i$ as levels, with $\mathbb{W}_1$ being the coarsest and $\mathbb{W}_\ell = \mathbb{R}^n$ the finest level. Each level is equipped with a weighted least-squares norm; level $\mathbb{W}_i$ has a norm of the form (1.5) with n replaced by n_i.

Let $A_i \in \mathbb{R}^{n_i \times n_i}$ be a representation of the integral operator (1.1) on level $\mathbb{W}_i$. This defines implicitly the restriction operators $R_i : \mathbb{R}^n \to \mathbb{W}_i$ and $Q_i : \mathbb{R}^n \to \mathbb{W}_i$, such that

$$A_i = R_i A Q_i^*. \tag{2.1}$$

Here Q_i^* denotes the adjoint of Q_i. For some discretizations $Q_i^* = R_i^*$. We define $R_\ell = Q_\ell = I$.

The choice of restrictions R_i and Q_i is in our experience less crucial for achieving accurate approximations of the desired solution $\hat{\boldsymbol{x}}$ of the noise-free system (1.2) than the choice of restriction operators $\tilde{R}_i : \mathbb{R}^n \to \mathbb{W}_i$ for reducing the available noise-contaminated right-hand side $\boldsymbol{b}^\delta$ in (1.4). Let

$$\boldsymbol{b}_i^\delta := \tilde{R}_i \boldsymbol{b}^\delta, \qquad 1 \leq i < \ell, \tag{2.2}$$

where the $\tilde{R}_i$ are determined by repeated local weighted averaging. Section 3 describes two such restriction operators and discusses their noise-reducing properties. For notational simplicity, we define $\tilde{R}_\ell = I$.

Assume that a sufficiently accurate approximate solution has been computed on level $i-1$ with $i \leq n$. The nonlinear prolongation operator $P_i : \mathbb{W}_{i-1} \to \mathbb{W}_i$ then determines an approximate solution on level i. It is defined by piecewise linear interpolation followed by integration of the Perona-Malik convection diffusion equation over a short time-interval; see Section 4 for details. The operators P_i, $1 < i \leq \ell$, are designed to be noise-reducing and edge-preserving. The choice of the P_i is important for the performance of the multilevel method. We compare in Section 5 results achieved with the nonlinear prolongation operators P_i and with linear prolongation operators $L_i : \mathbb{W}_{i-1} \to \mathbb{W}_i$, $1 < i \leq \ell$, defined by piecewise linear interpolation.

The multilevel method first computes an approximate solution in $\mathbb{W}_1$ of the smallest linear system of equations $A_1 \boldsymbol{x} = \boldsymbol{b}_1^\delta$, using one of the iterative methods GMRES, LSQR, MR-II, or RRGMRES. We refer to the iterative method chosen as IM in Algorithm 2.1 below. The iterations are terminated as soon as an iterate that satisfies a stopping rule related to the discrepancy principle has been determined; see below for a discussion of this stopping criterion. The prolongation P_2 maps this iterate from $\mathbb{W}_1$ into $\mathbb{W}_2$, and the mapped iterate is corrected in $\mathbb{W}_2$ with the chosen iterative method. Again, the iterations are terminated by a stopping rule related to the discrepancy principle. The approximate solution in $\mathbb{W}_2$ computed in this manner is mapped into $\mathbb{W}_3$ by P_3. The computations are continued in this fashion until an approximation of $\hat{\boldsymbol{x}}$ has been determined in $\mathbb{W}_\ell = \mathbb{R}^n$. The computations are summarized by the following algorithm.

Algorithm 2.1. *Multilevel Algorithm*

Input: A, $\boldsymbol{b}^\delta$, δ, $\ell \geq 1$ *(number of levels);*
Output: approximate solution $\tilde{\boldsymbol{x}} \in \mathbb{W}_\ell$ *of (1.2);*
Determine A_i *and* $\boldsymbol{b}_i^\delta$ *from (2.1) and (2.2), respectively, for* $1 \leq i \leq \ell$*;*
$\boldsymbol{x}_0 := \boldsymbol{0}$;
for $i := 1, 2, \ldots, \ell$ *do*
  $\boldsymbol{x}_{i,0} := P_i \boldsymbol{x}_{i-1}$;
  $\Delta \boldsymbol{x}_{i,m_i} := IM(A_i, \boldsymbol{b}_i^\delta - A_i \boldsymbol{x}_{i,0})$;
  Correction step: $\boldsymbol{x}_i := \boldsymbol{x}_{i,0} + \Delta \boldsymbol{x}_{i,m_i}$;
endfor
$\tilde{\boldsymbol{x}} := \boldsymbol{x}_\ell$; □

In the algorithm $\Delta \boldsymbol{x}_{i,m_i} := IM(A_i, \boldsymbol{b}_i^\delta - A_i \boldsymbol{x}_{i,0})$ denotes the computation of the approximate solution $\Delta \boldsymbol{x}_{i,m_i}$ of

$$A_i \boldsymbol{z}_i = \boldsymbol{b}_i^\delta - A_i \boldsymbol{x}_{i,0} \tag{2.3}$$

by m_i iterations with the chosen iterative method.

Example 2.1. Consider the solution of (2.3) by MR-II when A_i is symmetric or by RRGMRES when A_i is nonsymmetric. Define the residual vector $\boldsymbol{r}_i := \boldsymbol{b}_i^\delta - A_i \boldsymbol{x}_{i,0}$ and use the initial approximate solution $\Delta \boldsymbol{x}_{i,0} = \boldsymbol{0}$. Then the ℓth iterate, $\Delta \boldsymbol{x}_{i,\ell}$, determined by MR-II or RRGMRES solves the minimization problem

$$\|A_i \Delta \boldsymbol{x}_{i,\ell} - \boldsymbol{r}_i\| = \min_{\boldsymbol{z}_i \in \mathbb{K}_\ell(A_i, A_i \boldsymbol{r}_i)} \|A_i \boldsymbol{z}_i - \boldsymbol{r}_i\|, \qquad \boldsymbol{z}_i \in \mathbb{K}_\ell(A_i, A_i \boldsymbol{r}_i).$$

□

Our stopping rule on each level is analogous to (1.7). We assume that there are constants c_i independent of δ, such that

$$\|\boldsymbol{b}_i^\delta - \boldsymbol{b}_i\| \leq c_i \delta, \qquad 1 \leq i \leq \ell, \tag{2.4}$$

where δ is defined by (1.6). The noise-reducing property of the restriction operators $\tilde{R}_i$ suggests suitable choices for the coefficients c_i; see Section 3.

STOPPING RULE 2.2. *Let δ and the c_i be the same as in (2.4) and denote the iterates determined by the IM iterative method applied to the solution of (2.3) with initial iterate $\Delta\boldsymbol{x}_{i,0} = \mathbf{0}$ by $\Delta\boldsymbol{x}_{i,m}$, $m = 1, 2, \ldots$. Terminate the iteration on level i as soon as an iterate $\Delta\boldsymbol{x}_{i,m_i}$ that satisfies*

$$\|\boldsymbol{b}_i - A_i\boldsymbol{x}_{i,0} - A_i\Delta\boldsymbol{x}_{i,m_i}\| \leq c_i\delta \tag{2.5}$$

has been determined, where $m_i = m_{i,\delta}$ denotes the termination index. This index depends on δ.

The multilevel methods so defined can be shown to be regularization methods under various restrictions; see [21, 22, 29]. Linear regularization operators can be used as right preconditioners. These regularization operators change the solution subspace. Suitably chosen operators can improve the quality of the computed approximation of $\hat{\boldsymbol{x}}$; see, e.g., [2, 20, 30] for discussions on the choice and use of linear regularization operators.

3. Noise-reducing restriction operators. The choice of restriction operators depends on the space dimension of the set $\mathbb{S}$ in (1.1). This section discusses restriction operators that are applicable when $\mathbb{S}$ is a bounded interval. Let $x_i^{(j)}$ denote the jth component of the vector $\boldsymbol{x}_i \in \mathbb{W}_i$. We assume the discretization to be monotonic, i.e., $x_i^{(j)}$ corresponds to the solution of the integral equations (1.1) at a point $t_i^{(j)}$, $1 \leq j \leq n_j$, with $t_i^{(j)} < t_i^{(j+1)}$ for all j. We remark that when $\mathbb{S}$ is a square, restriction operators developed for the restoration of two-dimensional images can be applied; see [21, 22] for examples of such restriction operators.

This section introduces two noise-reducing restriction operators, the first of which is inspired by the "staircasing"-reducing method by Buades et al. [5]. Our second restriction operator is simpler, but may smooth edges more. It is well suited for linear discrete ill-posed problems, whose desired solution is the discretization of a smooth function.

Our first restriction operator is based on solving weighted local least-squares problems using a window consisting of three neighboring entries $x_i^{(j)}$. This is similar to the method proposed by Buades et al. [5]. Introduce the weight function

$$\omega_i^{(2j)}(s) := \exp\left(-\gamma\left(x_i^{(2j+s)} - x_i^{(2j)}\right)^2\right), \tag{3.1}$$

with γ a positive constant, and consider the least-squares problem

$$\min_{a_0,a_1} \sum_{s\in\{0,\pm1\}} \left(x_i^{(2j+s)} - (a_0 + a_1 s)\right)^2 \omega_i^{(2j)}(s) \tag{3.2}$$

for the coefficients $\{a_0, a_1\}$ of the linear function. Denote the solution by $\{\hat{a}_0, \hat{a}_1\}$ and let $x_{i-1}^{(j)} := \hat{a}_0$. The least-squares problems (3.2) are solved for all $1 \leq j \leq n_{i-1}$. This determines the vector $\boldsymbol{x}_{i-1} = [x_{i-1}^{(1)}, x_{i-1}^{(2)}, \ldots, x_{i-1}^{(n_{i-1})}]^T$ and defines the mapping $\tilde{M}_i : \mathbb{W}_i \to \mathbb{W}_{i-1}$.

The number of arithmetic floating point operations required to solve (3.2) for all entries of $\boldsymbol{x}_{i-1}$ is proportional to n_{i-1} and therefore modest. The solution of each least-squares problem can be computed by determining a modified QR-decomposition of a 3×2 matrix based on modified Householder transformations; see [15] for details.

Application of the mappings $\tilde{M}_k$ for $k = \ell, \ell+1, \ldots, i+1$, in order, defines the restriction operators

$$\tilde{R}_i : \mathbb{R}^n \to \mathbb{W}_i, \qquad \tilde{R}_i := \tilde{M}_{i+1}\tilde{M}_{i+2}\ldots\tilde{M}_\ell, \qquad 1 \le i < \ell. \tag{3.3}$$

Since least-squares approximation implies smoothing and the noise in $\boldsymbol{b}^\delta$ contains a significant high-frequency component, the restrictions $\boldsymbol{b}_i^\delta$ of $\boldsymbol{b}^\delta$, for $i < \ell$, defined by (2.2) typically contain less noise than $\boldsymbol{b}_\ell^\delta$. This is illustrated by the following result.

PROPOSITION 3.1. *Let $\gamma = 0$ in (3.1) and assume that the entries $e_\ell^{(j)}$ of the noise-vector $\boldsymbol{e}_\ell^\delta := \boldsymbol{b}^\delta - \boldsymbol{b}$ are uncorrelated random variables with zero mean. Let $\|\boldsymbol{e}_\ell^\delta\|_V^2$ denote the average variance of the entries, i.e.,*

$$\|\boldsymbol{e}_\ell^\delta\|_V^2 := \frac{1}{n_\ell}\sum_{j=1}^{n_\ell} Var(e_\ell^{(j)}).$$

Assume that the solution is periodic, so that boundary effects can be ignored. Let

$$n_{\ell-1} = \frac{1}{2}n_\ell \tag{3.4}$$

and

$$\boldsymbol{e}_{\ell-1}^\delta := \tilde{R}_{\ell-1}\boldsymbol{e}_\ell^\delta.$$

Then

$$\|\boldsymbol{e}_{\ell-1}^\delta\|_V \le \frac{2}{3}\|\boldsymbol{e}_\ell^\delta\|_V, \qquad \|\boldsymbol{e}_{\ell-1}^\delta\|_V \ge \frac{\sqrt{2}}{3}\|\boldsymbol{e}_\ell^\delta\|_V.$$

Proof. For $i = \ell$, the least-squares problem (3.2), with $\omega^{(2j)} = 1$ for all j, can be expressed as

$$\min_{\boldsymbol{a}\in\mathbb{R}^2} \|H\boldsymbol{a} - \boldsymbol{y}^{(j)}\|, \tag{3.5}$$

where

$$H = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & -1 \end{bmatrix}^T \in \mathbb{R}^{3\times 2},$$

$$\boldsymbol{a} = [a_0, a_1]^T \in \mathbb{R}^2,$$

$$\boldsymbol{y}^{(j)} = [x_\ell^{(2j-1)}, x_\ell^{(2j)}, x_\ell^{(2j+1)}]^T \in \mathbb{R}^3.$$

Denote the solution of (3.5) by $\hat{\boldsymbol{a}} = [\hat{a}_0, \hat{a}_1]^T$. Since H has orthogonal columns, we obtain

$$x_{\ell-1}^{(j)} := \hat{a}_0 = \frac{1}{3}\sum_{s\in\{0,\pm1\}} x_\ell^{(2j+s)}.$$

The component $e_{\ell-1}^{(j)}$ of the vector $\boldsymbol{e}_{\ell-1}^\delta$ represents the error (noise) in $x_{\ell-1}^{(j)}$ and is the average of the errors $e_\ell^{(2j+s)}$ in the three entries $x_\ell^{(2j+s)}$, $s \in \{0, \pm1\}$. Therefore,

$$\mathrm{Var}(e_{\ell-1}^{(j)}) = \frac{1}{9}\sum_{s\in\{0,\pm1\}} \mathrm{Var}(e_\ell^{(2j+s)}),$$

and it follows that

$$\|\boldsymbol{e}^{\delta}_{\ell-1}\|^2_V = \frac{1}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \mathrm{Var}(e^{(j)}_{\ell-1}) \tag{3.6}$$

$$= \frac{1}{9n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \sum_{s\in\{0,\pm1\}} \mathrm{Var}(e^{(2j+s)}_{\ell}) \tag{3.7}$$

$$\leq \frac{2}{9n_{\ell-1}} \sum_{j=1}^{n_{\ell}} \mathrm{Var}(e^{(j)}_{\ell}) = \frac{4}{9n_{\ell}} \sum_{j=1}^{n_{\ell}} \mathrm{Var}(e^{(j)}_{\ell}) = \frac{4}{9}\|\boldsymbol{e}^{\delta}_{\ell}\|^2_V. \tag{3.8}$$

The inequality (3.8) is a consequence of the fact that some errors $e^{(2j+s)}_{\ell}$ contribute to two errors $e^{(j)}_{\ell-1}$. Moreover, no entry of $\boldsymbol{e}_\ell$ contributes to more than two entries of $\boldsymbol{e}_{\ell-1}$. The equality on the left in (3.8) follows from (3.4).

Every entry of $\boldsymbol{e}_\ell$ contributes to at least one entry of $\boldsymbol{e}_{\ell-1}$. We therefore obtain analogously to (3.6)-(3.8) the inequality

$$\begin{aligned}\|\boldsymbol{e}^{\delta}_{\ell-1}\|^2_V &= \frac{1}{9n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \sum_{s\in\{0,\pm1\}} \mathrm{Var}(e^{(2j+s)}_{\ell}) \\ &\geq \frac{1}{9n_{\ell-1}} \sum_{j=1}^{n_{\ell}} \mathrm{Var}(e^{(j)}_{\ell}) = \frac{2}{9n_{\ell}} \sum_{j=1}^{n_{\ell}} \mathrm{Var}(e^{(j)}_{\ell}) = \frac{2}{9}\|\boldsymbol{e}^{\delta}_{\ell}\|^2_V.\end{aligned}$$

This establishes the proposition. □

COROLLARY 3.2. *Let the conditions of Proposition 3.1 hold and assume further that*

$$Var(e^{(j)}_{\ell}) = \eta^2, \qquad 1 \leq j \leq n_\ell.$$

Then

$$\|\boldsymbol{e}^{\delta}_{\ell-1}\|_V = \frac{1}{\sqrt{3}}\|\boldsymbol{e}^{\delta}_{\ell}\|_V.$$

Proof. The result follows from (3.6)-(3.7). □

Approximating $\|\boldsymbol{e}^{\delta}_{k}\|_V$ by $\|\boldsymbol{e}^{\delta}_{k}\|$ for $k = \ell-1, \ell$, we obtain from Corollary 3.2 that

$$\|\boldsymbol{e}^{\delta}_{\ell-1}\| \approx \frac{1}{\sqrt{3}}\|\boldsymbol{e}^{\delta}_{\ell}\|.$$

This suggests the choice of the coefficients

$$c_i = \left(\frac{1}{\sqrt{3}}\right)^{\ell-i} c, \qquad 1 \leq i \leq \ell, \tag{3.9}$$

in (2.5), where $c > 1$ is a user-specified constant. Generally, c is chosen close to unity when the available estimate of the norm of the noise is accurate.

The following simpler averaging scheme for the restriction operator can be attractive when the desired solution $\hat{\boldsymbol{x}}$ is known to represent a smooth function on the interval $\mathbb{S}$. Define the mappings $\tilde{M}_i : \mathbb{W}_i \to \mathbb{W}_{i-1}$, for $2 \leq i \leq \ell$, by

$$x^{(j)}_{i-1} := \omega_1 x^{(2j-1)}_{i} + \omega_2 x^{(2j)}_{i} + \omega_1 x^{(2j+1)}_{i}, \qquad 1 \leq j \leq n_{i-1}, \tag{3.10}$$

with

$$\omega_1 := \frac{1}{2+\sqrt{2}}, \qquad \omega_2 := \frac{\sqrt{2}}{2+\sqrt{2}}.$$

Let the restriction operators $\tilde{R}_i$ be determined by these $\tilde{M}_i$ according to (3.3).

PROPOSITION 3.3. *Assume that the entries $e_\ell^{(j)}$ of the noise-vector $\boldsymbol{e}_\ell^\delta := \boldsymbol{b}^\delta - \boldsymbol{b}$ are uncorrelated random variables with zero mean. Let (3.4) hold. Assume the solution to be periodic, so that boundary effects can be ignored. Let*

$$\boldsymbol{e}_{\ell-1}^\delta := \tilde{R}_{\ell-1}\boldsymbol{e}_\ell^\delta$$

with $\tilde{R}_\ell$ defined as above. Then

$$\|\boldsymbol{e}_{\ell-1}^\delta\|_V = \frac{1}{1+1/\sqrt{2}}\|\boldsymbol{e}_\ell^\delta\|_V. \tag{3.11}$$

where $\|\cdot\|_V$ is defined in Proposition 3.1.

Proof. It follows from (3.10) that

$$\mathrm{Var}(e_{\ell-1}^{(j)}) = \omega_1^2 \mathrm{Var}(e_\ell^{(2j-1)}) + \omega_2^2 \mathrm{Var}(e_\ell^{(2j)}) + \omega_1^2 \mathrm{Var}(e_\ell^{(2j+1)}).$$

Therefore, using (3.4), we obtain

$$\begin{aligned}
\|\boldsymbol{e}_{\ell-1}^\delta\|_V^2 &= \frac{1}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \mathrm{Var}(e_{\ell-1}^{(j)}) \\
&= \frac{\omega_1^2}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \Big(\mathrm{Var}(e_\ell^{(2j-1)}) + \mathrm{Var}(e_\ell^{(2j+1)})\Big) + \frac{\omega_2^2}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \mathrm{Var}(e_\ell^{(2j)}) \\
&= \frac{2\omega_1^2}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \mathrm{Var}(e_\ell^{(2j-1)}) + \frac{\omega_2^2}{n_{\ell-1}} \sum_{j=1}^{n_{\ell-1}} \mathrm{Var}(e_\ell^{(2j)}) \\
&= \frac{2\omega_2^2}{n_\ell} \sum_{j=1}^{n_\ell} \mathrm{Var}(e_\ell^{(j)}) = \frac{4}{(2+\sqrt{2})^2}\|\boldsymbol{e}_\ell^\delta\|_V^2,
\end{aligned}$$

which shows (3.11). □

Property (3.11) suggests that the coefficients

$$c_i = \left(\frac{1}{1+1/\sqrt{2}}\right)^{\ell-i} c, \qquad 1 \le i \le \ell, \tag{3.12}$$

be used in (2.5) when the mappings $\tilde{M}_i$ are determined by (3.10).

4. Edge-preserving nonlinear prolongation operators. We describe the nonlinear edge-preserving prolongation operators P_i used in the computed examples. They have previously been applied in [22] in the context of image restoration. Here we discuss their application to the solution of linear discrete ill-posed problems (1.4), with a matrix obtained by discretizing the integral operator (1.1) with $\mathbb{S}$ a bounded real interval.

The prolongation operators are made up of two parts: first the approximate solution $\boldsymbol{x}_{i-1} \in \mathbb{W}_{i-1}$ determined by the correction step of Algorithm 2.1 is mapped into

$\mathbb{W}_i$ by piecewise linear interpolation. This defines the linear operator L_i. Subsequently $L_i\boldsymbol{x}_{i-1}$ is smoothed by solving an initial-boundary value problem for a discretized Perona-Malik nonlinear diffusion equation. Consider for the moment $x_i := L_i\boldsymbol{x}_{i-1}$ a continuous function. Then, roughly, we integrate

$$\frac{\partial x}{\partial \tau} = \frac{\partial}{\partial s}\left(\psi'\left(\left|\frac{\partial}{\partial s}x\right|^2\right)\frac{\partial}{\partial s}x\right), \qquad x = x(\tau, s), \qquad s \in \mathbb{S}, \qquad \tau \geq 0, \tag{4.1}$$

over a short τ-interval in order to remove noise while preserving rapid spacial transitions, such as edges. We use the initial function $x(0,s) = x_i(s)$, $s \in \mathbb{S}$, and Neumann boundary condition zero. The function ψ' is the Perona-Malik diffusivity,

$$\psi'(s) = \frac{\rho}{s+\rho},$$

where ρ is a positive constant; see [25]. The differential equation (4.1) is designed to determine an element in $\mathbb{W}_i$ that has edges close to those of $\boldsymbol{x}_{i-1}$ and is smooth elsewhere.

Space-discretization of (4.1) by finite differences and time-integration by Euler's method yields

$$\begin{aligned} \frac{x^{(k+1,i)} - x^{(k,i)}}{\Delta\tau} &= \sum_{j\in\mathbb{N}(i)} \frac{p^{(k,j)} + p^{(k,i)}}{2(\Delta s)^2}(x^{(k,j)} - x^{(k,i)}), \\ x^{(k,i)} &\approx x(k\Delta\tau, s_i), \qquad s_i = s_0 + i\Delta s, \end{aligned} \tag{4.2}$$

where the set $\mathbb{N}(i)$ contains the indices of neighboring mesh points; if $x^{(k,i)}$ is an interior mesh-point, then $\mathbb{N}(i) = \{i-1, i+1\}$, while if $x^{(k,i)}$ is a boundary point, then $\mathbb{N}(i)$ only contains one index. The parameter Δs is the mesh size and

$$p^{(k,i)} := \psi'\left(\left(\frac{x^{(k,i+1)} - x^{(k,i-1)}}{2\Delta s}\right)^2\right),$$

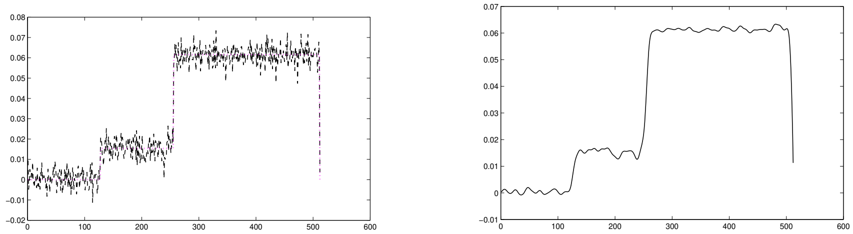
is a second order approximation of $\psi'\left(\left|\frac{\partial}{\partial s}x\right|^2\right)$ at $x^{(k,i)}$ by finite difference discretization. We carry out about 10 integration steps $\Delta\tau$. For reason of stability, we let $0 < \Delta\tau \leq 1/3$. The small number of steps avoids difficulties due to numerical instability and keeps the computational effort required for integration negligible. We note that nonlinear diffusion filtering models, similar to (4.1), are applied successfully in PDE-based methods for image processing problems; see, e.g., [11, 33].

5. Numerical examples. This section illustrates the application of Algorithm 2.1 to the solution of linear discrete ill-posed problems. We compare the performance of Algorithm 2.1 for piecewise linear and nonlinear prolongation operators. Piecewise linear prolongation previously has been applied in [29]. We also illustrate the denoising process used in the nonlinear prolongation operators P_i described in Section 4.

Let the noise-level be defined by

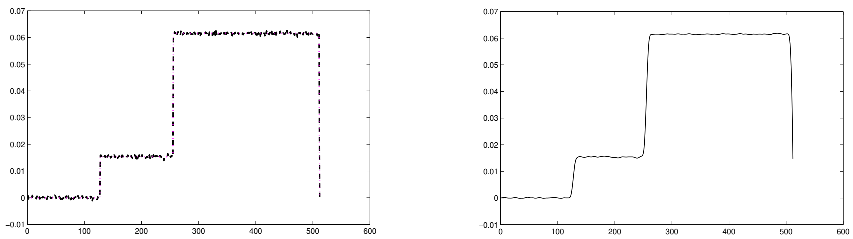
$$\nu = \frac{\delta}{\|\boldsymbol{b}\|},$$

where the error δ is given by (1.6). We assume that an accurate estimate of ν is available in all examples of this section, except in Example 5.5, and therefore choose the parameter c in (3.9) and (3.12) fairly close to unity.



(a) (b)

FIG. 5.1. *Example 5.1: Denoising of a contaminated signal by integration the discretized Perona-Malik equation (4.2) for* 20 *time steps. The unperturbed signal is a step function. (a) Contaminated signal, with additive Gaussian noise of level* $1 \cdot 10^{-1}$ *(dashed curve), (b) denoised signal (continuous curve).*



(a) (b)

FIG. 5.2. *Example 5.1: Denoising of a contaminated signal by integration the discretized Perona-Malik equation (4.2) for* 10 *time steps. The unperturbed signal is a step function. (a) Contaminated signal, with additive Gaussian noise of level* $1 \cdot 10^{-2}$ *(dashed curve), (b) denoised signal (continuous curve).*

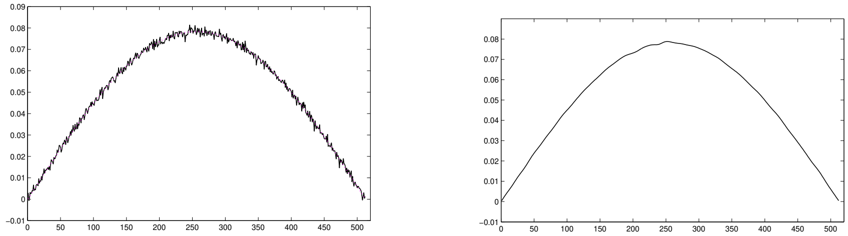
Example 5.1. We illustrate the denoising process described in Section 4. The process is applied to three noise-contaminated "signals" $\boldsymbol{b}^\delta$. Figure 5.1(a) shows a contaminated step function represented by a vector with 512 entries. The noise level is 10%. Figure 5.1(b) displays the denoised signal obtained by integrating the discretized Perona-Malik equation (4.2). Figure 5.2 differs from Figure 5.1 in that the noise-level is reduced to 1%. We remark that signals with high noise-levels may benefit from a larger number of time steps, than signals corrupted by little noise.

Finally, Figure 5.3 shows a noisy and denoised smooth signal. The noise level is 1%. Denoising is achieved by application of 10 integration steps. The step size for all computations for this example is $\Delta\tau = 0.3$. □

Example 5.2. This example compares the performance of multilevel methods using nonlinear or piecewise linear prolongations, with the corresponding one-level iterative methods. The methods are applied to the solution of the Fredholm integral equation of the first kind,

$$\int_0^{\pi/2} h(s,t)x(t)dt = b(s), \qquad 0 \le s \le \pi, \tag{5.1}$$

with $h(s,t) := \exp(s\cos(t))$ and $b(s) := 2\sinh(s)/s$, discussed by Baart [1]. It has the solution $x(t) = \sin(t)$. We use the MATLAB code `baart` from [19] to discretize (5.1) by a Galerkin method with 512 orthonormal box functions as test and trial



(a) (b)

FIG. 5.3. *Example 5.1: Denoising of a contaminated signal by integration the discretized Perona-Malik equation (4.2) for* 10 *time steps. The unperturbed signal is a smooth function. (a) Contaminated signal, with additive Gaussian noise of level* $1 \cdot 10^{-2}$ *(dashed curve), (b) denoised signal (continuous curve).*

		P_i		L_i	
ℓ	ν	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter
1	$1 \cdot 10^{-2}$	$3.51 \cdot 10^{-2}$	3		
2	$1 \cdot 10^{-2}$	$3.28 \cdot 10^{-2}$	3 1	$3.41 \cdot 10^{-2}$	3 1
3	$1 \cdot 10^{-2}$	$3.17 \cdot 10^{-2}$	3 1 1	$3.42 \cdot 10^{-2}$	3 1 1
5	$1 \cdot 10^{-2}$	$2.97 \cdot 10^{-2}$	3 2 1 1 1	$3.39 \cdot 10^{-2}$	3 2 1 1 1
1	$1 \cdot 10^{-3}$	$3.53 \cdot 10^{-2}$	3		
2	$1 \cdot 10^{-3}$	$3.36 \cdot 10^{-2}$	3 1	$3.55 \cdot 10^{-2}$	3 1
3	$1 \cdot 10^{-3}$	$3.10 \cdot 10^{-2}$	3 2 1	$3.54 \cdot 10^{-2}$	3 2 1
5	$1 \cdot 10^{-3}$	$1.94 \cdot 10^{-2}$	3 3 2 2 1	$3.46 \cdot 10^{-2}$	3 2 2 2 1

TABLE 5.1

Example 5.2: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ *of the problem* `baart` *and the computational effort required by Algorithm 2.1 using the RRGMRES iterative solver with nonlinear* (P_i) *or piecewise linear* (L_i) *prolongation operators. The restriction operators determine weighted averages (3.10).*

		P_i		L_i	
ℓ	ν	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter
1	$1 \cdot 10^{-2}$	$1.67 \cdot 10^{-1}$	3		
2	$1 \cdot 10^{-2}$	$1.62 \cdot 10^{-1}$	3 1	$1.67 \cdot 10^{-1}$	3 1
3	$1 \cdot 10^{-2}$	$1.57 \cdot 10^{-1}$	3 1 1	$1.67 \cdot 10^{-1}$	3 1 1
5	$1 \cdot 10^{-2}$	$1.30 \cdot 10^{-1}$	3 2 1 1 1	$1.77 \cdot 10^{-1}$	3 2 1 1 1
1	$1 \cdot 10^{-3}$	$1.66 \cdot 10^{-1}$	3		
2	$1 \cdot 10^{-3}$	$1.62 \cdot 10^{-1}$	3 1	$1.67 \cdot 10^{-1}$	3 1
3	$1 \cdot 10^{-3}$	$1.57 \cdot 10^{-1}$	3 2 1	$1.67 \cdot 10^{-1}$	3 2 1
5	$1 \cdot 10^{-3}$	$7.97 \cdot 10^{-2}$	4 3 2 2 1	$1.20 \cdot 10^{-1}$	4 2 2 2 1

TABLE 5.2

Example 5.2: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ *of the problem* `baart` *and the computational effort required by Algorithm 2.1 using the LSQR iterative solver with nonlinear* (P_i) *or piecewise linear* (L_i) *prolongation operators. The restriction operators determine weighted averages (3.10).*

functions. This yields the nonsymmetric matrix $A \in \mathbb{R}^{512\times 512}$. The code `baart` also provides a scaled discrete approximation $\hat{\boldsymbol{x}} \in \mathbb{R}^{512}$ of $x(t)$. The noise-free right-hand side of (1.2) is determined by $\boldsymbol{b} := A\hat{\boldsymbol{x}}$. Adding normally distributed "noise" with

		P_i		L_i	
ℓ	ν	$\Vert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\Vert/\Vert\hat{\boldsymbol{x}}\Vert$	# iter	$\Vert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\Vert/\Vert\hat{\boldsymbol{x}}\Vert$	# iter
1	$1\cdot10^{-2}$	$3.51\cdot10^{-2}$	3		
2	$1\cdot10^{-2}$	$3.26\cdot10^{-2}$	3 1	$3.41\cdot10^{-2}$	3 1
3	$1\cdot10^{-2}$	$3.10\cdot10^{-2}$	3 1 1	$3.41\cdot10^{-2}$	3 1 1
5	$1\cdot10^{-2}$	$2.51\cdot10^{-2}$	3 2 1 1 1	$3.35\cdot10^{-2}$	3 2 1 1 1
1	$1\cdot10^{-3}$	$3.53\cdot10^{-2}$	3		
2	$1\cdot10^{-3}$	$3.34\cdot10^{-2}$	3 1	$3.57\cdot10^{-2}$	3 1
3	$1\cdot10^{-3}$	$3.08\cdot10^{-2}$	3 2 1	$3.56\cdot10^{-2}$	3 2 1
5	$1\cdot10^{-3}$	$2.00\cdot10^{-2}$	3 2 2 3 1	$3.50\cdot10^{-2}$	3 2 2 2 1

TABLE 5.3

Example 5.2: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ of the problem `baart` *and the computational effort required by Algorithm 2.1 using the RRGMRES iterative solver with and nonlinear (P_i) or piecewise linear (L_i) prolongation operators. The restriction operators solve weighted local least-squares problems (3.2).*

zero mean, scaled to correspond to the noise-levels $\nu = 1\cdot10^{-2}$ or $\nu = 1\cdot10^{-3}$, yields the right-hand side $\boldsymbol{b}^\delta$ of (1.4); cf. (1.3). We also use the code `baart` to generate the matrices A_i of size $16\cdot2^i \times 16\cdot2^i$ for $1 \le i < 5$. These matrices, as well as $A_5 := A$, are numerically singular.

Computed results are reported in Tables 5.1-5.3. Let $\tilde{\boldsymbol{x}}$ denote the approximate solution determined by a multilevel or one-level method. The tables display the relative errors $\Vert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\Vert/\Vert\hat{\boldsymbol{x}}\Vert$ and the number of iterations for the noise levels $1\cdot10^{-2}$ and $1\cdot10^{-3}$. The columns labeled "# iter" display the number of iterations on each level. The ℓ-tuples show, from left to right, the number of iterations for increasing level index. Thus, the leftmost entry of each ℓ-tuple shows the number of iterations on the coarsest level, and the rightmost entry the number of iterations on the finest level. The column labeled "ℓ" displays the number of levels used. Thus, the rows labeled $\ell = 1$ show results for the one-level method.

Tables 5.1 and 5.2 show results obtained with Algorithm 2.1, based on the RRGMRES and LSQR iterative solvers, respectively. The restriction operators uses weighted averages (3.10). The RRGMRES-based multilevel method is seen to perform particularly well, with the relative error decreasing when the number of levels is increased. Moreover, the nonlinear prolongation operators P_i yield more accurate approximations of the desired solution than the linear prolongation operators L_i. All multilevel methods, with at least two levels, require only one iteration on the finest level. The one-level method requires three iterations. The multilevel method therefore can be faster for large problems, for which the evaluation of matrix-vector products with the matrix $A = A_\ell$ constitutes the dominant computational effort.

A comparison of restriction operators based on local weighted least-squares approximation (3.2) and weighted averaging (3.10) is provided by the Tables 5.3 and 5.1, respectively. The tables show weighted averaging to yield slightly higher accuracy. This depends on that the desired solution $\hat{\boldsymbol{x}}$ represents a smooth function. $\Box$

Example 5.3. Consider the Fredholm integral equation of the first kind

$$\int_{-6}^{6} h(s,t)x(t)ds = b(s), \qquad -6 \le s \le 6, \tag{5.2}$$

		P_i		L_i	
ℓ	ν	$\lVert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\rVert/\lVert\hat{\boldsymbol{x}}\rVert$	# iter	$\lVert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\rVert/\lVert\hat{\boldsymbol{x}}\rVert$	# iter
1	$1\cdot 10^{-2}$	$2.35\cdot 10^{-2}$	4		
2	$1\cdot 10^{-2}$	$2.31\cdot 10^{-2}$	5 1	$2.32\cdot 10^{-2}$	5 1
3	$1\cdot 10^{-2}$	$2.26\cdot 10^{-2}$	6 1 1	$2.28\cdot 10^{-2}$	6 1 1
5	$1\cdot 10^{-2}$	$2.01\cdot 10^{-2}$	9 3 1 1 1	$2.23\cdot 10^{-2}$	9 2 1 1 1
1	$1\cdot 10^{-3}$	$9.56\cdot 10^{-3}$	7		
2	$1\cdot 10^{-3}$	$9.08\cdot 10^{-3}$	8 5	$9.11\cdot 10^{-3}$	8 5
3	$1\cdot 10^{-3}$	$6.77\cdot 10^{-3}$	10 1 1	$6.84\cdot 10^{-3}$	10 1 1
5	$1\cdot 10^{-3}$	$6.53\cdot 10^{-3}$	11 8 3 1 2	$6.73\cdot 10^{-3}$	11 8 3 1 2

TABLE 5.4

Example 5.3: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ of the problem `phillips` and the computational effort required by Algorithm 2.1 using the MR-II iterative solver with nonlinear (P_i) or piecewise linear (L_i) prolongation operators. The restriction operators determine weighted averages (3.10).

		P_i		L_i	
ℓ	ν	$\lVert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\rVert/\lVert\hat{\boldsymbol{x}}\rVert$	# iter	$\lVert\tilde{\boldsymbol{x}}-\hat{\boldsymbol{x}}\rVert/\lVert\hat{\boldsymbol{x}}\rVert$	# iter
1	$1\cdot 10^{-2}$	$2.35\cdot 10^{-2}$	4		
2	$1\cdot 10^{-2}$	$2.30\cdot 10^{-2}$	5 1	$2.31\cdot 10^{-2}$	5 1
3	$1\cdot 10^{-2}$	$2.10\cdot 10^{-2}$	7 1 1	$2.12\cdot 10^{-2}$	7 1 1
5	$1\cdot 10^{-2}$	$4.11\cdot 10^{-2}$	10 7 1 1 1	$4.71\cdot 10^{-2}$	10 7 1 1 1
1	$1\cdot 10^{-3}$	$9.56\cdot 10^{-3}$	7		
2	$1\cdot 10^{-3}$	$9.35\cdot 10^{-3}$	8 4	$9.39\cdot 10^{-3}$	8 4
3	$1\cdot 10^{-3}$	$6.72\cdot 10^{-3}$	10 1 2	$6.81\cdot 10^{-3}$	10 1 1
5	$1\cdot 10^{-3}$	$1.19\cdot 10^{-2}$	12 6 3 2 1	$1.42\cdot 10^{-2}$	12 3 2 1 1

TABLE 5.5

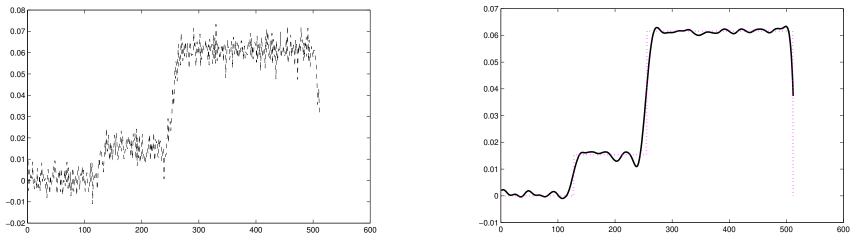
Example 5.3: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ of the problem `phillips` and the computational effort required by Algorithm 2.1 using the MR-II iterative solver with nonlinear (P_i) or piecewise linear (L_i) prolongation operators. The restriction operators solve weighted local least-squares problems (3.2).

discussed by Phillips [26]. Its solution, kernel, and right-hand side are given by

$$\begin{aligned} x(t) &:= \begin{cases} 1+\cos(\frac{\pi}{3}t), & \text{if } |t|<3, \\ 0, & \text{otherwise}, \end{cases} \\ h(s,t) &:= x(s-t), \\ b(s) &:= (6-|s|)(1+\frac{1}{2}\cos(\frac{\pi}{3}s)) + \frac{9}{2\pi}\sin(\frac{\pi}{3}|s|), \end{aligned}$$

respectively. We discretize the integral equation (5.2) using the MATLAB code `phillips` from [19]. Discretization by a Galerkin method with 512 orthonormal box functions as test and trial functions yields the symmetric indefinite matrix $A \in \mathbb{R}^{512\times 512}$, which is nearly singular; it has condition number $\kappa(A) := \|A\|\|A^{-1}\| = 1.81\cdot 10^{9}$. The matrices A_i of order $16\cdot 2^i$, $1\le i\le 5$, are all generated by the code `phillips`. The right-hand side vectors $\boldsymbol{b}$ and $\boldsymbol{b}^{\delta}$ of (1.2) and (1.4), respectively, as well as the desired solution $\hat{\boldsymbol{x}}$ of (1.2) are determined in the same fashion as in Example 5.2.

Tables 5.4 and 5.5 report the performance of Algorithm 2.1 based on the MR-II iterative solver with nonlinear or linear prolongation operators, P_i and L_i, for different



(a) (b)

Fig. 5.4. *Example 5.4: Restoration of piecewise constant signal that has been contaminated by blur and noise. (a) Blurred and noisy signal. (b) restored signal, determined by Algorithm 2.1 with three levels, using the MR-II iterative solver and nonlinear nonlinear prolongation operators P_i (continuous curve). The dotted curve displays the desired piecewise constant signal $\hat{\boldsymbol{x}}$.*

restriction operators.

The row for $\ell = 1$ displays results for the one-level MR-II method. We consider the noise-levels $1 \cdot 10^{-2}$ and $1 \cdot 10^{-3}$. A comparison of the tables shows restriction operators based on weighted averaging (3.10) to generally yield slightly smaller errors than restriction operators based on weighted local least-squares approximation (3.2). Nonlinear prolongation operators P_i are seen to determine more accurate approximations of $\hat{\boldsymbol{x}}$ than piecewise linear prolongation operators L_i. □

	P_i		L_i	
ℓ	$\\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\\|/\\|\hat{\boldsymbol{x}}\\|$	# iter	$\\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\\|/\\|\hat{\boldsymbol{x}}\\|$	# iter
1	$1.02 \cdot 10^{-1}$	1		
2	$8.70 \cdot 10^{-2}$	1 1	$8.78 \cdot 10^{-2}$	1 1
3	$8.08 \cdot 10^{-2}$	4 1 1	$8.88 \cdot 10^{-2}$	4 1 1

Table 5.6

Example 5.4: Errors in computed approximate solutions of the problem `signal` *and the computational effort required by Algorithm 2.1 using the MR-II iterative method with nonlinear (P_i) or piecewise linear (L_i) prolongation operators. The restriction operators solve weighted local least-squares problems (3.2).*

Example 5.4. We consider the determination of a discrete piecewise constant signal $\hat{\boldsymbol{x}} \in \mathbb{R}^{512}$ from a blurred and noisy version $\boldsymbol{b}^\delta \in \mathbb{R}^{512}$ defined on a uniform grid. We refer to this restoration problem as `signal`. The blurring is modeled by a convolution with a Gaussian with mean zero and variance 0.6. The blurring operator is a symmetric Toeplitz matrix $A \in \mathbb{R}^{512\times512}$. The blurred and noisy signal $\boldsymbol{b}^\delta$ is generated similarly as in Example 5.2. The noise-level is $1 \cdot 10^{-1}$.

The Toeplitz matrices A_i of order $64\cdot2^i$, $i = 1, 2$, are defined similarly as $A_3 := A$. The condition numbers for the matrices A_i grow rapidly with i; we have $\kappa(A_1) = 3.53 \cdot 10^6$, $\kappa(A_2) = 9.20 \cdot 10^{11}$, and $\kappa(A_3) = 1.19 \cdot 10^{17}$. Thus, the latter matrix is numerically singular.

Algorithm 2.1 is applied with the MR-II iterative method. Contaminated and restored signals, as well as $\hat{\boldsymbol{x}}$, are shown in Figure 5.4. Relative errors and computational effort are reported in Table 5.6. The table illustrates the better performance of the nonlinear prolongation operators. □

Example 5.5. We illustrated in Example 5.1 that integration of the Perona-Malik differential equation for a short time interval, with the contaminated data $\boldsymbol{b}^\delta$ as initial

ℓ	ν	δ	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter
1	$1 \cdot 10^{-2}$	$2.89 \cdot 10^{-2}$	$3.51 \cdot 10^{-2}$	3
3	$1 \cdot 10^{-2}$	$2.89 \cdot 10^{-2}$	$3.10 \cdot 10^{-2}$	3 1 1
1	$5 \cdot 10^{-3}$	$1.44 \cdot 10^{-2}$	$3.39 \cdot 10^{-2}$	3
3	$5 \cdot 10^{-3}$	$1.44 \cdot 10^{-2}$	$3.14 \cdot 10^{-2}$	3 1 1
1	$1 \cdot 10^{-3}$	$2.89 \cdot 10^{-3}$	$3.53 \cdot 10^{-2}$	3
3	$1 \cdot 10^{-3}$	$2.89 \cdot 10^{-3}$	$3.08 \cdot 10^{-2}$	3 2 1

TABLE 5.7

Example 5.5: Errors in computed approximate solutions $\tilde{\boldsymbol{x}}$ of the problem `baart` and the computational effort required by Algorithm 2.1 using the RRGMRES iterative solver with nonlinear prolongation operators and restriction operators that compute weighted local least-squares (3.2).

ℓ	ν	$\widetilde{\delta}$	$\|\tilde{\boldsymbol{x}} - \hat{\boldsymbol{x}}\|/\|\hat{\boldsymbol{x}}\|$	# iter
1	$1 \cdot 10^{-2}$	$2.55 \cdot 10^{-2}$	$3.49 \cdot 10^{-2}$	3
3	$1 \cdot 10^{-2}$	$2.55 \cdot 10^{-2}$	$3.35 \cdot 10^{-2}$	3 1 1
1	$5 \cdot 10^{-3}$	$1.28 \cdot 10^{-2}$	$3.40 \cdot 10^{-2}$	3
3	$5 \cdot 10^{-3}$	$1.28 \cdot 10^{-2}$	$3.11 \cdot 10^{-2}$	3 1 1
1	$1 \cdot 10^{-3}$	$2.87 \cdot 10^{-3}$	$3.59 \cdot 10^{-2}$	3
3	$1 \cdot 10^{-3}$	$2.87 \cdot 10^{-3}$	$3.25 \cdot 10^{-2}$	3 2 1

TABLE 5.8

Example 5.5: Estimates of the discrepancy, $\widetilde{\delta}$, determined by integration of the discretized Perona-Malik equation, and computed results obtained when using this estimate in Algorithm 2.1 instead of the exact discrepancy. The iterative method, prolongation and restriction operators are the same as for Table 5.7.

values, reduces the noise in $\boldsymbol{b}^\delta$, without changing important features significantly. The difference between the noisy and integrated vectors gives an estimate for the norm of the noise, δ, in $\boldsymbol{b}^\delta$; cf. (1.6). We denote this estimate by $\widetilde{\delta}$. This example discusses the use of $\widetilde{\delta}$ instead of δ in Stopping Rule 2.2 of Algorithm 2.1.

Consider the problem `baart` of Example 5.2 and determine the matrix $A \in \mathbb{R}^{512\times 512}$ and the vectors $\hat{\boldsymbol{x}}, \boldsymbol{b}, \boldsymbol{b}^\delta \in \mathbb{R}^{512}$ similarly as in Example 5.2. The use of the exact discrepancy (1.6) in Algorithm 2.1 yields Table 5.7, which reports results for an RRGMRES-based three-level method with nonlinear prolongation operators P_i and restriction operators defined by local weighted least-squares (3.2), as well as for the RRGMRES method applied on the finest level only.

In order to determine an estimate of the discrepancy, we integrate the discretized Perona-Malik nonlinear diffusion equation (4.2) for 10 time steps $\Delta\tau = 0.2$ with initial value $\boldsymbol{b}^\delta$. This gives the (partially) denoised vector $\widetilde{\boldsymbol{b}}$ and the computed estimate $\widetilde{\delta} := \|\boldsymbol{b}^\delta - \widetilde{\boldsymbol{b}}\|$ of the discrepancy.

We apply Algorithm 2.1 with the same iterative methods, restriction and prolongation operators as for Table 5.7, but with $\widetilde{\delta}$ replacing δ in Stopping Rule 2.2. Table 5.8 shows results for three-level and one-level methods for several noise-levels. Comparison of Tables 5.7 and 5.8 shows the solution methods to require essentially the same number of iterations and determine approximations $\tilde{\boldsymbol{x}}$ of $\hat{\boldsymbol{x}}$ of about the same accuracy when δ is replaced by its estimate $\widetilde{\delta}$. The tables, as well as numerous other computed examples, suggest that for linear discrete ill-posed problems (1.4), for which no estimate for the noise-level is available, quite accurate estimates can be computed by integrating the discretized Perona-Malik differential equations for a few time steps. The estimated discrepancy, $\widetilde{\delta}$, can be used in Algorithm 2.1, as well as in one-level iterative methods, instead of the exact discrepancy (1.6). □

6. Conclusion and extensions. This paper discusses an adaption of the multilevel methods developed in [21] for image restoration problems to the solution of general linear discrete ill-posed problems. The computed examples show the proposed multilevel to yield higher accuracy than corresponding one-level iterative methods. Moreover, the multilevel methods require fewer matrix-vector product evaluations on the finest level than the corresponding one-level methods. These matrix-vector product evaluations constitute the dominant computational work for large-scale problems.

For the problems considered, RRGMRES-based multilevel methods give more accurate approximations of $\hat{\boldsymbol{x}}$ than LSQR-based multilevel methods. This is different from our experience for image restoration problems reported in [21]. The reason for this is presently being investigated.

Numerical examples show that integration of the "data" $\boldsymbol{b}^{\delta}$ with the nonlinear Perona-Malik convection diffusion equations gives accurate estimates for the noise-level in the data. Other approaches, of course, also can be used to estimate the noise-level, including extrapolation methods [3, 4, 27], as well as the L-curve and variants thereof [10, 17, 28].

REFERENCES

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

[2] C. Brezinski, M. Redivo-Zaglia, G. Rodriguez, and S. Seatzu, Multi-parameter regularization techniques for ill-conditioned linear systems, Numer. Math., 94 (2003), pp. 203–228.

[3] C. Brezinski, G. Rodriguez, and S. Seatzu, Error estimates for linear systems with applications to regularization, Numer. Algorithms, 49 (2008), pp. 85–104.

[4] C. Brezinski, G. Rodriguez, and S. Seatzu, Error estimates for the regularization of least squares problems, Numer. Algorithms, in press.

[5] A. Buades, B. Coll, and J. M. Morel, The staircasing effect in neighborhood filters and its solution, IEEE Trans. Image Processing, 15 (2006), pp. 1499–1505.

[6] A. Buades, B. Coll, and J. M. Morel, A review of image denoising algorithms, with a new one, Multiscale Model. Simul., 4 (2005), pp. 490–530.

[7] D. Calvetti, B. Lewis, and L. Reichel, On the regularizing properties of the GMRES method, Numer. Math., 91 (2002), pp. 605–625.

[8] D. Calvetti, B. Lewis, and L. Reichel, On the choice of subspace for iterative methods for linear discrete ill-posed problems, Int. J. Appl. Math. Comput. Sci., 11 (2001), pp. 1069–1092.

[9] D. Calvetti, L. Reichel, and Q. Zhang, Conjugate gradient algorithms for symmetric inconsistent linear systems, in Proceedings of the Cornelius Lanczos International Centenary Conference, eds. J. D. Brown, M. T. Chu, D. C. Ellison, and R. J. Plemmons, SIAM, Philadelphia, 1994, pp. 267–272.

[10] J. L. Castellanos, S. Gómez, and V. Guerra, The triangle method for finding the corner of the L-curve, Appl. Numer. Math., 43 (2002), pp. 359–373.

[11] S. Corsaro, K. Mikula, A. Sarti, and F. Sgallari, Semi-implicit covolume method in 3D image segmentation, SIAM J. Sci. Comput., 28 (2006), pp. 2248–2265.

[12] M. Donatelli and S. Serra-Capizzano, On the regularization power of multigrid-type algorithms, SIAM J. Sci. Comput., 27 (2006), pp. 2053–2076.

[13] M. Donatelli and S. Serra-Capizzano, Filter factor analysis of an iterative multilevel regularizing method, Electron. Trans. Numer. Anal., 29 (2008), pp. 163–177.

[14] H. W. Engl, M. Hanke, and A. Neubauer, Regularization of Inverse Problems, Kluwer, Dordrecht, 1996.

[15] M. Gulliksson and P.-Å. Wedin, Modifying the QR decomposition to constrained and weighted least-squares, SIAM J. Matrix Anal., 13 (1992), pp. 1298–1313.

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

[17] P. C. Hansen, T. K. Jensen, and G. Rodriguez, An adaptive pruning algorithm for the discrete L-curve criterion, J. Comput. Appl. Math., 198 (2007), pp. 483–492.

[18] P. C. Hansen, Rank-Deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion, SIAM, Philadelphia, 1997.
[19] P. C. Hansen, Regularization tools, version 4.0 for MATLAB 7.3, Numer. Algorithms, 46 (2007), pp. 189–294.
[20] S. Morigi, L. Reichel, and F. Sgallari, Orthogonal projection regularization operators, Numer. Algorithms, 44 (2007), pp. 99–114.
[21] S. Morigi, L. Reichel, and F. Sgallari, Cascadic multilevel methods for fast nonsymmetric blur- and noise-removal, submitted for publication.
[22] S. Morigi, L. Reichel, F. Sgallari, and A. Shyshkov, Cascadic multiresolution methods for image deblurring, SIAM J. Imaging Sci., 1 (2008), pp. 51–74.
[23] A. S. Nemirovskii, The regularization properties of the adjoint gradient method in ill-posed problems, USSR Comput. Math. Math. Phys, 26 (2) (1986), pp. 7–16.
[24] C. C. Paige and M. A. Saunders, LSQR: An algorithm for sparse linear equations and sparse least squares, ACM Trans. Math. Software, 8 (1982), pp. 43–71.
[25] P. Perona and J. Malik, Scale-space and edge detection using anisotropic diffusion, IEEE Trans. Pattern Anal. Mach. Intell., 12 (1990), pp. 629–639.
[26] D. L. Phillips, A technique for the numerical solution of certain integral equations of the first kind, J. ACM, 9 (1962), pp. 84–97.
[27] L. Reichel, G. Rodriguez, and S. Seatzu, Error estimates for large-scale ill-posed problems, Numer. Algorithms, in press.
[28] L. Reichel and H. Sadok, A new L-curve for ill-posed problems, J. Comput. Appl. Math., 219 (2008), pp. 493–508.
[29] L. Reichel and A. Shyshkov, Cascadic multilevel methods for ill-posed problems, J. Comput. Appl. Math., in press.
[30] L. Reichel and Q. Ye, Simple square smoothing regularization operators, Electron. Trans. Numer. Anal., in press.
[31] Y. Saad, Iterative methods for Sparse Linear Systems, 2nd ed., SIAM, Philadelphia, 2003.
[32] O. Scherzer, An iterative multi level algorithm for solving nonlinear ill-posed problems, Numer. Math., 80 (1998), pp. 579–600.
[33] J. Weickert, B. M. H. Romeny, and M. A. Viergever, Efficient and reliable schemes for nonlinear diffusion filtering, IEEE Trans. Image Process., 7 (1998), pp. 398-410.