Cascadic Multilevel Methods for Ill-Posed Problems

Lothar Reichel [a,1,*], Andriy Shyshkov [a,1],

[a] *Department of Mathematical Sciences, Kent State University, Kent, OH 44242, USA.*

Dedicated to Bill Gragg on the occasion of his 70th birthday.

Abstract

Multilevel methods are popular for the solution of well-posed problems, such as certain boundary value problems for partial differential equations and Fredholm integral equations of the second kind. However, little is known about the behavior of multilevel methods when applied to the solution of linear ill-posed problems, such as Fredholm integral equations of the first kind, with a right-hand side that is contaminated by error. This paper shows that cascadic multilevel methods with a conjugate gradient-type method as basic iterative scheme are regularization methods. The iterations are terminated by a stopping rule based on the discrepancy principle.

Key words: regularization, ill-posed problem, multilevel method, conjugate gradient method

1 Introduction

Bill Gragg's many contributions to scientific computing include work on ill-posed problems [9], iterative solution of symmetric, possibly indefinite linear systems of equations [10], and Toeplitz matrices [1,2]. This paper is concerned with all these topics.

* Corresponding author.
Email addresses: `reichel@math.kent.edu` (Lothar Reichel), `ashyshko@math.kent.edu` (Andriy Shyshkov).

[1] Research supported in part by an OBR Research Challenge Grant.

Many problems in science and engineering can be formulated as a Fredholm integral equation of the first kind

$$\int_{\Omega} \kappa(t,s)x(s)ds = b(t), \qquad t \in \Omega. \tag{1}$$

Here Ω denotes a compact Jordan measurable subset of $\mathbb{R} \times \ldots \times \mathbb{R}$, and the kernel κ and right-hand side b are smooth functions on $\Omega \times \Omega$ and Ω, respectively. The computation of the solution x of (1) is an ill-posed problem because i) the integral equation might not have a solution, ii) the solution might not be unique, and iii) the solution – if it exists and is unique – does not depend continuously on the right-hand side. The computation of a meaningful approximate solution of (1) in finite precision arithmetic therefore is delicate; see, e.g., Engl et al. [8] or Groetsch [11] for discussions on the solution of ill-posed problems. In the present paper, we assume that (1) is consistent and has a solution in a Hilbert space $\mathcal{X}$ with norm $\|\cdot\|$. For instance, $\mathcal{X}$ may be $L_2(\Omega)$. Often one is interested in determining the unique solution of minimal norm. We denote this solution by $\hat{x}$.

In applications, generally, not b, but a corrupted version, which we denote by b^δ, is available. We assume that a constant $\delta > 0$ is known, such that the inequality

$$\|b^\delta - b\| \leq \delta \tag{2}$$

holds. The difference $b^\delta - b$ may, for instance, stem from measurement errors and is referred to as "noise."

Our task is to determine an approximate solution x^δ of

$$\int_{\Omega} \kappa(t,s)x(s)ds = b^\delta(t), \qquad t \in \Omega, \tag{3}$$

such that x^δ provides an accurate approximation of $\hat{x}$. Equation (3) is not required to be consistent.

In operator notation, we express (1) and (3) as

$$Ax = b \tag{4}$$

and

$$Ax = b^\delta, \tag{5}$$

respectively. The operator $A : \mathcal{X} \to \mathcal{Y}$ is compact, where $\mathcal{X}$ and $\mathcal{Y}$ are Hilbert spaces. Thus, A has an unbounded inverse and may be singular. The right-hand side b is assumed to be in the range of A, denoted by $\mathcal{R}(A)$, but b^δ generally is not.

We seek to determine an approximation of the minimal-norm solution $\hat{x}$ of (4) by first replacing the operator A in (5) by an operator $A_{\rm reg}$ that approximates A and has a bounded inverse on $\mathcal{Y}$, and then solving the modified equation so obtained,

$$A_{\rm reg} x = b^\delta. \tag{6}$$

The replacement of A by $A_{\rm reg}$ is referred to as regularization and $A_{\rm reg}$ as a regularized operator. We would like to choose $A_{\rm reg}$ so that the solution x^δ of (6) is a meaningful approximation of $\hat{x}$.

One of the most popular regularization methods is Tikhonov regularization, which in its simplest form is defined by

$$(A^* A + \lambda I)x = A^* b^\delta, \tag{7}$$

i.e., $(A_{\rm reg})^{-1} = (A^*A + \lambda I)^{-1} A^*$. Here I is the identity operator and $\lambda > 0$ is a regularization parameter. The latter determines how sensitive the solution x^δ of (7) is to perturbations in the right-hand side b^δ and how close x^δ is to the solution $\hat{x}$ of (4); see, e.g., Engl et al. [8] and Groetsch [11] for discussions on Tikhonov regularization.

For any fixed $\lambda > 0$, equation (7) is a Fredholm integral equation of the second kind and, therefore, the computation of its solution is a well-posed problem. Several two- and multi-level methods for the solution of the Tikhonov equation (7) have been described in the literature; see, e.g., Chen et al. [6], Hanke and Vogel [13], Huckle and Staudacher [15], Jacobsen et al. [16], and King [17]. For a large number of ill-posed problems, these methods determine accurate approximations of the solution of the Tikhonov equation (7) faster than standard (one-level) iterative methods.

The cascadic multilevel method of the present paper is applied to the unregularized problem (5). Regularization is achieved by restricting the number of iterations on each level using the discrepancy principle, defined in Section 2. Thus, the operator $A_{\rm reg}$ associated with the cascadic multilevel method is defined implicitly. For instance, let the basic iterative scheme be CGNR (the conjugate gradient method applied to the normal equations). We apply CGNR on the coarsest discretization level until the computed approximate solution satisfies the discrepancy principle. Then the coarsest-level solution is prolongated to the next finer discretization level and iterations with CGNR are carried out on this level until the computed approximate solution satisfies the discrepancy principle. The computations are continued in this manner until an approximate solution on the finest discretization level has been found that satisfies the discrepancy principle. We remark that if the iterations are not terminated sufficiently early, then the error in b^δ may propagate to the computed approximate solution and render the latter a useless approximation of $\hat{x}$. We establish in Section 3 that the CGNR-based cascadic multilevel method is a regularization method in a well-defined sense.

The application of CGNR as basic iterative method in the multilevel method is appropriate when A is not self-adjoint. The computed iterates live in $\mathcal{R}(A^*)$ and therefore are orthogonal to $\mathcal{N}(A)$, the null space of A. Here and throughout this paper A^* denotes the adjoint of A.

When A is self-adjoint, the computational work often can be reduced by using an iterative method of conjugate gradient (CG) type different from CGNR as basic iterative method. Section 3 also describes multilevel methods for self-adjoint ill-posed problems based on a suitable minimal residual method.

The application of multigrid methods directly to the unregularized problem (5) recently also has been proposed by Donatello and Serra-Capizzano [7], who with computed examples show the promise of this approach. The regularization properties of the multigrid methods used are not analyzed in [7].

Cascadic multilevel methods typically are able to determine an approximate solution of (5) that satisfies the discrepancy principle with less arithmetic work than application of the CG-type method, which is used for the basic iterations, on the finest level only. We refer to the latter method as a one-level CG-type method, or simply as a CG-type method. A cascadic Landweber-based iterative method for nonlinear ill-posed problems has been analyzed by Scherzer [22]. Numerical examples reported in [22] show this method to require many iterations.

Multilevel methods have for many years been applied successfully to the solution of well-posed boundary value problems for partial differential equations; see, e.g., Trottenberg et al. [23] and references therein. In particular, a CG-based cascadic multigrid method has been analyzed by Bornemann and Deuflhard [5]. However, the design of multilevel methods for this kind of problems differs significantly from multilevel methods for ill-posed problems. This depends on that highly oscillatory eigenfunctions, which need to be damped, in the former problems are associated with eigenvalues of large magnitude, while they are associated with eigenvalues of small magnitude for the latter problems.

This paper is organized as follows. Section 2 reviews CG-type methods and the discrepancy principle. In particular, we discuss the regularization properties of CG-type methods. Cascadic multilevel methods based on different CG-type methods are described in Sections 3, where also regularization properties of these methods are shown. Section 4 presents a few computed examples and concluding remarks can be found in Section 5.

2 CG-type methods and the discrepancy principle

Most of this section focuses on the CGNR method and its regularizing property. This method is suitable for the approximate solution of equation (5) when A is not self-adjoint. At the end of this section, we review related results for CG-type methods for ill-posed problems with a self-adjoint operator A. The regularization property of these methods is central for our proofs in Section 3 of the fact that cascadic multilevel methods based on these CG-type methods are regularization methods.

The CGNR method for the solution of (5) is the conjugate gradient method applied to the normal equations

$$A^*Ax = A^*b^\delta \tag{8}$$

associated with (5). Let x_0^δ denote the initial iterate and define the associated residual vector $r_0^\delta = b^\delta - Ax_0^\delta$. Introduce the Krylov subspaces

$$\mathcal{K}_k(A^*A, A^*r_0^\delta) = \text{span}\{A^*r_0^\delta, (A^*A)A^*r_0^\delta, \ldots, (A^*A)^{k-1}A^*r_0^\delta\} \tag{9}$$

for $k = 1, 2, 3, \ldots$. The kth iterate, x_k^δ, determined by CGNR applied to (5) with initial iterate x_0^δ satisfies

$$\|Ax_k^\delta - b^\delta\| = \min_{x \in x_0^\delta + \mathcal{K}_k(A^*A, A^*r_0^\delta)} \|Ax - b^\delta\|, \quad x_k^\delta \in \; x_0^\delta + \mathcal{K}_k(A^*A, A^*r_0^\delta), \tag{10}$$

i.e., CGNR is a minimal residual method. Note that if x_0^δ is orthogonal to $\mathcal{N}(A)$, then so are all iterates x_k^δ, $k = 1, 2, 3, \ldots$. This is the case, e.g., when $x_0^\delta = 0$. If in addition b^δ is noise-free, i.e., if b^δ is replaced by b in (10), then the iterates x_k^δ determined by CGNR converge to $\hat{x}$, the minimal-norm solution of (4).

Algorithm 1 describes CGNR. The computations can be organized so that each iteration only requires the evaluation of one matrix-vector product with A and one with A^*. The algorithm is also referred to as CGLS; see Björck [4] for discussions on CGLS and LSQR, an alternate implementation.

Algorithm 1: CGNR

Input: A, b^δ, x_0^δ, $m \geq 1$ (Number of iterations);
Output: approximate solution x_m^δ of (4);
$r_0^\delta := b^\delta - Ax_0$; $d := A^* r_0^\delta$;
$k = 0$;
while $k < m$ do
$\alpha := \|A^* r_k^\delta\|^2 / \|Ad\|^2$;
$x_{k+1}^\delta := x_k^\delta + \alpha d$;
$r_{k+1}^\delta := r_k^\delta - \alpha Ad$;
$\beta := \|A^* r_{k+1}^\delta\|^2 / \|A^* r_k^\delta\|^2$;
$d := A^* r_{k+1}^\delta + \beta d$;
$k := k + 1$;
endwhile

The residual vector $r = b^\delta - Ax$ is sometimes referred to as the discrepancy associated with x. The discrepancy principle furnishes a criterion for choosing the number of iterations, m, in Algorithm 1.

Definition (Discrepancy Principle). *Let b^δ satisfy (2) for some $\delta \geq 0$, and let $\tau > 1$ be a constant independent of δ. The vector x is said to satisfy the discrepancy principle if $\|b^\delta - Ax\| \leq \tau\delta$.*

We will be interested in reducing $\delta > 0$, while keeping $\tau > 1$ fixed, in the Discrepancy Principle.

Stopping Rule 2.1 *Let b^δ, δ, and τ be the same as in the Discrepancy Principle. Terminate the iterations with Algorithm 1 when, for the first time,*

$$\|b^\delta - Ax_k^\delta\| \leq \tau\delta. \tag{11}$$

Denote the resulting stopping index by $k(\delta)$.

Note that generally $k(\delta)$ increases monotonically as δ decreases to zero with τ kept fixed. This depends on that the right-hand side bound in (11) gets smaller when δ decreases. Bounds for the growth of $k(\delta)$ as δ decreases are provided by [12, Corollary 6.18].

An iterative method equipped with Stopping Rule 2.1 is said to be a *regularization method* if the computed iterates $x_{k(\delta)}^\delta$ satisfy

$$\lim_{\delta \searrow 0} \sup_{\|b - b^\delta\| \leq \delta} \|\hat{x} - x_{k(\delta)}^\delta\| = 0, \tag{12}$$

where $\hat{x}$ is the minimal-norm solution of (4). We remark that the constant τ in Stopping Rule 2.1 is kept fixed as δ is decreased to zero. The following result, first proved by Nemirovskii [18], shows that CGNR is a regularization method when applied to the solution of (5). A proof is also provided by Hanke [12, Theorem 3.12].

Theorem 2.2 *Assume that equation (4) is consistent and that b^δ satisfies (2) for some $\delta > 0$. Terminate the iterations with Algorithm 1 according to Stopping Rule 2.1, with $\tau > 1$ a fixed constant. Let $k(\delta)$ denote the stopping index and $x^\delta_{k(\delta)}$ the associated iterate determined by CGNR. Then $x^\delta_{k(\delta)} \to \hat{x}$ as $\delta \searrow 0$, where $\hat{x}$ denotes the minimal-norm solution of (4).*

We turn to the case when the operator A is self-adjoint. Let the kth iterate, x^δ_k, be determined by a minimal residual method so as to satisfy

$$\|Ax^\delta_k - b^\delta\| = \min_{x \in x^\delta_0 + \mathcal{K}_k(A, Ar^\delta_0)} \|Ax - b^\delta\|, \qquad x^\delta_k \in x^\delta_0 + \mathcal{K}_k(A, Ar^\delta_0), \tag{13}$$

with initial iterate x^δ_0 and $r^\delta_0 = b^\delta - Ax^\delta_0$. Note that the iterates are orthogonal to $\mathcal{N}(A)$ provided that x^δ_0 is orthogonal to $\mathcal{N}(A)$. The following analog of Theorem 2.2 holds.

Theorem 2.3 *Let the operator A be self-adjoint, let equation (4) be consistent, and assume that b^δ satisfies (2) for some $\delta > 0$. Let the iterates x^δ_k be generated by a minimal residual method and satisfy (13). Terminate the iterations according to Stopping Rule 2.1, with $\tau > 1$ a fixed constant. Let $k(\delta)$ denote the stopping index and $x^\delta_{k(\delta)}$ the associated iterate. Then $x^\delta_{k(\delta)} \to \hat{x}$ as $\delta \searrow 0$, where $\hat{x}$ denotes the minimal-norm solution of (4).*

The above result is shown by Hanke [12, Theorem 6.15]. The case when A is semidefinite was first discussed by Plato [21]; see also Hanke [12, Chapter 3].

An implementation of a minimal residual method, referred to as MR-II, which determines iterates x^δ_k that satisfy (13) is provided by Hanke [12]. We use this implementation in computed examples of Section 4. The iterates (13) also can be determined by a simple modification of the MINRES algorithm by Paige and Saunders [19]. The computation of the iterate x^δ_k defined by (13) requires $k+1$ matrix-vector product evaluations with A. The exploitation of self-adjointness of A generally reduces the number of matrix-vector product evaluations required to determine an iterate x^δ_k that satisfies the discrepancy principle.

3 Multilevel methods based on CG-type iteration

In this section we present multilevel methods for the solution of (5). We first discuss the use of CGNR as basic iterative method and show that with an appropriate stopping rule, the multilevel method for the solution of (5) so obtained is a regularization method. The use of other CG-type methods as basic iterative method in multilevel methods is discussed at the end of this section.

Let $S_1 \subset S_2 \subset \ldots \subset S_\ell$ be a sequence of nested linear subspaces of $L_2(\Omega)$ of increasing dimensions with $S_\ell = L_2(\Omega)$. Each subspace is equipped with a norm, which we denote by $\|\cdot\|$. Introduce, for $1 \le i \le \ell$, the restriction and prolongation operators $R_i : L_2(\Omega) \to S_i$ and $Q_i : S_i \to L_2(\Omega)$, respectively, with R_ℓ and Q_ℓ identity operators, and define

$$b_i = R_i b, \qquad b_i^{\delta_i} = R_i b^\delta, \qquad A_i = R_i A Q_i.$$

When A is self-adjoint, it is convenient to choose $Q_i = R_i^*$, the adjoint of R_i. Thus, A_i is the restriction of A to S_i with $A_\ell = A$. It is convenient to consider the noise in the restriction $b_i^{\delta_i}$ of b^δ to S_i to be independent for $1 \le i \le \ell$ in the convergence proofs below. We therefore use the superscript δ_i for the restriction $b_i^{\delta_i}$ of b^δ. We require that there are constants c_i, independent of δ_i, such that

$$\|b_i - b_i^{\delta_i}\| \le c_i \delta_i, \qquad 1 \le i \le \ell, \tag{14}$$

for all $\delta_i \ge 0$, where $\|\cdot\|$ denotes the norm of S_i. The coefficient c_i depends on the norm of S_i; this is illustrated by Remark 4.1 of Section 4. Below, we reduce the $\delta_i > 0$, while keeping the c_i fixed. We assume that the restriction operators R_i are such that $\delta_i = \delta_i(\delta)$ decrease as δ decreases with $\delta_i(0) = 0$ for $1 \le i \le \ell$. In the computations reported in Section 4, we let

$$\delta_i = \delta, \qquad 1 \le i \le \ell. \tag{15}$$

We also need prolongation operators $P_i : S_{i-1} \to S_i$ for $1 < i \le \ell$. In the computed examples, we let the P_i be piecewise linear; see Section 4 for details. In addition, Algorithm 2 requires a mapping P_1, such that $P_1(0) = 0 \in S_1$.

The "one-way" CGNR-based multilevel method of the present paper first determines an approximate solution of $A_1 x = b_1^{\delta_1}$ in S_1 by CGNR. The iterations with CGNR are terminated as soon as an iterate that satisfies a stopping rule related to the discrepancy principle has been determined. This iterate is mapped from S_1 into S_2 by P_2. We then apply CGNR to compute a correction in S_2 of this mapped iterate. Again, the CGNR-iterations are terminated by a stopping rule related to the discrepancy principle. The approximate solution in S_2 determined in this fashion is mapped into S_3 by P_3. The computations are continued in this manner until an approximation of $\hat{x}$ has been determined in S_ℓ. We refer to this scheme as Multilevel CGNR (ML-CGNR).

Algorithm 2: ML-CGNR

Input: A, b^δ, $\ell \geq 1$ (number of levels), $\delta_1, \delta_2, \ldots, \delta_\ell$, $c_1, c_2, \ldots, c_\ell$ (coefficients for the stopping rule);

Output: approximate solution $x_\ell^\delta \in S_\ell$ of (4);

$x_0^{\delta_0} := 0;$
for $i := 1, 2, \ldots, \ell$ do
$x_{i,0}^{\delta_{i-1}} := P_i x_{i-1}^{\delta_{i-1}};$
$\Delta x_{i,m_i(\delta_i)}^{\delta_i} := \mathrm{CGNR}(A_i, b_i^{\delta_i} - A_i x_{i,0}^{\delta_{i-1}});$
$x_i^{\delta_i} := x_{i,0}^{\delta_{i-1}} + \Delta x_{i,m_i(\delta_i)}^{\delta_i};$
endfor
$x_\ell^\delta := x_\ell^{\delta_\ell};$

Algorithm 2 describes the ML-CGNR method. In the algorithm, $\Delta x_{i,m_i(\delta_i)}^{\delta_i} := \mathrm{CGNR}(A_i, b_i^{\delta_i} - A_i x_{i,0}^{\delta_{i-1}})$ denotes the computation of the approximate solution $\Delta x_{i,m_i(\delta_i)}^{\delta_i}$ of the equation

$$A_i z = b_i^{\delta_i} - A_i x_{i,0}^{\delta_{i-1}} \tag{16}$$

by application of $m_i(\delta_i)$ steps of CGNR with initial iterate $\Delta x_{i,0} = 0$. We discuss the choice of $m_i(\delta_i)$ below.

Let x_i denote the minimal-norm solution of the equation

$$A_i z = b_i \tag{17}$$

on level i. We assume that the restriction operator R_i is such that (17) is consistent. Moreover, it is convenient to require the range of the prolongation operator P_i to be orthogonal to $\mathcal{N}(A_i)$, i.e.,

$$\mathcal{R}(P_i) \subset \mathcal{R}(A_i^*), \qquad 1 < i \leq \ell. \tag{18}$$

We comment on this requirement at the end of this section.

Introduce the equation

$$A_i z = b_i - A_i x_{i,0}^{\delta_{i-1}}, \tag{19}$$

which is consistent and has minimal-norm solution

$$\Delta x_i = x_i - x_{i,0}^{\delta_{i-1}}. \tag{20}$$

Algorithm 2 implicitly defines a regularized operator A_{reg} by taking $m_i(\delta_i)$ steps of CGNR on level i for $i = 1, 2, \ldots, \ell$. In general, the iterates $\Delta x_{i,m_i}^{\delta_i}$

do not converge to the minimal-norm solution (20) of (19) as the number of iterations, m_i, increases without bound; in fact, the norm $\|\Delta x_i - \Delta x^{\delta}_{i,m_i}\|$ typically grows with m_i for m_i sufficiently large. It is therefore important to terminate the iterations on each level after a suitable number of step.

Stopping Rule 3.1 *Let the c_i and δ_i be the same as in (14) and denote the iterates determined on level i by CGNR applied to the solution of (16) by $\Delta x^{\delta_i}_{i,m_i}$, $m_i = 1, 2, \dots$, with initial iterate $\Delta x_{i,0} = 0$. Terminate the iterations as soon as an iterate has been determined, such that*

$$\|b^{\delta_i}_i - A_i x^{\delta_{i-1}}_{i,0} - A_i \Delta x^{\delta_i}_{i,m_i}\| \leq \tau c_i \delta_i, \tag{21}$$

where $\tau > 1$ is a constant independent of the c_i and δ_i. We denote the termination index by $m_i(\delta_i)$ and the corresponding iterate by $\Delta x^{\delta_i}_i = \Delta x^{\delta_i}_{i,m_i(\delta_i)}$.

The following theorem discusses convergence of the approximate solutions $x^{\delta_i}_i$ determined by Algorithm 2 on level i towards the minimal-norm solutions x_i of the noise-free projected problems (17). In particular, $x^{\delta_\ell}_\ell$ converges to $\hat{x}$, the minimal-norm solution of the noise-free problem (4). A analogous result for equations (4) and (5) with a self-adjoint operator A, with CGNR replaced by MR-II in Algorithm 2 is shown towards the end of the section.

Theorem 3.2 *Let $A_\ell = A$ and $b_\ell = b$. Assume that the equations (17) are consistent for $1 \leq i \leq \ell$ and that (18) holds. Let the projected contaminated right-hand sides $b^{\delta_i}_i$ satisfy (14). Terminate the iterations with CGNR in Algorithm 2 on levels $1, 2, \dots, \ell$ according to Stopping Rule 3.1. This yields the iterates $x^{\delta_i}_i$ for levels $1 \leq i \leq \ell$. Then the ML-CGNR method described by Algorithm 2 is a regularization method on each level, i.e.,*

$$\lim_{\delta_i \searrow 0} \sup_{\|b_i - b^{\delta_i}_i\| \leq c_i \delta_i} \|x_i - x^{\delta_i}_i\| = 0, \qquad 1 \leq i \leq \ell, \tag{22}$$

where x_i is the minimal norm solution of (17) with $x_\ell = \hat{x}$.

Proof. We will show that for an arbitrary $\epsilon > 0$, there are positive $\delta_1, \delta_2, \dots, \delta_\ell$, depending on ϵ, such that

$$\|x_i - x^{\delta_i}_i\| \leq \epsilon, \qquad 1 \leq i \leq \ell. \tag{23}$$

This then shows (22).

First consider level $i = 1$ and apply CGNR to the solution of

$$A_1 z = b^{\delta_1}_1.$$

By Theorem 2.2 there is a $\delta_1 > 0$, such that equation (23) holds for $i = 1$.

We turn to level $i = 2$. Since $\mathcal{R}(P_2)$ is orthogonal to $\mathcal{N}(A_2)$, the vector $x_{2,0}^{\delta_1}$ has no component in $\mathcal{N}(A_2)$. It follows from (14) that

$$\|b_2 - A_2 x_{2,0}^{\delta_1} - (b_2^{\delta_2} - A_2 x_{2,0}^{\delta_1})\| \leq c_2\delta_2.$$

Application of CGNR to (16) for $i = 2$ with initial iterate $\Delta x_{2,0} = 0$ and Stopping Rule 3.1 yields the approximate solution $\Delta x_2^{\delta_2} = \Delta x_{2,m_2(\delta_2)}^{\delta_2}$. It follows from Theorem 2.2 that we may choose δ_2, so that

$$\|\Delta x_2 - \Delta x_2^{\delta_2}\| \leq \epsilon,$$

where Δx_2 is defined by (20). Using the definition of $x_2^{\delta_2}$ in Algorithm 2, we obtain (23) for $i = 2$. We now can proceed in this fashion for increasing values of i. This shows (23) for all i, and thereby the theorem. $\Box$

Corollary 3.1 *Let $A_\ell = A$, $b_\ell = b$, $b_\ell^{\delta_\ell} = b^\delta$, $\delta_\ell = \delta$, and $c_\ell = 1$, where δ satisfies (2). Assume that $\mathcal{R}(P_\ell) \subset \mathcal{R}(A^*)$. Let the products $c_i\delta_i > 0$, $1 \leq i < \ell$, be fixed and large enough to secure that Stopping Rule 3.1 yields termination of the iterations after finitely many steps on levels $1 \leq i < \ell$. Then the ML-CGNR method described by Algorithm 2 with Stopping Rule 3.1 is a regularization method on level ℓ, i.e.,*

$$\lim_{\delta \searrow 0} \sup_{\|b - b^\delta\| \leq \delta} \|\hat{x} - x_\ell^\delta\| = 0, \tag{24}$$

where x_ℓ^δ is the approximate solution determined by Algorithm 2 on level ℓ.

Proof. We first note that consistency of (17) holds for $i = \ell$ by the assumptions made in Section 1. The sole purpose of the computations on levels $1 \leq i < \ell$ is to determine an initial approximate solution $x_{\ell,0}^{\delta_{\ell-1}}$ for level ℓ. We are not concerned with how the iterates $x_i^{\delta_i}$ on levels $1 \leq i < \ell$ relate to the minimal norm solutions of the systems (17) for $1 \leq i < \ell$. Therefore property (18) only has to hold for $i = \ell$. Since the systems (17) are not required to be consistent for $1 \leq i < \ell$, the right-hand sides in (21), i.e., the products $c_i\delta_i$, $1 \leq i < \ell$, have to be large enough to secure finite termination of the iterations on these levels. The corollary now follows from Theorem 2.2 or from the last step ($i = \ell$) of the proof of Theorem 3.2. $\Box$

We turn to the situation when A is self-adjoint. Then CGNR is replaced by MR-II in Algorithm 2. We refer to the method so obtained as ML-MR-II.

Theorem 3.3 *Let the conditions of Theorem 3.2 hold and assume that A is self-adjoint. Let the iterates $x_i^{\delta_i}$, $1 \leq i \leq \ell$, be generated by the ML-MR-*

II method, i.e., by Algorithm 2 with CGNR replaced by MR-II. Terminate the iterations with MR-II in Algorithm 2 on levels $1, 2, \ldots, \ell$ *according to Stopping Rule 3.1 This yields the iterates* $x_i^{\delta_i}$ *for levels* $1 \leq i \leq \ell$. *The ML-MR-II method so defined is a regularization method on each level, i.e.,*

$$\lim_{\delta_i \searrow 0} \sup_{\|b_i - b_i^{\delta_i}\| \leq c_i \delta_i} \|x_i - x_i^{\delta_i}\| = 0, \qquad 1 \leq i \leq \ell,$$

where x_i *is the minimal norm solution of (17) with* $x_\ell = \hat{x}$.

Proof. The result can be shown similarly as Theorem 3.2 by using the properties of the iterates (13) collected in Theorem 2.3. □

Corollary 3.2 *Let* $A_\ell = A$, $b_\ell = b$, $b_\ell^{\delta_\ell} = b^\delta$ *be as in Theorem 3.3 with* $\delta_\ell = \delta$, *and let the products* $c_i \delta_i$, $1 \leq i < \ell$, *satisfy the conditions of Corollary 3.1. Assume that* $\mathcal{R}(P_\ell) \subset \mathcal{R}(A)$. *Let the iterates* $x_i^{\delta_i}$, $1 \leq i \leq \ell$, *be generated by the ML-MR-II method, i.e., by Algorithm 2 with CGNR replaced by MR-II. The ML-MR-II method so defined is a regularization method on level* ℓ, *i.e.,*

$$\lim_{\delta \searrow 0} \sup_{\|b - b^\delta\| \leq \delta} \|\hat{x} - x_\ell^\delta\| = 0, \tag{25}$$

where x_ℓ^δ *is the approximate solution determined by Algorithm 2 on level* ℓ.

Proof. The result can be shown similarly as Corollary 3.1. □

We conclude this section with a comment on the condition that $\mathcal{R}(P_\ell)$ be orthogonal to $\mathcal{N}(A)$. The other inclusions in (18) can be treated similarly. The conditions (18) are difficult to verify but have not been an issue in actual computations; see Section 4 for further comments. Let $P_{\mathcal{N}(A)}$ denote the orthogonal projector onto $\mathcal{N}(A)$. Then

$$x_{\ell,0}^{\delta_{\ell-1}} = (I - P_{\mathcal{N}(A)}) x_{\ell,0}^{\delta_{\ell-1}} + P_{\mathcal{N}(A)} x_{\ell,0}^{\delta_{\ell-1}}.$$

The computation of $x_\ell^{\delta_\ell}$ from $x_{\ell,0}^{\delta_{\ell-1}}$ in Algorithm 2 proceeds independently of $P_{\mathcal{N}(A)} x_{\ell,0}^{\delta_{\ell-1}}$. We have

$$x_\ell^{\delta_\ell} = (I - P_{\mathcal{N}(A)}) x_\ell^{\delta_\ell} + P_{\mathcal{N}(A)} x_{\ell,0}^{\delta_{\ell-1}}.$$

Hence,

$$\lim_{\delta \searrow 0} \|\hat{x} - x_\ell^{\delta_\ell}\| = \|P_{\mathcal{N}(A)} x_{\ell,0}^{\delta_{\ell-1}}\|,$$

	CGNR		ML-CGNR	
$\frac{\delta}{\|b\|}$	$m(\delta)$	$\frac{\|x^\delta_{8,m(\delta)}-\hat{x}\|}{\|\hat{x}\|}$	$m_i(\delta)$	$\frac{\|x^\delta_{8,m_8(\delta)}-\hat{x}\|}{\|\hat{x}\|}$
$1\cdot 10^{-1}$	3	0.0934	2, 1, 1, 1, 1, 1, 1, 1	0.0842
$1\cdot 10^{-2}$	4	0.0248	5, 5, 3, 2, 1, 1, 1, 1	0.0343
$1\cdot 10^{-3}$	4	0.0243	8, 6, 6, 4, 3, 1, 1, 1	0.0243
$1\cdot 10^{-4}$	11	0.0064	9, 13, 9, 9, 5, 4, 3, 2	0.0076

Table 1
Example 4.1: Termination indices $m(\delta)$ for CGNR with Stopping Rule 2.1 determined by δ and $\tau = 1.25$, as well as relative errors in the computed approximate solutions $x^\delta_{8,m(\delta)}$, and termination indices $m_1(\delta), m_2(\delta), \ldots, m_k(\delta)$ for ML-CGNR with Stopping Rule 3.1 determined by δ and $c = 1.25$, as well as relative errors in the computed approximate solutions $x^\delta_{8,m_8(\delta)}$.

i.e., we obtain an accurate approximation of $\hat{x}$ as $\delta \searrow 0$ if $P_{\mathcal{N}(A)}x^{\delta_{\ell-1}}_{\ell,0}$ is small. We remark that in all computed examples of Section 4, $P_{\mathcal{N}(A)}x^{\delta_{\ell-1}}_{\ell,0}$ is tiny.

4 Computed examples

This section compares ML-CGNR and ML-MR-II to one-level CGNR and MR-II, respectively. In the computed examples, the sets $S_1 \subset S_2 \subset \ldots \subset S_\ell$ are used to represent discretizations of continuous functions in $L_2(\Omega)$ with $\dim(S_i) = n_i$. Specifically, in the first two examples Ω is an interval and S_i is the set of piecewise linear functions determined by interpolation at n_i equidistant nodes. We may identify these functions with their values at the nodes, which we represent by vectors in $\mathbb{R}^{n_i}$ equipped with the weighted Euclidean norm

$$\|v\| = \left(\frac{1}{n_i}\sum_{j=1}^{n_i}(v^{(j)})^2\right)^{1/2}, \qquad v = [v^{(1)}, v^{(2)}, \ldots, v^{(n_i)}]^T \in S_i. \tag{26}$$

An inner product is defined similarly. Vectors with subscript i live in $\mathbb{R}^{n_i}$. The set S_ℓ is used to represent functions on the finest level and is identified with $\mathbb{R}^{n_\ell}$. We let $b = b_\ell$, $\hat{x} = x_\ell$, and $A = A_\ell$. The restriction operator R_i consists of n_i rows of I_ℓ, the identity matrix of order n_ℓ. It follows that $R_i R_i^* = I_i$ for all i. All computations are carried out in Matlab with machine epsilon $\epsilon \approx 2\cdot 10^{-16}$.

Example 4.1. Consider the Fredholm integral equation of the first kind

$$\int_{-6}^{6} \kappa(t,s)x(s)ds = b(t), \qquad -6 \le t \le 6, \tag{27}$$

discussed by Phillips [20]. Its solution, kernel, and right-hand side are given by

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

We discretize this integral equation by $\ell = 8$ Nyström methods based on composite trapezoidal quadrature rules with equidistant nodes. The number of nodes on the ith level is $n_i = 4\cdot 2^i+1$; thus, there are 1025 nodes in the finest discretization. This yields the nonsymmetric matrix $A = A_\ell \in \mathbb{R}^{1025\times 1025}$ and right-hand side $b = b_\ell \in \mathbb{R}^{1025}$. The condition number of the matrix A, defined by $\kappa(A) = \|A\|\|A^{-1}\|$ with $\|\cdot\|$ denoting the operator norm induced by the norm (26) for $i=\ell$, is about $1.9\cdot 10^{10}$, i.e., A is nearly numerically singular. There are only 9 nodes on the coarsest grid and the matrix $A_1 \in \mathbb{R}^{9\times 9}$ is not very ill-conditioned; we have $\kappa(A_1) = 4.2\cdot 10^1$.

In order to determine the "noisy" right-hand side $b_8^\delta = b^\delta$, we generate a vector w with normally distributed entries with mean zero and variance one, define the "noise-vector"

$$e = w\|b\|\cdot 10^{-\eta}, \tag{28}$$

for some $\eta \geq 0$, and let

$$b^\delta = b + e. \tag{29}$$

It follows from the strong law of large numbers that the noise level satisfies

$$\frac{\|b-b^\delta\|}{\|b\|} = \frac{\|e\|}{\|b\|} \approx 1\cdot 10^{-\eta}. \tag{30}$$

We use the values

$$\delta = \delta_1 = \cdots = \delta_\ell = \|b\|\cdot 10^{-\eta} \tag{31}$$

in the Stopping Rules 2.1 and 3.1. Moreover, we let the coefficients c_i in (21) be equal. Therefore, we generally omit the subscripts i of the δ_i and c_i in (14) and (21) in this section.

Column 2 of Table 1 displays the number of iterations $m(\delta)$ required by (one-level) CGNR applied to $A_8 x_8^\delta = b_8^\delta$ with initial iterate $x_{8,0}^\delta = 0$ to obtain an approximate solution $x_{8,m(\delta)}^\delta$ that satisfies Stopping Rule 2.1 for $\tau = 1.25$ and different noise levels (30). Column 3 of the table shows the relative error $\|x_{8,m(\delta)}^\delta - \hat{x}\|/\|\hat{x}\|$ in the computed approximate solutions. These columns show the accuracy and the number of iterations to increase as the noise level is decreased.

We turn to the ML-CGNR method implemented by Algorithm 2. The noisy right-hand side b^δ is projected to the subspaces S_i recursively to obtain the vectors b_i^δ, $1 \leq i \leq 8$. The jth component, $(b_i^\delta)^{(j)}$, of b_i^δ is given by

$$(b_i^\delta)^{(j)} = (b_{i+1}^\delta)^{(2j-1)}, \qquad 1 \leq j \leq n_i, \qquad 1 \leq i \leq 8. \tag{32}$$

This defines the restriction operators R_i. The prolongation operators P_i are defined by local averaging,

$$P_i = \begin{bmatrix} 1 & & & & & & \\ 1/2 & 1/2 & & & & & \\ 1/4 & 1/2 & 1/4 & & & & \\ & 1/2 & 1/2 & & & & \\ & 1/4 & 1/2 & 1/4 & & & \\ & & 1/2 & 1/2 & & & \\ & & \ddots & \ddots & \ddots & & \\ & & & 1/4 & 1/2 & 1/4 \\ & & & & 1/2 & 1/2 \\ & & & & & 1 \end{bmatrix} \in \mathbb{R}^{n_i \times n_{i-1}}.$$

The performance the ML-CGNR method implemented by Algorithm 2 with Stopping Rule 3.1, determined by $c = 1.25$ and several values of δ, is illustrated by Table 1. The columns with header $m_i(\delta)$ show, from left to right, the number of iterations required on the levels $1, 2, \ldots, 8$. Since for all i the matrix A_i has four times as many entries as A_{i-1}, the dominating computational work is the evaluation of matrix-vector products on the finest discretization level. In this example, ML-CGNR reduces the number of iterations on the finest level to at most 1/3 of number of iterations required by one-level CGNR.

The last column of Table 1 displays the relative error in the computed approximate solutions $x_{8,m_8(\delta)}^\delta$ determined by Algorithm 2. The accuracy is seen

	CGNR		ML-CGNR	
$\frac{\delta}{\|b\|}$	$m(\delta)$	$\frac{\|x^{\delta}_{8,m(\delta)}-\hat{x}\|}{\|\hat{x}\|}$	$m_i(\delta)$	$\frac{\|x^{\delta}_{8,m_8(\delta)}-\hat{x}\|}{\|\hat{x}\|}$
$1 \cdot 10^{-1}$	2	0.3412	2, 1, 1, 1, 1, 1, 1, 1	0.2686
$1 \cdot 10^{-2}$	3	0.1662	2, 2, 1, 3, 1, 1, 1, 1	0.1110
$1 \cdot 10^{-3}$	3	0.1657	3, 3, 2, 1, 1, 1, 1, 1	0.1065
$1 \cdot 10^{-4}$	4	0.1143	4, 3, 3, 3, 2, 1, 1, 1	0.0669

Table 2
Example 4.2: Termination indices $m(\delta)$ for CGNR with Stopping Rule 2.1 determined by δ and $\tau = 1.25$, as well as relative errors in the computed approximate solutions $x^{\delta}_{8,m(\delta)}$, and termination indices $m_1(\delta), m_2(\delta), \ldots, m_k(\delta)$ for ML-CGNR with Stopping Rule 3.1 determined by δ and $c = 1.25$, as well as relative errors in the computed approximate solutions $x^{\delta}_{8,m_8(\delta)}$.

to be about the same as for the approximate solutions $x_{8,m(\delta)}$ determined by CGNR, but the computational effort required by ML-CGNR is considerably smaller. □

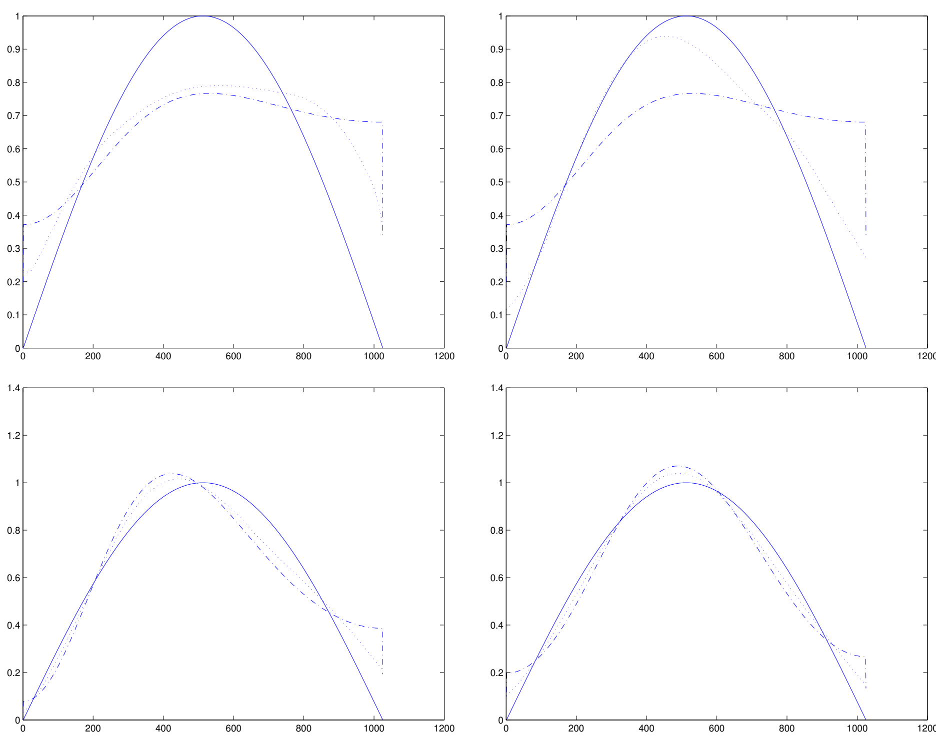

Fig. 1. Example 4.2: The exact solution $\hat{x}$ of the noise-free problem (continuous graphs), the approximate solutions of the noisy problem computed by CGNR (dash–dotted graphs), and by ML-CGNR (dotted graphs). The top-left, top-right, bottom-left, and bottom-right graphs display, in order, results for the noise levels 10^{-1}, 10^{-2}, 10^{-3}, and 10^{-4}.

Example 4.2. We consider the Fredholm integral equation of the first kind,

$$\int_{0}^{\pi/2} \kappa(s,t)x(s)ds = b(t), \qquad 0 \leq t \leq \pi, \tag{33}$$

where $\kappa(s,t) = \exp(s\cos(t))$ and $b(t) = 2\sinh(t)/t$, discussed by Baart [3]. The solution is given by $x(t) = \sin(t)$. We discretize (33) in the same way as in Example 4.1 and use the same restriction and prolongation operators as in that example. Thus, the multilevel method uses eight levels, with the nonsymmetric matrix $A = A_8 \in \mathbb{R}^{1025\times1025}$ representing the integral operator on the finest discretization level. Matlab yield $\kappa(A_8) = 4.9 \cdot 10^{21}$, i.e., the matrix A_8 is numerically singular. The "noise" in $b^\delta = b_8^\delta$ is defined in the same manner as in Example 4.1. Table 2 is analogous to Table 1 and shows the performance of CGNR and ML-CGNR. The latter method is seen to yield better approximations of $\hat{x}$ with less arithmetic effort.

Figure 1 displays the computed solutions for different noise levels. The continuous graphs depict $\hat{x}$, the dash-dotted graphs show approximate solutions determined by CGNR, and the dotted graphs display approximate solutions determined by ML-CGNR. The latter can be seen to furnish more accurate approximations of $\hat{x}$ than the dash-dotted graphs for the same noise level.

The matrices A_i in the present example have "numerical null spaces" of co-dimension about eight. Hence, the matrices $A_2, A_3, \ldots, A_8$ have null spaces of large dimension. Nevertheless, accuracy of the computed solutions x_8^δ for different noise levels is not destroyed by components in the null spaces of the matrices A_i; in fact, ML-MR-II is seen to yield higher accuracy than one-level MR-II. $\square$

Remark 4.1 We are in a position to discuss the choice of constants $c_i = c$ in (14) and (21). Let the components of the vector $w = [w^{(1)}, w^{(2)}, \ldots, w^{(n_k)}]^T \in \mathbb{R}^{n_k}$ be normally distributed with mean zero and variance one. Then $\sum_{j=1}^{n_k}(w^{(j)})^2$ has a Chi-square distribution with n_k degrees of freedom.

Let the vector $e_i = b_i^\delta - b_i$ consist of n_i components of $e_k = b_k^\delta - b_k$. This is the case in Examples 4.1 and 4.2, where $e_i = R_i e_k$. It follows from (28) that the components of e_i can be expressed as

$$e_i^{(j)} = e_k^{(\ell_j)} = w^{(\ell_j)}\|b_k\| \cdot 10^{-\eta}, \tag{34}$$

where $w^{(\ell_j)}$ denotes an enumeration of the entries of w.

Let $P\{\cdot\}$ denote probability. Then it follows from (34) and the definition (31) of δ that

$$
\begin{aligned}
P\{\|b_i - b_i^\delta\| \leq c\delta\} &= P\{\|e_i\|^2 \leq c^2\delta^2\} \\
&= P\left\{\frac{1}{n_i}\sum_{j=1}^{n_i}(e_i^{(j)})^2 \leq c^2\delta^2\right\} \\
&= P\left\{\sum_{j=1}^{n_i}(w^{(\ell_j)})^2 < c^2 n_i\right\}.
\end{aligned}
$$

Using tables for Chi-square distribution, we find that on level $i = 1$ with 9 nodes, the inequality (14) with $c_1 = c = 1.25$ holds with a probability larger than 74%. On level $i = 2$ with 17 nodes and $c_2 = c = 1.25$, the inequality (14) holds with a probability larger than 78%. The probability of (14) being true for $i \geq 3$ with $c_i = c = 1.25$ is much larger. We conclude that the probability of (14) to hold on every level can be increased by choosing more nodes on the coarsest level and, of course, by increasing the values of the c_i. □

In Examples 4.1 and 4.2, we assumed that the error in the vectors b_i^δ is caused by the noise in b^δ. This assumption is justified when there coarse level is fine enough to give a small discretization error. In the following example, this is not the case.

Fig. 2. Example 4.3: The available blurred and noisy image.

Example 4.3. This example is concerned with image deblurring. This problem can be thought of stemming from the discretization of an integral equation of the form (3) on a uniform grid by piecewise constant functions with Ω being

Fig. 3. Example 4.3: Image restored by ML-MR-II.

the unit square. The sets S_i are made up of piecewise constant functions and are identified with vectors in $\mathbb{R}^{n_i}$.

The available blurred and noisy image, shown by Figure 2, is of size 817×817 pixels. The blur is generated by the blurring operator $A \in \mathbb{R}^{817^2 \times 817^2}$ defined by the Matlab code `blur.m` from Regularization Tools by Hansen [14] with parameters `band`$= 33$, `sigma`$= 3$. The block Toeplitz matrix A with banded Toeplitz blocks so obtained models blurring by a Gaussian point spread function with variance `sigma`$^2 = 9$. The parameter `band` specifies the half-bandwidth for the Toeplitz blocks. The matrix A represents the blurring operator on the 4th and finest grid, i.e., $A_4 = A$. We also use the code `blur.m` to generate the matrices $A_1 \in \mathbb{R}^{103^2 \times 103^2}$, $A_2 \in \mathbb{R}^{205^2 \times 205^2}$, and $A_3 \in \mathbb{R}^{409^2 \times 409^2}$ that model the blurring operator on coarser grids. We use different values of the parameter `band` in order to get better approximation of $A_4 = A$ on the coarser grids. For A_2 we use `band`$= 17$, for A_3 `band`$= 9$ and for A_4 `band`$= 5$. All matrices A_i generated are symmetric.

We define the perturbed right-hand side b_4^δ by (29) with $\eta = 1 \cdot 10^{-2}$. The projections b_i^δ, $i = 1, 2, 3$, of b_4^δ onto the coarser grids are determined by considering each b_i^δ a matrix with entries $(b_i^\delta)^{(s,t)}$, i.e., $(b_i^\delta)^{(s,t)}$ is the value of pixel (s, t). The projections are now defined by

$$(b_i^\delta)^{(s,t)} = (b_{i+1}^\delta)^{(2s-1,2t-1)}. \tag{35}$$

This defines the restriction operators R_i.

The prolongation operators P_i are also defined by regarding the images as matrices. Thus, the image represented by x_i^δ on level i has the entries $(x_i^\delta)^{(s,t)}$, and each entry represents a pixel value. The prolongation operator P_i is given by

$$
\begin{aligned}
(x_i^\delta)^{(s,t)} &= (x_{i-1}^\delta)^{((s+1)/2,(t+1)/2)}, && \text{for } s,t \text{ odd},\\
(x_i^\delta)^{(s,t)} &= \tfrac{1}{2}\big((x_{i-1}^\delta)^{((s+1)/2,t/2)} + (x_{i-1}^\delta)^{((s+1)/2,t/2+1)}\big), && \text{for } s \text{ odd},\ t \text{ even},\\
(x_i^\delta)^{(s,t)} &= \tfrac{1}{2}\big((x_{i-1}^\delta)^{(s/2,(t+1)/2)} + (x_{i-1}^\delta)^{(s/2+1,(t+1)/2)}\big), && \text{for } s \text{ even},\ t \text{ odd},\\
(x_i^\delta)^{(s,t)} &= \tfrac{1}{2}\big((x_{i-1}^\delta)^{(s/2,t/2)} + (x_{i-1}^\delta)^{(s/2+1,t/2+1)}\big), && \text{for } s,t \text{ even}.
\end{aligned}
$$

Since the matrices A_i are symmetric, we can use MR-II as the basic iteration scheme in a multilevel method. Due to the large discretization error in the vectors b_1^δ, b_2^δ, and b_3^δ, when compared with b_4^δ, we carry out only one MR-II iteration on level i, for $1 \leq i \leq 3$. Stopping Rule 3.1 is applied on the finest level only. We also refer to this scheme as ML-MR-II.

The computations are terminated after 5 iterations on level $i = 4$. This yields $x_{4,5}^\delta$. Since pixel values are nonnegative, we may set the negative entries of $x_{4,5}^\delta$ to zero. This yields the vector $x_{4,5}^{\delta,+}$ with relative error $\|x_{4,5}^{\delta,+} - \hat{x}\|/\|\hat{x}\| = 0.0596$.

For comparison, we carry out five iterations of (one-level) MR-II on the finest level to obtain $x_{4,5}^\delta$. Setting the negative entries to zero yields $x_{4,5}^{\delta,+}$ with relative error $\|x_{4,5}^{\delta,+} - \hat{x}\|/\|\hat{x}\| = 0.0622$. Note that the computation of the iterates $x_{4,5}^{\delta,+}$ by ML-MR-II and one-level MR-II requires essentially the same arithmetic work, but the iterate determined by ML-MR-II has a smaller relative error.

In the present example, we obtain worse restorations if more than one iterations are carried out on all but the finest level. This depends on that the discretization error in b_i^δ is large compared with the error $b_i - b_i^\delta$ for $1 \leq i \leq 3$. We will discuss the influence of the discretization error further in a forthcoming paper. □

5 Conclusion

This paper shows that cascadic multilevel methods with CGNR or MR-II as basic iteration scheme are regularization methods when applied to ill-posed problems. Computed examples illustrate that the approximate solutions determined by cascadic multilevel methods are of about the same or higher quality than the approximate solutions determined by CGNR or MR-II iterations on the finest level only, but the computational effort required by multilevel methods is lower.

Acknowledgment

We would like to thank Kazim Khan for discussions and comments.

References

[1] G. S. Ammar and W. B. Gragg, The generalized Schur algorithm for the superfast solution of Toeplitz systems, in Rational Approximation and its Applications in Mathematics and Physics, J. Gilewicz, M. Pindor and W. Siemaszko, eds., Lecture Notes in Mathematics # 1237, Springer, Berlin, 1987 315–330.

[2] G. S. Ammar and W. B. Gragg, Numerical experience with a superfast real Toeplitz solver, Linear Algebra Appl. 121 (1989) 185–206.

[3] 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) 241–247.

[4] Å. Björck, Numerical Methods for Least Squares Problems, SIAM, Philadelphia, 1996.

[5] F. A. Bornemann and P. Deuflhard, The cascadic multigrid method for elliptic problems, Numer. Math. 75 (1996) 135–152.

[6] Z. Chen, Y. Xu, and H. Yang, A multilevel augmentation method for solving ill-posed operator equations, Inverse Problems 22 (2006) 155–174.

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

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

[9] J. W. Evans, W. B. Gragg, and R. J. LeVeque, On the least squares exponential sum approximation, Math. Comp. 34 (1980) 203–211.

[10] W. B. Gragg and L. Reichel, On the application of orthogonal polynomials to the iterative solution of linear systems of equations with indefinite or non-Hermitian matrices, Linear Algebra Appl. 88–89 (1987) 349–371.

[11] C. W. Groetsch, The Theory of Tikhonov Regularization for Fredholm Equations of the First Kind, Pitman, London, 1984.

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

[13] M. Hanke and C. R. Vogel, Two-level preconditioners for regularized inverse problems I: Theory, Numer. Math. 83 (1999) 385–402.

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

[15] T. Huckle and J. Staudacher, Multigrid preconditioning and Toeplitz matrices, Electron. Trans. Numer. Anal. 13 (2002) 81–105.

[16] M. Jacobsen, P. C. Hansen, and M. A. Saunders, Subspace preconditioned LSQR for discrete ill-posed problems, BIT 43 (2003) 975–989.

[17] J. T. King, Multilevel algorithms for ill-posed problems, Numer. Math. 61 (1992) 311–334.

[18] A. S. Nemirovskii, The regularization properties of the adjoint gradient method in ill-posed problems, USSR Comput. Math. Math. Phys. 26(2) (1986) 7–16.

[19] C. C. Paige and M. A. Saunders, Solution of sparse indefinite systems of linear equations, SIAM J. Numer. Anal. 12 (1975) 617–629.

[20] D. L. Phillips, A technique for the numerical solution of certain integral equations of the first kind, J. ACM 9 (1962) 84–97.

[21] R. Plato, Optimal algorithms for linear ill-posed problems yield regularization methods, Numer. Funct. Anal. Optim. 11 (1990) 111–118.

[22] O. Scherzer, An iterative multi level algorithm for solving nonlinear ill-posed problems, Numer. Math. 80 (1998) 579–600.

[23] U. Trottenberg, C. W. Osterlee, and A. Schüler, Multigrid, Academic Press, Orlando, 2001.