Range-relaxed criteria for the choosing Lagrange multipliers in nonstationary iterated Tikhonov method

R. Boiger[†]   A. Leitão[‡]   B. F. Svaiter[§]

January 23, 2018

Abstract

In this article we propose a novel *nonstationary iterated Tikhonov* (NIT) type method for obtaining stable approximate solutions to ill-posed operator equations modeled by linear operators acting between Hilbert spaces. Geometrical properties of the problem are used to derive a new strategy for choosing the sequence of regularization parameters (Lagrange multipliers) for the NIT iteration. Convergence analysis for this new method is provided. Numerical experiments are presented for two distinct applications: I) A 2D elliptic parameter identification problem (Inverse Potential Problem); II) An image deblurring problem. The results obtained validate the efficiency of our method compared with standard implementations of the NIT method (where a geometrical choice is typically used for the sequence of Lagrange multipliers).

Keywords. Ill-posed problems; Linear operators; Iterated Tikhonov method; Nonstationary methods.

AMS Classification: 65J20, 47J06.

1 Introduction

In this article we propose a new *nonstationary Iterated Tikhonov* (NIT) type method [5, Sec. 1.2] for obtaining stable approximations of linear ill-posed problems. The Lagrange multiplier is chosen so as to guarantee the residual of the next iterate to be in a *range*. Previous strategies for choosing the Lagrange multiplier in each iteration of NIT type methods either prescribe (*a priori*) a geometrical increase of this multiplier [17] or require (*a posteriori*) the residual at the next iterate to assume a prescribed value which depends on the current residual.

In those NIT methods that prescribe a geometrical increase of the Lagrange multipliers, the use of a too large geometric factor may lead to numerical instabilities and failure of convergence, whereas the use of a too small factor leads to a slow convergent method (see Figures 1 and 2); these features are highly dependent on the problem at hand and, in general, it is not clear how to adequately choose the geometric factor.

[†]Insitut für Mathematik, Alpen-Adria Universität Klagenfurt, Universitätsstrasse 65-67, 9020 Klagenfurt, Austria, `romana.boiger@aau.at`.

[‡]Department of Mathematics, Federal University of St. Catarina, P.O. Box 476, 88040-900 Florianópolis, Brazil, `acgleitao@gmail.com`.

[§]IMPA, Estr. Dona Castorina 110, 22460-320 Rio de Janeiro, Brazil, `benar@impa.br`.

In those NIT methods that require the residual at the next iterate to assume a prescribed value, at each iteration one needs to solve a nonlinear equation which involves the resolvent of an ill-posed operator [16, 8]. This is accomplished by means of iterative methods (*e.g.* Newton) which require, at each of their steps, the solution of a linear system for a different operator. Consequently, only the number of iterations required for these methods does not fully quantify their computational costs.

The main contribution in this article is the proposal of a novel *a posteriori* strategy for choosing the Lagrange multipliers in NIT methods. Since we prescribe the residual of the next iterate to be in a range, the set of feasible Lagrange multipliers, at each iteration, is a *non-degenerate interval*, which renders feasible their economical computation (as explained later on). Many relevant theoretical convergence properties present at previous *a posteriori* methods [12] (e.g., residual convergence rates, stability, semi-convergence) still hold for our novel strategy. We also explore the feature of a feasible interval for the Lagrange multiplier to speed up a Newton-like method for computing it. The resulting method proves, in our preliminary numerical experiments, to be more efficient than the geometrical choice of the Lagrange multipliers [17], typically used in implementations of NIT type methods, for low noise levels.

The *inverse problem* we are interested in consists of determining an unknown quantity $x \in X$ from the set of data $y \in Y$, where X, Y are Hilbert spaces. In practical situations, one does not know the data exactly; instead, only approximate measured data $y^\delta \in Y$ are available with

$$\|y^\delta - y\|_Y \ \leq \ \delta \,, \tag{1}$$

where $\delta > 0$ is the (known) noise level. The available data y^δ are obtained by indirect measurements of the parameter x, this process being described by the ill-posed operator equation

$$A\,x \ = \ y \,, \tag{2}$$

where $A : X \to Y$ is a bounded linear operator, whose inverse $A^{-1} : R(A) \to X$ either does not exist, or is not continuous. Consequently, approximate solutions are extremely sensitive to noise in the data.

Linear ill-posed problems are commonly found in applications ranging from image analysis to parameter identification in mathematical models. There is a vast literature on iterative methods for the stable solution of (2). We refer the reader to the books [15, 19, 2, 25, 26, 1, 23, 12, 27, 21] and the references therein. Iterated Tikhonov (IT) type methods for solving the ill-posed problem (2) are defined by iteration formula

$$x_k^\delta \ = \ \arg\min_{x \in X} \left\{ \lambda_k \|Ax - y^\delta\|^2 + \|x - x_{k-1}^\delta\|^2 \right\} ,$$

what corresponds to

$$\begin{aligned} x_k^\delta &= x_{k-1}^\delta - (I + \lambda_k A^* A)^{-1} \lambda_k A^* (Ax_{k-1}^\delta - y^\delta) \,, \\ &= (\lambda_k^{-1} I + A^* A)^{-1} \left[\lambda_k^{-1} x_{k-1}^\delta + A^* y^\delta \right] \end{aligned} \tag{3}$$

where $A^* : Y \to X$ is the adjoint operator to A. The parameter $\lambda_k > 0$ can be viewed as the Lagrange multiplier of the problem of projecting x_{k-1}^δ onto a levelset of $\|Ax - y^\delta\|^2$. If the sequence $\{\lambda_k = \lambda\}$ is constant, iteration (3) is called *stationary* IT (or SIT), otherwise it is denominated *nonstationary* IT (or NIT).

In the NIT methods, each λ_k is either chosen *a priori* (e.g., in geometric progression) or it is chosen *a posteriori*. In the *a posteriori* variants, λ_k is chosen so that the next iterate has a

prescribed residual which is either a fixed fraction of the current residual or a fraction which depends also on the noise level [8, eq. 2.11b in Alg. 1]. In other words, λ_k is chosen in (3) so that

$$\|Ax_k^\delta - y^\delta\| = \Phi(\|Ax_{k-1}^\delta - y^\delta\|, \delta) \tag{4}$$

(see [9] for yet another strategy). We propose λ_k to be choose so that

$$\delta \leq \|Ax_k^\delta - y^\delta\| \leq \Psi(\|Ax_{k-1}^\delta - y^\delta\|, \delta). \tag{5}$$

Our upper bound for the residual also depends on the current residual and the noise level (as in [8]), although we use a different formula to choose it (see eq. (8)).

The SIT method for solving (2) was considered in [25, 15], where a well developed convergence analysis can be found (see also [24], were Lardy analyzed the particular choice $\lambda_k = 1$). It is worth distinguishing between the SIT method and the *iterated Tikhonov methods of order n* [22, 11, 28], where the number of iterations (namely n) is fixed. In this case $\lambda_k = \lambda > 0$, $k = 0, \ldots, n-1$, and λ plays the role of the regularization parameter.

The NIT method was addressed by many authors, e.g. [13, 17, 5]. In numerical implementations of this method, the geometrical choice $\lambda_k = q^k$, $q > 1$, is a commonly adopted strategy and we shall refer to the resulting method as gNIT method (*geometrical nonstationary IT method*).

The numerical performances of NIT type methods are superior to the ones of SIT type methods. This fact is illustrated by Example 1.1.

Example 1.1. *A linear system modeled by the Hilbert matrix $H^{25 \times 25}$ is considered in Figure 1 (random noise of level $\delta = 0.001\%$ is used). In this benchmark problem the SIT method is implemented for $\lambda_k = 2$ (RED), while the gNIT method is implemented for $\lambda_k = 2^k$ (BLACK) and $\lambda_k = 3^k$ (BLUE).*

The computation $\lambda_k = q^k$ in the gNIT is straightforward; however, the choice of $q > 1$ is *exogenous* to (2), (1) and it is not clear which are the good values for q. Indeed, as shown in the next example, increasing the constant q may lead either to faster convergence or failure to converge:

Example 1.2. *We set $A = H^{25 \times 25}$, $X = Y = \mathbb{R}^{25}$ and random noisy data with $\delta = 10^{-5}\%$. In Figure 2 the gNIT method is implemented for $\lambda_k = 2^k$ (BLACK), $\lambda_k = 3^k$ (BLUE) and $\lambda_k = 4^k$ (ORANGE); the last implementation does not converge.*

The above described issues motivated the use of *a posteriori* choices for the Lagrange multipliers, which requires the residual at the next iterate to assume a prescribed value dependent on the current residual and also on the noise level.

Next we briefly review some relevant convergence results for IT type methods.

- In [5] Brill and Schock proved that, in the exact data case, the NIT method (3) converges to a solution of $Ax = y$ if and only if $\sum \lambda_k = \infty$. Moreover, a convergence rate result was established under the additional assumption $\sum \lambda_k^2 < \infty$.
- The assumptions needed in [5] in order to derive convergence rates are neither satisfied for the sIT, nor for the NIT with the geometrical choice $\lambda_k = q^k$, $q > 1$.
- In [17] rates of convergence are established for the stationary Lardy's method [24] as well as for the NIT with geometrical choice of λ_k. Under the source condition $x^\dagger \in Rg((A^*A)^\nu)$,
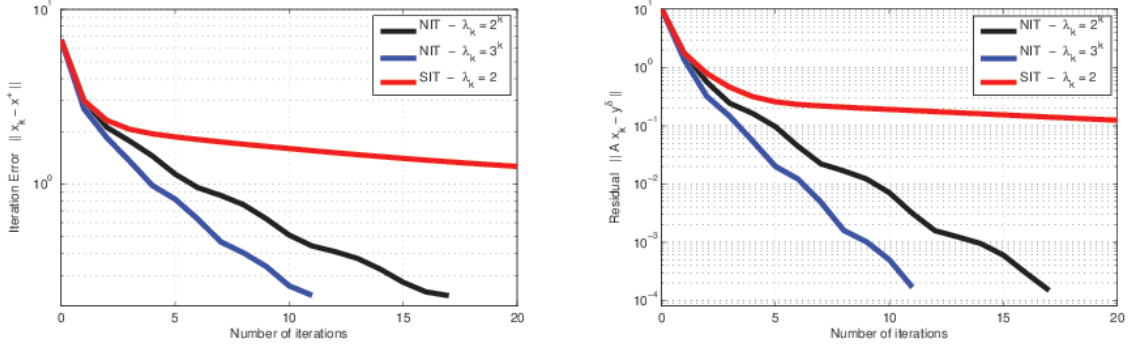
Figure 1: Comparison between SIT and NIT type methods: $A = H^{25 \times 25}$ is the square Hilbert matrix; artificial random noise ($\delta = 10^{-3}\%$) is added to the data. The stopping criteria (discrepancy principle with $\tau = 2$) is reached after 18 steps (BLACK), 12 steps (BLUE) and 83737 steps (RED).
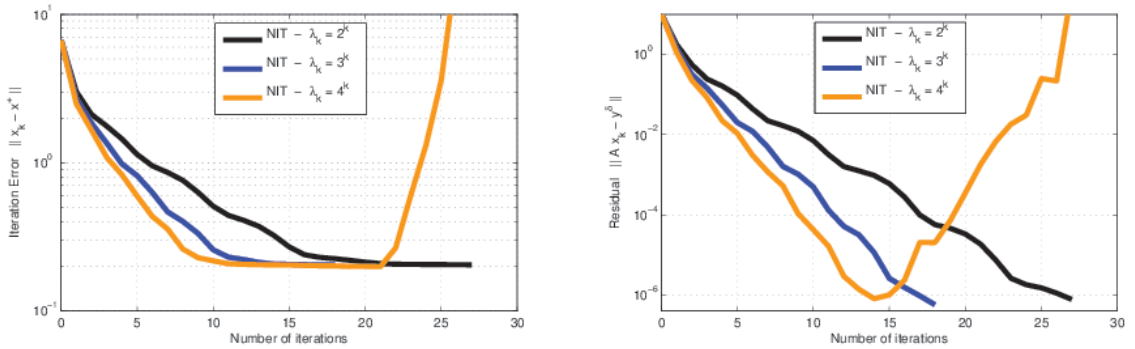

Figure 2: Unstable behavior of gNIT type methods: matrix $A \in \mathbb{R}^{25 \times 25}$ as in Figure 1; artificial random noise ($\delta = 10^{-5}\%$) is added to the data. The stopping criteria (discrepancy principle with $\tau = 2$) is reached after 27 steps (BLACK), 18 steps (BLUE). In the last implementation (ORANGE), the gNIT method becomes unstable before the stopping criteria is reached.

where $x^\dagger = A^\dagger y$ is the normal solution of $Ax = y$[1] and $\nu > 0$, [2] the linear rate of convergence $\|x_k - x^\dagger\| = O(q^{k\nu})$ is proven.

The article is organized as follows: In Section 2 we introduce the new method (rrNIT), which is proposed and analyzed in this manuscript. Moreover, a detailed formulation of this method is given, and some preliminary estimates are also derived. In Section 3 a convergence analysis of the rrNIT method is presented. In Section 4 we discuss the algorithmic implementation of the rrNIT method. In particular, we address the challenging numerical issue of efficiently computing the Lagrange multipliers λ_k. Section 5 is devoted to numerical experiments. The *Image deblurring* problem and the *Inverse Potential Problem* are considered in Subsections 5.1 and 5.2 respectively. Section 6 is dedicated to final remarks and conclusions.

---

[1]I.e., $x^\dagger$ is the unique vector satisfying $x^\dagger \in D(A) \cap N(A)^\perp$ and $Ax^\dagger = y$.
[2]See [17, Theorem 2.1] for details on the positive the scalar ν.

2 Range-relaxed non-stationary iterated Tikhonov method

In this section first we propose in Subsection 2.1 a conceptual projection method for solving (1), (2). In this method, each iterate is obtained projecting the previous one onto a level set of the residual function. The level set is prescribed to belong to a range of level sets, instead of being a single one. Second, we propose in Subsection 2.2 an implementable version of the conceptual method where the projection is computed via Lagrange multipliers. Finally, we derive some basic properties of the new proposed method.

The implementable method proposed here happens to be a new NIT method where, in each iteration, the set of feasible choices for the Lagrange multipliers is an interval, instead of a single real number. For this reason, we call the new method a range-relaxed Non-stationary Iterative Tikhonov Method (rrNIT).

For the remaining of this article we suppose that the following assumptions hold true:

(A1) There exist $x^\star \in X$ such that $Ax^\star = y$, where $y \in Rg(A)$ are the exact data.

(A2) The operator $A : X \to Y$ is linear, bounded and ill-posed, i.e., even if the operator $A^{-1} : R(A) \to X$ (the left inverse of A) exists, it is not continuous.

2.1 A Successive Orthogonal range-relaxed Projections Method

We use the notation Ω_μ, for $\mu \geq 0$, to denote the μ-levelset of the residual functional $\|Ax - y^\delta\|$, that is,

$$\Omega_\mu := \{x \in X : \|Ax - y^\delta\|^2 \leq \mu^2\}. \tag{6}$$

The basic geometric properties of the levelsets Ω_μ, described next, are instrumental in the forthcoming analysis.

Proposition 2.1. *Let Ω_μ be as in (6).*

1. *For each $\mu \geq 0$, the set Ω_μ is closed and convex.*

2. *If $\mu' \geq \mu > 0$ then $\Omega_\mu \subset \Omega_{\mu'}$.*

3. *If $\mu \geq \delta$ then $A^{-1}(y) \subset \Omega_\mu$.*

4. *If $\mu > \delta$ then Ω_μ has non-empty interior.*

Notice that all available information about the solution set $A^{-1}(y)$ is contained in (1), (2). Thus, in the absence of additional information, Ω_δ is the set of best possible approximate solutions for the inverse problem under consideration.[3]

Nevertheless the levelset Ω_δ is, in general, unbounded and it is desirable to exclude those approximate solutions with "too large" norms. Moreover, very often a crude estimation $\hat{x}$ to the solution of (2) is available. In this context, it is natural to consider the projection problem

$$\begin{cases} \min_x & \|x - \hat{x}\|^2 \\ \text{s.t.} & \|Ax - y^\delta\|^2 \leq \mu^2. \end{cases} \tag{7}$$

where $\mu \geq 0$. Observe that if $\|A\hat{x} - y^\delta\| > \mu \geq \delta$, then the solution of this projection problem is closer to Ω_δ than $\hat{x}$ and has a smaller residual than $\hat{x}$.

---

[3] I.e., given two elements in Ω_δ, it is not possible to distinguish which of them better approximates $x^\star$.

The considerations in the preceding paragraph show that it is possible, at least conceptually, to devise projection methods for solving our ill-posed problem. Let us briefly discuss the conditioning problem (7) with respect to parameter μ:

(i) for $0 \leq \mu < \delta$ problem (7) may be unfeasible, that is, it may become the problem of projecting $\hat{x}$ onto an *empty set*;

(ii) for $\mu = \delta$, in view of (i), problem (7) is in general ill-posed with respect to the parameter μ;

(iii) for $\mu > \delta$ problem (7) is well posed, and it is natural to expect that the larger the μ the better conditioned it becomes.

A compromise between reducing the residual norm $\|Ax - y^\delta\|$ and preventing ill-posedness of the projection problem would be to choose

$$\hat{\mu} \;=\; p\|A\hat{x} - y^\delta\| + (1 - p)\delta \,,$$

where $0 < p < 1$ quantify this compromise. However,

1. projecting $\hat{x}$ onto $\Omega_{\hat{\mu}}$, which is a *pre-defined* level set of $\|Ax - y^\delta\|^2$, entails an additional numerical difficulty: the projection has to be computed by solving a linear system where the Lagrange multiplier is implicitly defined by an *algebraic equation*;

2. the projection of $\hat{x}$ onto any levelset Ω_μ with $\delta \leq \mu \leq \hat{\mu}$ is as good as (or even better than) the projection of $\hat{x}$ onto $\Omega_{\hat{\mu}}$.

In view of these observations, we shall generate x_k^δ from $\hat{x} = x_{k-1}^\delta$ by projecting it onto any one of the range of convex sets $(\Omega_\mu)_{\delta \leq \mu \leq \hat{\mu}}$, that is, by solving the *range-relaxed* projection problem of computing (x, μ) such that

$$\begin{cases} \min_x & \|x - \hat{x}\|^2 \\ \text{s.t.} & \|Ax - y^\delta\|^2 \leq \mu^2 \,, \qquad \delta \leq \mu \leq p\|A\hat{x} - y^\delta\| + (1 - p)\delta \,, \end{cases} \tag{8}$$

whenever $x_{k-1}^\delta \notin \Omega_\delta$. Observe that this problem has multiple solutions. The advantage of this strategy is that the set of feasible Lagrange multipliers of the above problem is an interval with non-empty interior, as we will discuss latter, instead of a single point.

In what follows we use the notation $\mathbf{P}_\Omega$ to denote the orthogonal projection onto Ω, for $\emptyset \neq \Omega \subset X$ closed and convex. The discussion in the previous paragraph leads us to propose the *conceptual* successive orthogonal range-relaxed projection Method for problem (1), (2) described in Algorithm 1.

---

[1] choose an initial guess $x_0 \in X$;

[2] choose $p \in (0, 1)$ and $\tau > 1$;

[3] for $k \geq 1$ do

    [3.1] compute (x_k^δ, μ_k),

        $x_k^\delta = \mathbf{P}_{\Omega_{\mu_k}}(x_{k-1}^\delta)$, $\delta \leq \mu_k \leq p\|Ax_{k-1}^\delta - y^\delta\| + (1 - p)\delta$;

    [3.2] stop to iterate at step $k^* \geq 1$ s.t. $\|Ax_{k^*}^\delta - y^\delta\| < \tau\delta$ for the first time.

Algorithm 1: Successive orthogonal range-relaxed projection method.

Since $\|Ax_k^\delta - y^\delta\| = \mu_k$, the variable μ_k is redundant, nevertheless, its use in the conceptual Algorithm 1 clarify the kind of projection problem used to compute x_k^δ.

By its definition, Algorithm 1 is a method of successive orthogonal projections onto level sets of $\|Ax - y^\delta\|$. As the level set used in each iteration shall be in a range, we denominate it a successive orthogonal *range-relaxed* projection method. Each iterate is obtained from the previous one by projecting it onto a convex set that contains the solution set and in which the residual in any point is strictly smaller that the residual at the previous iterate. Therefore the errors as well as the residuals are strictly decreasing along the iterates and the sequence of iterates is bounded.

2.2 A Range-relaxed non-stationary iterated Tikhonov algorithm

In order to derive an implementable version of the conceptual method (Algorithm 1) discussed in the previous section, we need to specify how to compute the range-relaxed projections (at Step [3]). With this aim, recall that the canonical Lagrangian of problem (7) is

$$\mathcal{L}(x, \lambda) = \frac{\lambda}{2}(\|Ax - y^\delta\|^2 - \mu^2) + \frac{1}{2}\|x - \hat{x}\|^2. \tag{9}$$

For each $\lambda > 0$, $\mathcal{L}(\cdot, \lambda) : X \to \mathbb{R}$ has a unique minimizer x' which is also characterized as the unique solution of $\nabla_x \mathcal{L}(x, \lambda) = 0$, that is,

$$x' = \hat{x} - \lambda(I + \lambda A^*A)^{-1}A^*(A\hat{x} - y^\delta).$$

The next lemma summarizes the solution theory for the projection problem (7) by means of Lagrange multiplier [29, Sec. 5.7]. Recall that $\mathbf{P}_\Omega$ denotes the orthogonal projection onto Ω.

Lemma 2.2. *Suppose* $\|A\hat{x} - y^\delta\| > \mu > \delta$. *The following assertions are equivalent*

1. $x' = \mathbf{P}_{\Omega_\mu}(\hat{x})$;

2. x' *is the* *solution of* (7);

3. $x' = \hat{x} - \lambda^*(I + \lambda^*A^*A)^{-1}A^*(A\hat{x} - y^\delta)$, $\lambda^* > 0$ *and*

$$\|Ax' - y^\delta\| = \mu$$

Proof. Equivalence between items 1 and 2 follows from the definition of orthogonal projections onto closed convex sets. Equivalence between items 2 and 3 is a classical Lagrange multipliers result (see, e.g., [29, Theorem 5.15]). $\qquad\square$

In the next lemma we address the range-relaxed projection problem (8); its proof follows from Lemma 2.2.

Lemma 2.3. *Suppose* $\|A\hat{x} - y^\delta\| > \delta$ *and* $0 < p < 1$. *The following assertions are equivalent*

1. $x' = \mathbf{P}_{\Omega_\mu}(\hat{x})$ *and* $\delta \le \mu \le p\|A\hat{x} - y^\delta\| + (1 - p)\delta$;

2. $(x', \mu) \in X \times \mathbb{R}$ *is a solution of* (8);

3. $x' = \hat{x} - \lambda(I + \lambda A^*A)^{-1}A^*(A\hat{x} - y^\delta)$, $\lambda > 0$,

$$\delta \le \|Ax' - y^\delta\| \le p\|A\hat{x} - y^\delta\| + (1 - p)\delta,$$

and $\mu = \|Ax' - y^\delta\|$.

It follows from Lemma 2.3 that solving the range-relaxed projection problem in (8) boils down to solving the inequalities

$$\delta \leq \|Ax' - y^\delta\| \leq p\|A\hat{x} - y^\delta\| + (1-p)\delta,$$
$$\text{where } x' = \hat{x} - \lambda(I + \lambda A^*A)^{-1}A^*(A\hat{x} - y^\delta) \tag{10}$$

and defining $x = x'$ and $\mu = \|Ax' - y^\delta\|$. We use this result to propose an implementable version of Algorithm 1 as follows:

---

[1] choose an initial guess $x_0 \in X$;

[2] choose $p \in (0,1)$, $\tau > 1$ and set $k := 0$;

[3] while $\left(\|Ax_k^\delta - y^\delta\| > \tau\delta\right)$ do

    [3.1] $k := k + 1$;

    [3.2] compute λ_k and x_k^δ such that
$$x_k^\delta = x_{k-1}^\delta - \lambda_k\left(I + \lambda_k A^*A\right)^{-1}A^*(Ax_{k-1}^\delta - y^\delta),$$
$$\delta \leq \|Ax_k^\delta - y^\delta\| \leq p\|Ax_{k-1}^\delta - y^\delta\| + (1-p)\delta$$

Algorithm 2: The rrNIT method.

As in Algorithm 1, the stopping index in the above algorithm is defined by the discrepancy principle

$$k^* := \min\{k \geq 1\,;\; \|Ax_j - y^\delta\| > \tau\delta,\; j = 0, \ldots, k-1 \quad \text{and} \quad \|Ax_k - y^\delta\| \leq \tau\delta\}. \tag{11}$$

The computational burden of the above algorithm resides in the computation of step [3.2], which requires the solution of a linear system whose corresponding residual shall be in a given range. In other words, λ_k shall be a solution of (10) with $\hat{x} = x_{k-1}^\delta$.

The next lemma provides a lower bound for $\lambda = \lambda_k$ which will be used in the convergence analysis of Algorithm 2, presented in Section 3. This lower bound can also be used for used as initial guess to compute λ_k.

Lemma 2.4. *Under the assumptions of Lemma 2.2,*

$$\lambda^* \geq \frac{(\|A\hat{x} - y^\delta\| - \mu)\|A\hat{x} - y^\delta\|}{\|A^*(A\hat{x} - y^\delta)\|^2}.$$

Proof. To simplify the notation, let

$$z := \mathbf{P}_{\Omega_\mu}\hat{x}, \qquad b := A\hat{x} - y^\delta.$$

From the assumption $\|A\hat{x} - y^\delta\| > \mu > \delta$, it follows that $\hat{x} \notin \Omega_\delta$. Therefore $\|Az - y^\delta\| = \mu$,

$$\|A(z - \hat{x})\| \geq \|A\hat{x} - y^\delta\| - \|Az - y^\delta\| = \|b\| - \mu, \tag{12}$$

and

$$\mu^2 = \|A(z - \hat{x}) + b\|^2 = \|A(z - \hat{x})\|^2 + \|b\|^2 + 2\langle A(z - \hat{x}), b\rangle.$$

Direct combination of the above equation with the previous inequality yields

$$\begin{aligned}
-2\langle A(z - \hat{x}), b\rangle &= \|A(z - \hat{x})\|^2 + \|b\|^2 - \mu^2 \\
&\geq \left(\|b\| - \mu\right)^2 + \|b\|^2 - \mu^2 \\
&= 2\|b\|\left(\|b\| - \mu\right).
\end{aligned}$$

Therefore, if follows from (12), Cauchy-Schwartz inequality, and the definition of $\mathbf{P}_{\Omega_\mu}$ that

$$
\begin{aligned}
\left(\|b\| - \mu \right) \|b\| & \leq \; \left\langle -(z - \hat{x}), \, A^* b \right\rangle \\
& = \; \left\langle \lambda^* (I + \lambda^* A^* A)^{-1} A^* b, \, A^* b \right\rangle \\
& \leq \; \lambda^* \, \|(I + \lambda^* A^* A)^{-1}\| \, \|A^* b\|^2 \\
& \leq \; \lambda^* \, \|A^* b\|^2 \,,
\end{aligned}
$$

proving the lemma. $\qquad\square$

Corollary 2.5. *Let the sequences (x_k^δ) and (λ_k) be defined by the rrNIT method (Algorithm 2), with $\delta \geq 0$ and $y^\delta \in Y$ as in (1) and let $\mu_k := \|A x_k^\delta - y^\delta\|$. Then*

$$
\lambda_k \; \geq \; \frac{\left(\|A x_{k-1}^\delta - y^\delta\| - \mu_k \right) \|A x_{k-1}^\delta - y^\delta\|}{\|A^* (A x_{k-1}^\delta - y^\delta)\|^2} \,, \quad k = 1, \dots, k^*. \tag{13}
$$

In the exact data case (i.e., $\delta = 0$) the above estimate simplifies to $\lambda_k \geq (1 - p) \|A\|^{-2}$.

Proof. From (10) and the definition of μ_k it follows that $\delta < \mu_k < \|A x_{k-1}^\delta - y^\delta\|$. Thus, (13) follows from Lemma 2.4 with $\hat{x} = x_{k-1}^\delta$, $\lambda^* = \lambda_k$ and $\mu = \mu_k$ (in the proof of that lemma it holds $z = x_k^\delta$).

In the exact data case, it follows from (13), together with Assumption **(A2)**, that $\lambda_k \geq \left(\|A x_{k-1} - y\| - \mu_k \right) \|A\|^{-2} \|A x_{k-1} - y\|^{-1}$. Moreover, since $\delta = 0$ we have $\mu_k \leq p \|A x_{k-1} - y\|$. Combining these two facts, the second assertion follows. $\qquad\square$

3   Convergence Analysis

We begin this section by establishing an estimate for the decay of the residual $\|A x_k^\delta - y^\delta\|$.

Proposition 3.1. *Let (x_k^δ) be the sequence defined by the rrNIT method (Algorithm 2), with $\delta \geq 0$ and $y^\delta \in Y$ as in (1). Then*

$$
\left[\|A x_k^\delta - y^\delta\| - \delta \right] \; \leq \; p \left[\|A x_{k-1}^\delta - y^\delta\| - \delta \right] \; \leq \; p^k \left[\|A x_0 - y^\delta\| - \delta \right], \quad k = 1, \dots, k^* \,,
$$

where $k^* \in \mathbb{N}$ is defined by (11).

Proof. It is enough to verify the first inequality. Recall that $x_k^\delta \in \Omega_{\mu_k}$, where $\delta \leq \mu_k \leq p \|A x_{k-1}^\delta - y^\delta\| + (1 - p)\delta$. Consequently, $\|A x_k^\delta - y^\delta\| \leq p \|A x_{k-1}^\delta - y^\delta\| + (1 - p)\delta$, and the first inequality follows. $\qquad\square$

Now we are ready to prove finiteness of k^* and to provide an upper bound for it, whenever $\delta > 0$.

Corollary 3.2. *Let (x_k^δ) be the sequence defined by the rrNIT method (Algorithm 2), with $\delta > 0$ and $y^\delta \in Y$ as in (1). Then the stopping index k^* defined in (11) satisfies*

$$
k^* \; \leq \; |\ln p|^{-1} \ln \left[\frac{\|A x_0 - y^\delta\| - \delta}{(\tau - 1)\delta} \right] + 1 \,.
$$

Proof. We may assume $\|Ax_0 - y^\delta\| > \tau\delta$.[4] From (11) follows $\tau\delta < \|Ax_k^\delta - y^\delta\|$, $k = 0, \ldots k^* - 1$. This inequality (for $k = k^* - 1$), together with Proposition 3.1 imply that

$$(\tau - 1)\delta \ < \ \|Ax_{k^*-1}^\delta - y^\delta\| - \delta \ \le \ p^{k^*-1}\left[\|Ax_0 - y^\delta\| - \delta\right],$$

completing the proof (recall that $p \in (0, 1)$). $\qquad\square$

Monotonicity of the iteration error $\|x^\star - x_k^\delta\|$ was already established in Section 2. In the next proposition we estimate the "gain" $\|x^\star - x_{k-1}^\delta\|^2 - \|x^\star - x_k^\delta\|^2$ in the rrNIT method.

Proposition 3.3. *Let (x_k^δ) be the sequence defined by the rrNIT method (Algorithm 2), with $\delta > 0$ and $y^\delta \in Y$ as in (1). Then*

$$\|x^\star - x_{k-1}^\delta\|^2 - \|x^\star - x_k^\delta\|^2 \ = \ \|x_k^\delta - x_{k-1}^\delta\|^2 + \lambda_k\|A(x^\star - x_k^\delta)\|^2 + \lambda_k\left[r(x_k^\delta) - r(x^\star)\right], \ (14)$$

for $k = 1, \ldots, k^*$, where $r(x) := \|Ax - y^\delta\|^2$. Consequently,

$$\|x^\star - x_{k-1}^\delta\|^2 - \|x^\star - x_k^\delta\|^2 \ \ge \ \lambda_k^2 \|A^*(Ax_k^\delta - y^\delta)\|^2 + \lambda_k\left[\|A(x^\star - x_k^\delta)\|^2 + (\tau^2 - 1)\,\delta^2\right], \ (15)$$

for $k = 1, \ldots, k^* - 1$; moreover,

$$\|x^\star - x_{k^*-1}^\delta\|^2 - \|x^\star - x_{k^*}^\delta\|^2 \ \ge \ \lambda_{k^*}^2 \|A^*(Ax_{k^*}^\delta - y^\delta)\|^2 + \lambda_{k^*} \|A(x^\star - x_{k^*}^\delta)\|^2. \qquad (16)$$

Proof. First we derive (14). Due to the definition of $(x_k^\delta, \lambda^\delta)$ in Algorithm 2, the Lagrangian $\mathcal{L}$ in (9) (with $\mu = \mu_k$ and $\hat{x} = x_{k-1}$) satisfies

$$\mathcal{L}(x, \lambda_k) \ = \ \mathcal{L}(x_k^\delta, \lambda_k) + \mathcal{L}_x(x_k^\delta, \lambda_k)(x - x_k^\delta) + \frac{1}{2}\left\langle (x - x_k^\delta), H(\lambda_k)(x - x_k^\delta)\right\rangle,$$

where $H(\lambda_k) = (I + \lambda_k A^*A)$ is the Hessian of $\mathcal{L}(\cdot, \lambda_k)$ at $x = x_k^\delta$. Since $\mathcal{L}_x(x_k^\delta, \lambda_k) = 0$, we have

$$\mathcal{L}(x, \lambda_k) = \mathcal{L}(x_k^\delta, \lambda_k) + \frac{1}{2}\left\langle (x - x_k^\delta), (I + \lambda_k A^*A)(x - x_k^\delta)\right\rangle,$$

that is,

$$\|x - x_{k-1}^\delta\|^2 + \lambda_k\left[r(x) - \mu_k^2\right] = \|x_k^\delta - x_{k-1}^\delta\|^2 + \lambda_k\left[r(x_k^\delta) - \mu_k^2\right] + \|x - x_k^\delta\|^2 + \lambda_k\|A(x - x_k^\delta)\|^2.$$

Now, choosing $x = x^\star$, one establishes (14). Inequality (15), on the other hand, follows from (14) together with $r(x^\star) \le \delta^2$ and $r(x_k^\delta) > \tau^2\delta^2$, for $k = 1, \ldots, k^* - 1$. Analogously, (16) follows from (14) together with $r(x_{k^*}^\delta) > \delta^2$ (see Algorithm 2). $\qquad\square$

Corollary 3.4. *In the exact data case, i.e., $\delta = 0$ and $y^\delta = y \in R(A)$, then (14) becomes*

$$\|x^\star - x_{k-1}\|^2 - \|x^\star - x_k\|^2 \ = \ \|x_k - x_{k-1}\|^2 + 2\lambda_k \|Ax_k - y\|^2,$$

from which follows $\sum_{k\ge 1} \lambda_k \|Ax_k - y\|^2 \ < \ \infty$ and $\sum_{k\ge 1} \|x_k - x_{k-1}\|^2 \ < \ \infty$ (the last inequality means that the operator describing the rrNIT iteration is a reasonable wanderer in the sense of [6]).

Next we prove strong convergence of the rrNIT method (in the exact data case) to a solution of the inverse problem (2). The estimate in Lemma 2.4 plays a key role in this proof.

Theorem 3.5. *Let (x_k) and (λ_k) be the sequences defined by the rrNIT method (Algorithm 2), with $\delta = 0$ and $y^\delta = y \in R(A)$. Then (x_k) converges strongly to some $x^* \in X$. Moreover, $Ax^* = y$.*

Proof. From the second assertion in Corollary 2.5 it follows that $\sum_{k\ge 1} \lambda_k = \infty$. The proof now follows from [5, Theor. 1.4]. $\qquad\square$

---

[4]Otherwise the iteration does not start, i.e., $k^* = 0$.

4 Numerical Implementation

In this section the implementation of Algorithm 2 is reviewed. We discuss the implementation of step [3.2] of that algorithm by means of a Newton-like method, and how we accelerated this computation.

As discussed in Section 2.2, at step [3] of Algorithm 2, $\lambda_k \geq 0$ is to be obtained as a solution of the scalar rational inequalities (or a inclusion) (10) with $\hat{x} = x_{k-1}^\delta$, that is,

$$
\delta \leq \|A_k x_k^\delta - y^\delta\| \leq p\|A x_{k-1}^\delta - y^\delta\| + (1-p)\delta, \tag{17}
$$
$$
\text{where } x_k^\delta = x_{k-1}^\delta - \lambda(I + \lambda A^* A)^{-1} A^* (A x_k^\delta - y^\delta) \text{ with } \lambda > 0.
$$

Define, at iteration k, $\pi_k(\lambda)$ as the candidate for x_k^δ obtained from x_{k-1}^δ with the Lagrange multiplier λ and let $G_k(\lambda)$ be the square residual at that point, that is,

$$
\pi_k(\lambda) := x_{k-1}^\delta - \lambda(I + \lambda A^* A)^{-1} A^* (A x_{k-1}^\delta - y^\delta), \tag{18a}
$$
$$
:= (\lambda^{-1} I + A^* A)^{-1} (\lambda^{-1} x_{k-1} + A^* y^\delta), \tag{18b}
$$
$$
G_k(\lambda) := \|A \pi_k(\lambda) - y^\delta\|^2. \tag{18c}
$$

With this notation, (17) writes

$$
\delta^2 \leq G_k(\lambda) \leq (p\|A x_{k-1}^\delta - y^\delta\| + (1-p)\delta)^2 \quad \text{with } \lambda > 0. \tag{19}
$$

As earlier mentioned, the set of feasible Lagrange multipliers is a non-degenerate interval in each iteration of the rrNIT method.

Proposition 4.1. *Suppose Algorithm 2 reaches iteration k and $\|A x_k^\delta - y^\delta\| > \tau\delta$. Then, for $\lambda \geq 0$*

$$
\frac{d}{d\lambda} G_k(\lambda) = -2 \left\langle A^* (A \pi_k(\lambda) - y^\delta), (I + \lambda A^* A)^{-1} A^* (A \pi_k(\lambda) - y^\delta) \right\rangle
$$
$$
= -2\lambda^{-3} \left\langle x_{k-1} - \pi_k(\lambda), (\lambda^{-1} I + A^* A)^{-1} (x_{k-1} - \pi_k(\lambda)) \right\rangle.
$$

Moreover, G_k is strictly decreasing in $(0, \infty)$, the solution set of (19) is $[\lambda_{\min}, \lambda_{\max}] \cap \mathbb{R}$ where

$$
\lambda_{\min} := \min\{\lambda > 0 : G_k \leq (p\|A x_{k-1}^\delta - y^\delta\| + (1-p)\delta)^2\}
$$
$$
\lambda_{\max} := \sup\{\lambda > 0 : \delta^2 \leq G_k(\lambda)\},
$$

and $0 < \lambda_{\min} < \lambda_{\max} \leq \infty$.

We will solve (19) by means of a Newton-type method. Newton's method for solving (19), would be to take some $\lambda_{k,0} >$ and to iterate

$$
\lambda_{k,j+1} = \lambda_{k,j} - \frac{G_k(\lambda_{k,j}) - \delta^2}{G_k'(\lambda_{k,j})}
$$

as long as (19) is not satisfied; when (19) is satisfied, the last $\lambda_{k,j}$ is used as λ_k. We will introduce a number of modifications in this iteration to accelerate it:

(M1) A "greedy" version of Newton's method will be used, aiming at $G_k(\lambda) = 0$, that is, the numerator on the above fraction will be $G_k(\lambda_{k,j})$.

11

(M2) Newton's step will be dynamically over-relaxed by a factor w_j, as described below.

(M3) We choose $\lambda_{k,0}$ using Lemma 2.4 for $k = 0$ and information gathered at previous iterations of $k \geq 2$, ad described below.

Regarding modifications (M1) and (M2), while $\lambda = \lambda_{k,j}$ does not satisfy (19), we use the iteration

$$\lambda_{k,j+1} \; := \; \lambda_{k,j} - \omega_j \frac{G_k(\lambda_{k,j})}{G'_k(\lambda_{k,j})} \,, \tag{20}$$

where the over-relaxation factor ω_j is chosen as follows:
— for $j = 0$, $\omega_0 = 1$;
— for $j \geq 1$, after computing $\lambda_{k,j}$, $\pi_k(\lambda_{k,j})$, and $G_k(\lambda_{k,j})$,
      if $G_k(\lambda_{k,j-1}) > 2\big(p\|Ax_{k-1}^\delta - y^\delta\| + (1-p)\delta\big)^2$
         then $\omega_j = 2\,\omega_{j-1}$
         else $\omega_j = 1$
Regarding modification (M3):
— for $k = 1$, $\lambda_{1,0}$ is the lower bound provided by Lemma 2.4;
— for $k = 2$, $\lambda_{2,0} = \lambda_1$,
— for $k \geq 3$, we use a linear extrapolation on $\log \lambda$ from the two previous iterates as starting point, that is, $\lambda_{k,0} = \lambda_{k-1}^2/\lambda_{k-2}$.

    The acceleration effect caused by modifications (M1), (M2) and (M3) is illustrated in Example 4.2 below.

    Over-relaxation is a well established technique for accelerating iterative methods for solving linear and non-linear equations, the SOR method being is a classical example. This facts motivated the introduction of over-relaxation as in (M2).

    In our numerical experiments, we observed that the sequence λ_k increases exponentially. This fact motivated the use of (liner) extrapolation (in the log) for its initial value from iteration 3 on (modification (M3)).

    It is worth noticing that, in step 3, each "inner iteration" (20) requires the solution of either two linear systems per failed inner iteration or one linear system at the successful last inner iteration. On the other hand, in the gNIT method the computation of $\lambda_k = q^k$ (for some *a priori* chosen $q > 1$) is straightforward. Consequently, one needs to solve one linear system (modeled by $(I + q^k A^* A)$) in each step of the gNIT method. This facts motivated us to use the accumulated number of linear system to measure the performance of the different NIT method by plotting the residual and the error as a function of this quantity (see Figure 3).

Example 4.2. *The benchmark problem presented in Example 1.1 is revisited and solved by the rrNIT method using the Newton-method (20) with combinations of modifications (M1)-(M3): In Figure 3 this inverse problem is solved using:*
(BLUE) modification (M1);
(PINK) modifications (M1), (M2);
(RED) modifications (M1), (M2), (M3);
Notice that, in Figure 3, the x-axis denotes the accumulated number of linear systems. This choice allows an better comparison of the efficiency of the different rrNIT implementations.

    Figure 3 illustrates that the *cumulative* effect of modifications (M1), (M2), and (M3) is to accelerate the computation of λ_k as required in step [3] of Algorithm 2.

    A pseudo-code version of our implementation of Algorithm 2 with the strategies above discussed is presented in Appendix A for the sake of completeness.
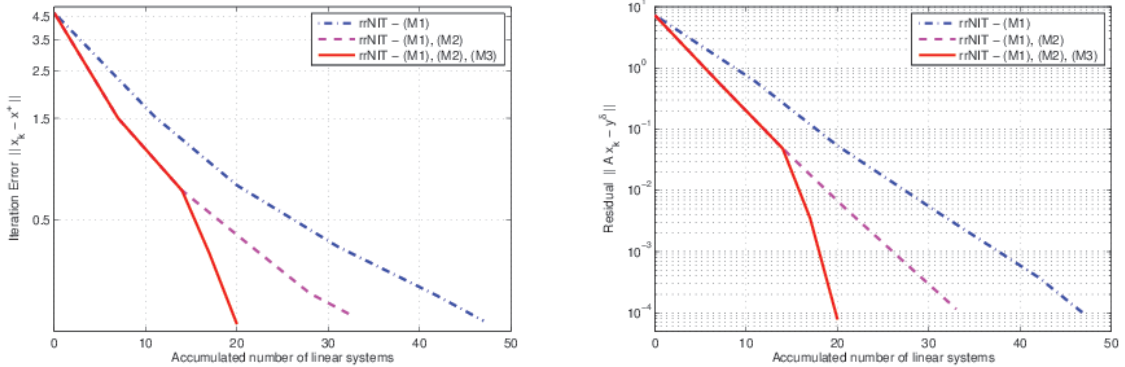
Figure 3: Implementation of rrNIT method: acceleration caused by modifications (M1), (M2), (M3) to iteration (20).

5   Numerical experiments

In this section, Algorithm 3 (see Appendix A) implemented for solving two well known linear ill-posed problems. In Section 5.1 the *Image Deblurring Problem* [4, 3] is considered, while in Section 5.2 we address the *Inverse Potential Problem* [18, 20] in 2D, which is an elliptic parameter identification problem.

In both Sections 5.1 and 5.2, the performance of Algorithm 3 is compared to the performance of two well established methods: The gNIT method with $\lambda_k = 2^k$; the NIT method proposed in [8].

5.1   Image deblurring problem

Image deblurring problems [4] are finite dimensional problems modeled, in general, by high dimensional linear systems of the form (2). In this setting, $x \in X = \mathbb{R}^n$ represents the pixel values of an unknown true image, while $y \in Y = X$ contains the pixel values of the observed (blurred) image. In practice, only noisy blurred data $y^\delta \in Y$ satisfying (1) is available.

The matrix A describes the blurring phenomenon [3, 4]. We consider the simple situation where the blur of the image is modeled by a space invariant point spread function (PSF). In the continuous model, the blurring process is represented by an integral operator of convolution type and (2) corresponds to an integral equation of the first kind [12]. In our discrete setting, after incorporating appropriate boundary conditions into the model, the discrete convolution is evaluated by means of the FFT algorithm. We added to the exact data (the convoluted image) a normally distributed noise with zero mean and suitable variance for achieving a prescribed relative noise level .

The computation of our deblurring experiment was conducted using MATLAB 2012a. The corresponding setup is shown in Figure 4: (a) True image $x \in \mathbb{R}^n$, $n = 256^2$ (Cameraman 256×256); (b) PSF is the rotationally symmetric Gaussian low-pass filter of size [257 257] and standard deviation $\sigma = 4$ (command `fspecial('gaussian', [257 257], 4.0)`); (c) Exact data $y = Ax \in \mathbb{R}^n$ (blurred image). The noise was generated used the `randn` routine while the FFT was computed using the `fft2` routine.

Three distinct scenarios are considered, where the relative noise level $\|y - y^\delta\|/\|y\|$ corresponds to $10^{-1}\%$, $10^{-3}\%$ and $10^{-6}\%$ respectively (in the third scenario, the choice of the noise level is motivated by MATLAB's double-precision accuracy).
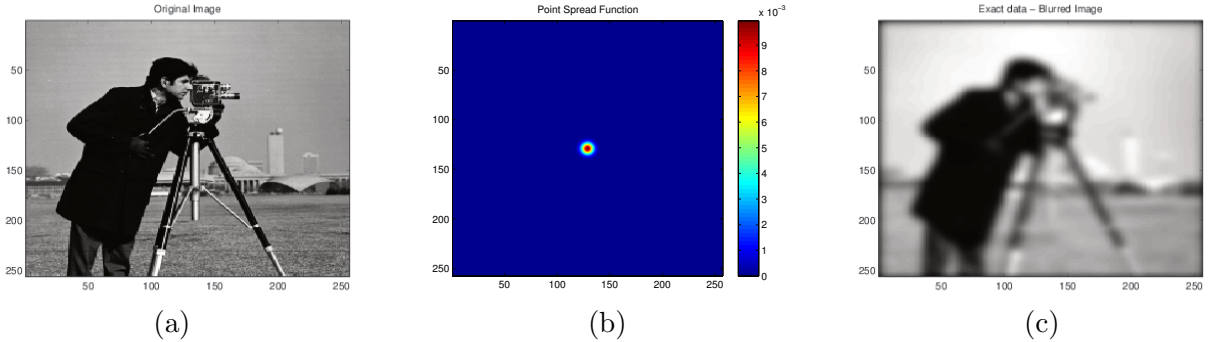
13

Figure 4: Image deblurring problem: setup of the inverse problem. (a) Original image x; (b) Point spread function; (c) Blurred image y.

In Figure 5 the following methods are compared for the second scenario: (BLACK) gNIT with $\lambda_k = 2^k$; (RED) rrNIT method in Algorithm 3 (with $p = 0.2$); (BLUE) Hanke-Donatelli NIT method in [8]. The pictures in Figure 5 show: (TOP) relative error $\|x^\star - x_k^\delta\|/\|x^\star\|$; (BOTTOM) residual $\|Ax_k^\delta - y^\delta\|$. The x-axis in the these pictures is scaled by the accumulated number of linear systems solved. This choice allows an easier comparison between the efficiency of the different methods.

All methods are stopped according to the discrepancy principle with $\tau = 3$. As initial guess we choose $x_0 = y^\delta$ (the noisy blurred image).

The numerical results concerning all three scenarios are summarized in Table 1. In this table we show, for each scenario, the total number of linear systems solved, as well as the number of iterations needed to reach the stop criteria.

	gNIT	NIT in [8]	rrNIT
$10^{-1}\%$	6 (6)	15 (5)	7 (4)
$10^{-3}\%$	17 (17)	23 (7)	11 (7)
$10^{-6}\%$	36 (36)	43 (11)	16 (11)

Table 1: Image deblurring problem: total number of linear systems for different noise levels with the number of iterations in parentheses.

The restored images for the second scenario ($\delta = 10^{-3}\%$) are presented in Figure 6. From left to right: gNIT, NIT in [8], and rrNIT.

5.2 Inverse potential problem

In what follows we address the *inverse potential problem* [14, 7, 18, 30]. Generalizations of this inverse problem appear in many relevant applications including Inverse Gravimetry [20, 30], EEG [10], and EMG [31].

The forward problem considered here consists in solving on a Lipschitz domain $\Omega \subset \mathbb{R}^d$, for a given source function $x \in L_2(\Omega)$, the boundary value problem

$$-\Delta u = x, \text{ in } \Omega, \quad u = 0 \text{ on } \partial\Omega. \tag{21}$$

The corresponding inverse problem is the so called *inverse potential problem* (IPP), which consists of recovering an L_2–function x, from measurements of the Dirichlet data of its corresponding potential on the boundary of Ω, i.e., $y := u_\nu|_{\partial\Omega} \in L_2(\partial\Omega)$. This problem is modeled
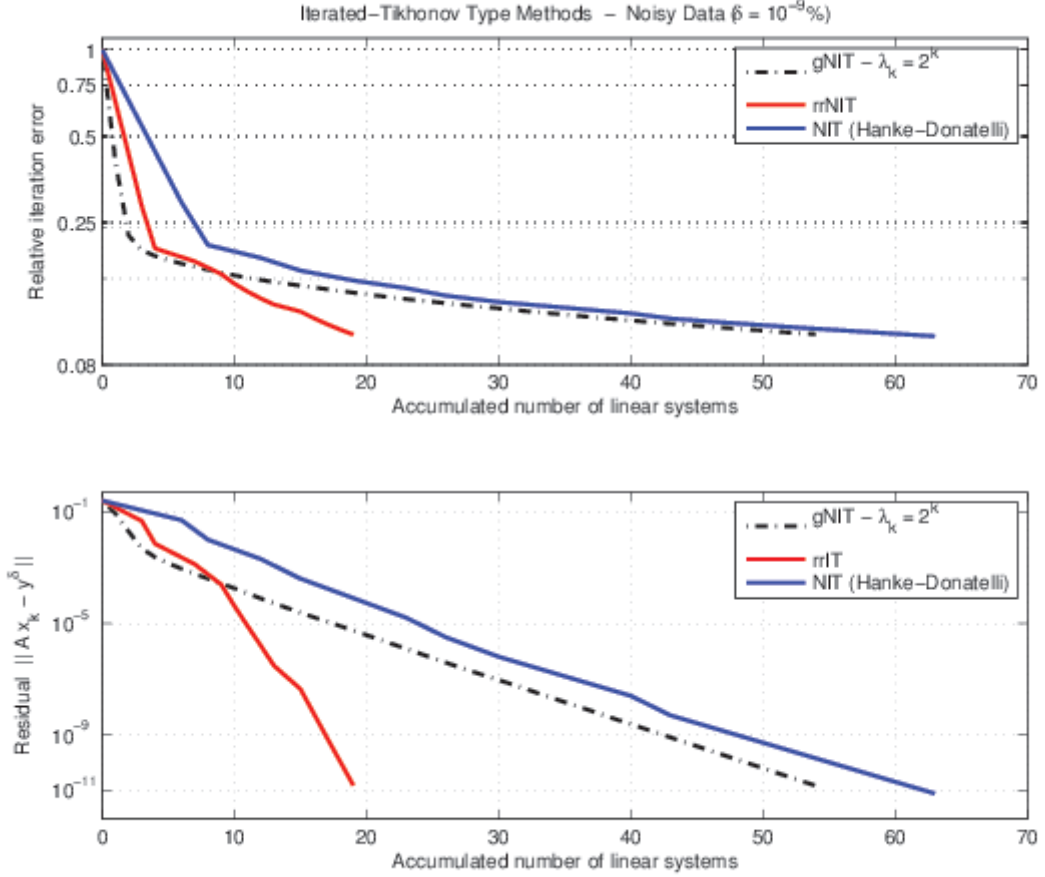
Figure 5: Image deblurring problem: second scenario $\delta = 10^{-3}\%$.

by the linear operator $A : L_2(\Omega) \to L_2(\partial\Omega)$ defined by $Ax := u_\nu|_{\partial\Omega}$, where $u \in H_0^1(\Omega)$ is the unique solution of (21) [18]. Using this notation, the IPP can be written in the abbreviated form (2), where the available noisy data $y^\delta \in L_2(\partial\Omega)$ satisfies (1).

In our experiments we follow [7] in the experimental setup, selecting $\Omega = (0,1) \times (0,1)$ and assuming that the unknown parameter $x^\star$ is an H^1-function with sharp gradients shown in Figure 7 (a). After solving Problem (21) for such $x = x^\star$, we added to the exact Dirichlet data a normally distributed noise with zero mean and suitable variance for achieving a prescribed relative noise level. In our numerical implementations we set $p = 0.1$, $\tau = 3$ (discrepancy principle constant) and the initial guess $x_0 \equiv 1.5$ (constant function in Ω).

As in Section 5.1, three distinct scenarios are considered, where the relative noise level $\|y - y^\delta\|/\|y\|$ corresponds to $10^{-1}\%$, $10^{-3}\%$ and $10^{-6}\%$ respectively.

In Figure 8 the following methods are compared for the second scenario: (BLACK) gNIT with $\lambda_k = 2^k$; (RED) rrNIT method in Algorithm 3 (with $p = 0.1$); (BLUE) NIT method proposed in [8].[5] What concerns Algorithm 3, the corresponding iterate x_4^δ and iteration error $\|x^\star - x_4^\delta\|$ are shown in Figure 7 (b) and (c) respectively.

The pictures in Figure 5 show: (TOP) relative error $\|x^\star - x_k^\delta\|/\|x^\star\|$; (BOTTOM) residual

---

[5]As before, this NIT method was implemented with $q = 0.6$ and $\rho = 10^{-4}$. For the computation of the Lagrange multipliers, a scalar equation was solved using an over-relaxed Newton method and a precision of 5%.
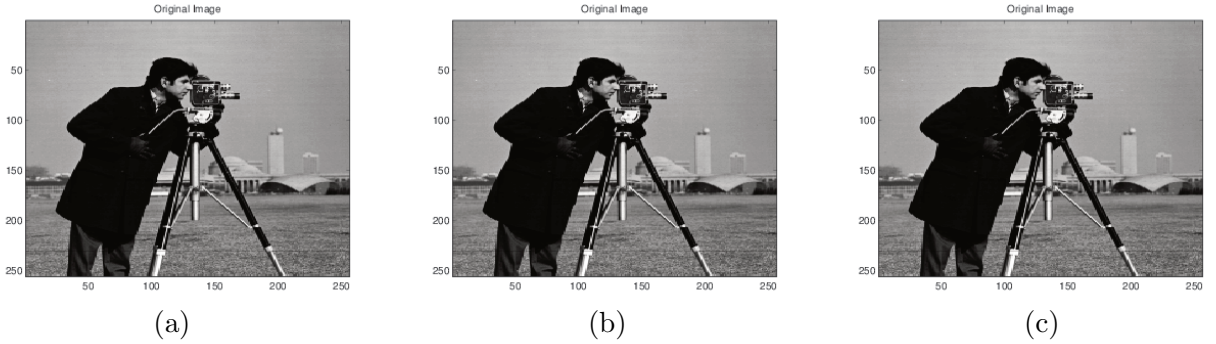
Figure 6: Image deblurring problem: restored images for the second scenario. (a) gNIT, (b) NIT in [8], (c) rrNIT.

$\|Ax_k^\delta - y^\delta\|$. The x-axis in the these pictures is scaled by the accumulated number of linear systems solved.

The numerical results concerning all three scenarios are summarized in Table 2. In this table we show, for each scenario, the total number of linear systems solved, as well as the number of iterations needed to reach the stop criteria.

	gNIT	NIT in [8]	rrNIT
$10^{-1}\%$	6 (6)	11 (3)	6 (3)
$10^{-3}\%$	10 (10)	34 (5)	10 (5)
$10^{-6}\%$	13 (13)	86 (7)	12 (6)

Table 2: Inverse potential problem: total number of linear systems for different noise levels with the number of iterations in parentheses.

5.3 Remarks

In the two above discussed inverse problems, for all scenarios, both *a posteriori* NIT type methods (rrNIT in Algorithm 3 and NIT in [8]) require similar number of steps to reach discrepancy. However, the total numerical effort of rrNIT is much smaller than the one of NIT in [8], and is comparable to the total numerical effort of the gNIT method (see Tables 1 and 2).

Specially in the third scenario (small noise level), the NIT method [8] needs several Newton steps to compute the Lagrange multipliers in the final iterations. Algorithm 3, on the other hand, needs only few Newton steps to compute each one of the Lagrange multipliers solving (19).

Notice the exponential decay of the residual in the rrNIT method (Figures 5 and 8), which is in accordance to Proposition 3.1. We also observed exponential growth of the corresponding Lagrange multipliers.

The NIT method proposed in [8] was implemented with $q = 0.6$ and $\rho = 10^{-4}$ as described in [8, Sec.5]. For the computation of the Lagrange multipliers, a scalar equation was solved using an over-relaxed Newton method with a precision of 1% (Section 5.1) and 5% (Section 5.2).
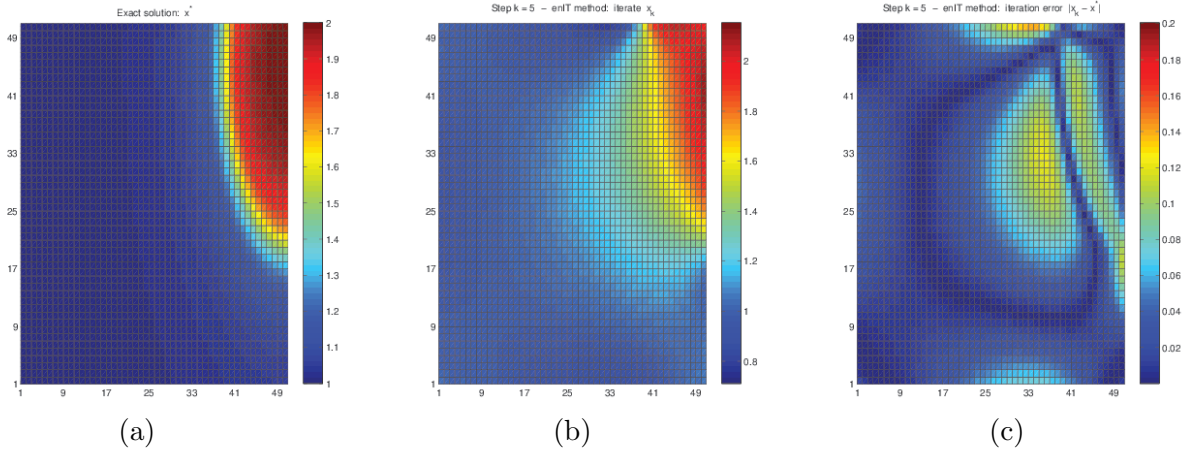
Figure 7: Inverse potential problem: rrNIT method in Algorithm 3. (a) Exact solution $x^\star$; (b) Approximate solution x_k^δ, for $k = 5$; (c) Iteration error $\|x^\star - x_5^\delta\|$.

6 Conclusions

We investigate NIT type methods for computing stable approximate solutions to ill-posed linear operator equations. The main contributions of this article is a novel strategy for choosing a sequence of Lagrange multipliers for the NIT iteration, allowing each of this multipliers to belong to a non-degenerate interval. We also derived an efficient numerical algorithm based on this strategy for computing the required Lagrange multipliers.

We prove monotonicity of the proposed rrNIT method, and exponential decay of the residual $\|Ax_k^\delta - y^\delta\|^2$. Moreover, we provide estimates to the "gain" $\|x^\star - x_{k-1}^\delta\|^2 - \|x^\star - x_k^\delta\|^2$, and to the Lagrange multipliers λ_k. A convergence proof in the case of exact data is provided.

An algorithmic implementation of the rrNIT method is proposed (Algorithm 3, where the computation of Lagrange multipliers are accomplished using an over relaxed Newton-like method, with appropriate choice of the initial guess. The resulting rrNIT method is competitive with gNIT and also with other commonly used *a posteriori* method ; not only from the point of view of the total number of iterations, but also from the point of view of the overall numerical effort required.

Our algorithm is tested for two well known applications with three noise levels: the inverse potential problem, and the image deblurring problem. The results obtained validate the efficiency of our method.
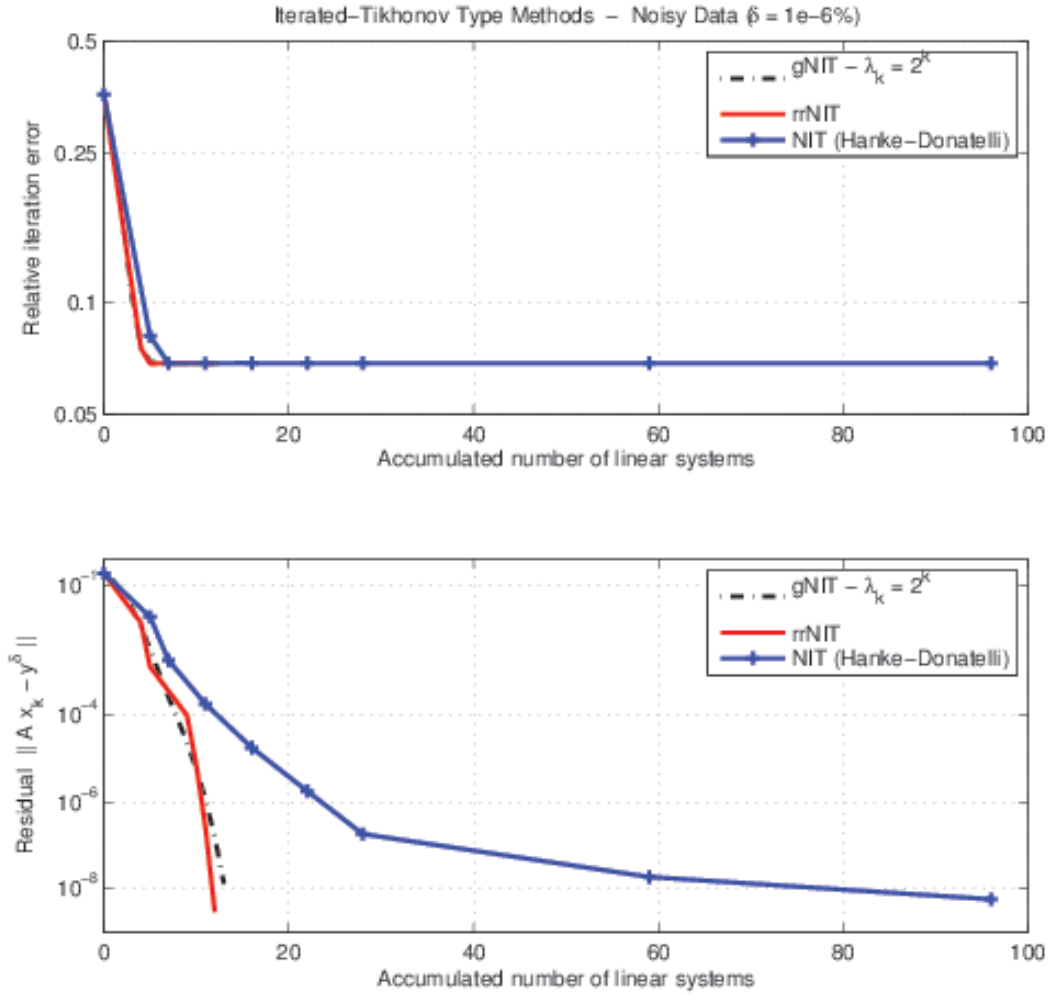
Acknowledgments

Figure 8: Inverse potential problem: second scenario $\delta = 10^{-3}\%$.

Appendix A

In what follows we present a detailed algorithm for the rrNIT method, which takes into account the above discussed strategies, namely: initial guess choice and over-relaxation.

Algorithm 3, an implementable rrNIT method for solving ill-posed linear problems, is written in a tutorial way.[6] Presented in this form, one recognizes that the major computational effort in each iteration consists in the computation of the M_λ operators. In the first iteration ($k = 1$) this task is solved in steps [3.3] and [3.7]; in the subsequent iterations it is solved in the Newton-method [3.7].

The above discussed choice of the initial guess $\lambda_{k,0}$ for the Newton-method (20) is evaluated in step [3.3]. Moreover, the computation of the over-relaxation parameters ω_j is implemented in loop [3.7].

---

[6]Indeed, the inversion of $(I + \lambda A^* A)$ is not always possible.

18

[1] choose an initial guess $x_0 \in X$;

[2] choose $p \in (0,1)$, $\tau > 1$ and set $k := 0$;

[3] while $\left(\|Ax_k - y^\delta\|_Y > \tau\delta \right)$ do

    [3.1] $k := k + 1$;

    [3.2] $\theta_k := p\|Ax_{k-1}^\delta - y^\delta\|_Y + (1-p)\delta$;

    [3.3] if $(k = 1)$ then

        $\lambda_{k,0} := \|Ax_{k-1}^\delta - y^\delta\|\left(\|Ax_{k-1}^\delta - y_\delta\| - \theta_k\right) / \|A^*(Ax_{k-1}^\delta - y^\delta)\|^2$;

        $M_{\lambda_{k,0}} := (I + \lambda_{k,0}A^*A)^{-1}$;

      else

        $\lambda_{k,0} := \lambda_{k-1}$;

        $M_{\lambda_{k,0}} := M_{\lambda_{k-1,1}}$;

      endif

    [3.4] $x_{\lambda_{k,0}} := x_{k-1}^\delta - \lambda_{k,0} M_{\lambda_{k,0}} A^*(Ax_{k-1}^\delta - y^\delta)$;

    [3.5] compute $G_k(\lambda_{k,0}) = \|Ax_{\lambda_{k,0}} - y^\delta\|^2$;

    [3.6] $j := 0$; $\omega_0 := 1$;

    [3.7] while $\left(G_k(\lambda_{k,j}) > \theta_k^2\right)$ do

        compute $DG_k(\lambda_{k,j}) = \left\langle A^*(Ax_{\lambda_{k,j}} - y^\delta), M_{\lambda_{k,j}} A^*(Ax_{\lambda_{k,j}} - y^\delta)\right\rangle$;

        $j := j + 1$;

        $\lambda_{k,j} := \lambda_{k,j-1} - \omega_{j-1} G_k(\lambda_{k,j-1})/DG_k(\lambda_{k,j-1})$;

        $M_{\lambda_{k,j}} := (I + \lambda_{k,j}A^*A)^{-1}$;

        $x_{\lambda_{k,j}} := x_{k-1}^\delta - \lambda_{k,j} M_{\lambda_{k,j}} A^*(Ax_{k-1}^\delta - y^\delta)$;

        compute $G_k(\lambda_{k,j}) = \|Ax_{\lambda_{k,j}} - y^\delta\|^2$;

        update over-relaxation parameter ω_j;

      end of while [3.7]

    [3.8] $x_k^\delta := x_{\lambda_{k,j}}$; $\lambda_k := \lambda_{k,j}$;

  end of while [3]

Algorithm 3: Numerical algorithm for the rrNIT method.

References

[1] A.B. Bakushinsky and M.Y. Kokurin, *Iterative Methods for Approximate Solution of Inverse Problems*, Mathematics and Its Applications, vol. 577, Springer, Dordrecht, 2004.

[2] J. Baumeister, *Stable Solution of Inverse Problems*, Advanced Lectures in Mathematics, Friedr. Vieweg & Sohn, Braunschweig, 1987. MR 889048

[3] M. Bertero, *Image deblurring with Poisson data: from cells to galaxies*, Inverse Problems **25** (2009), no. 12, 123006.

[4] M. Bertero and P. Boccacci, *Introduction to Inverse Problems in Imaging*, Advanced Lectures in Mathematics, IOP Publishing, Bristol, 1998.

[5] M. Brill and E. Schock, *Iterative solution of ill-posed problems: A survey*, ch. in Model Optimization in Exploration Geophysics, Ed. A. Vogel, pp. 13–38, Vieweg, Braunschweig, 1987.

[6] F.E. Browder and W.V. Petryshyn, *Construction of fixed points of nonlinear mappings in Hilbert space*, Journal of Mathematical Analysis and Applications **20** (1967), no. 2, 197–228.

[7] A. De Cezaro, A. Leitão, and X.-C. Tai, *On multiple level-set regularization methods for inverse problems*, Inverse Problems **25** (2009), 035004.

[8] M. Donatelli and M. Hanke, *Fast nonstationary preconditioned iterative methods for ill-posed problems, with application to image deblurring*, Inverse Problems **29** (2013), no. 9, 095008, 16.

[9] Marco Donatelli, *On nondecreasing sequences of regularization parameters for nonstationary iterated Tikhonov*, Numer. Algorithms **60** (2012), no. 4, 651–668. MR 2946487

[10] A. El Badia and M. Farah, *Identification of dipole sources in an elliptic equation from boundary measurements: application to the inverse EEG problem*, J. Inverse Ill-Posed Probl. **14** (2006), no. 4, 331–353.

[11] H.W. Engl, *On the choice of the regularization parameter for iterated Tikhonov regularization of ill-posed problems*, J. Approx. Theory **49** (1987), no. 1, 55–63.

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

[13] A.G. Fakeev, *A class of iterative processes for solving degenerate systems of linear algebraic equations*, U.S.S.R. Comput. Math. Math. Phys. **21** (1981), no. 3, 15–22.

[14] F. Frühauf, O. Scherzer, and A. Leitão, *Analysis of Regularization Methods for the Solution of Ill-Posed Problems Involving Discontinuous Operators*, SIAM J. Numer. Anal. **43** (2005), 767–786.

[15] C. W. Groetsch, *The Theory of Tikhonov Regularization for Fredholm Equations of the First Kind*, Research Notes in Mathematics, vol. 105, Pitman (Advanced Publishing Program), Boston, MA, 1984.

[16] M. Hanke, *A regularizing Levenberg-Marquardt scheme, with applications to inverse groundwater filtration problems*, Inverse Problems **13** (1997), no. 1, 79–95.

[17] M. Hanke and C. W. Groetsch, *Nonstationary Iterated Tikhonov Regularization*, J. Optim. Theory Appl. **98** (1998), no. 1, 37–53.

[18] F. Hettlich and W. Rundell, *Iterative methods for the reconstruction of an inverse potential problem*, Inverse Problems **12** (1996), 251–266.

[19] B. Hofmann, *Regularization for Applied Inverse and Ill-Posed Problems*, Teubner-Texte zur Mathematik [Teubner Texts in Mathematics], vol. 85, BSB B. G. Teubner Verlagsgesellschaft, Leipzig, 1986, A numerical approach, With German, French and Russian summaries.

[20] Victor Isakov, *Inverse Problems for Partial Differential Equations*, second ed., Applied Mathematical Sciences, vol. 127, Springer, New York, 2006.

[21] B. Kaltenbacher, A. Neubauer, and O. Scherzer, *Iterative Regularization Methods for Nonlinear Ill-Posed Problems*, Radon Series on Computational and Applied Mathematics, vol. 6, Walter de Gruyter GmbH & Co. KG, Berlin, 2008.

[22] J.T. King and D. Chillingworth, *Approximation of generalized inverses by iterated regularization*, Numer. Funct. Anal. Optim. **1** (1979), 499–513.

[23] A. Kirsch, *An Introduction to the Mathematical Theory of Inverse Problems*, Applied Mathematical Sciences, vol. 120, Springer-Verlag, New York, 1996.

[24] L. J. Lardy, *A series representation for the generalized inverse of a closed linear operator*, Atti della Accademia Nazionale dei Lincei, Rendiconti della Classe di Scienze Fisiche, Matematiche, e Naturali, Serie VIII **58** (1975), 152–157.

[25] A. Louis, *Inverse und schlecht gestellte Probleme*, B.G. Teubner, Stuttgart, 1989.

[26] V.A. Morozov, *Regularization Methods for Ill–Posed Problems*, CRC Press, Boca Raton, 1993.

[27] F. Natterer and F. Wübbeling, *Mathematical Methods in Image Reconstruction*, SIAM, Philadelphia, 2001.

[28] O. Scherzer, *Convergence rates of iterated Tikhonov regularized solutions of nonlinear ill-posed problems*, Numer. Math. **66** (1993), no. 2, 259–279.

[29] J.L. Troutman, *Variational Calculus and Optimal Control*, second ed., Undergraduate Texts in Mathematics, Springer-Verlag, New York, 1996, Optimization with elementary convexity.

[30] K. van den Doel, U. M. Ascher, and A. Leitão, *Multiple Level Sets for Piecewise Constant Surface Reconstruction in Highly Ill-Posed Problems*, Journal of Scientific Computing **43** (2010), no. 1, 44–66.

[31] Kees van den Doel, Uri M. Ascher, and Dinesh K. Pai, *Computed myography: three-dimensional reconstruction of motor functions from surface EMG data*, Inverse Problems **24** (2008), no. 6, 065010, 17.