IMPROVED REGULARIZING ITERATIVE METHODS FOR ILL-POSED NONLINEAR SYSTEMS

S. BELLA VIA† AND B. MORINI†

Abstract. In this paper we address the numerical solution of nonlinear ill-posed systems by iterative regularization methods in the classes of Levenberg-Marquardt, trust-region and adaptive quadratic regularization procedures. Both with exact and noisy data, our focus is on the potential to approach a solution of the unperturbed systems without assumptions on its vicinity to the initial guess. Regularizing properties of the methods proposed are shown theoretically and validated numerically along with enhanced convergence.

Keywords: Ill-posed nonlinear systems of equations, regularization, nonlinear stepsize control, local and global convergence properties.

1. Introduction. The numerical solution of systems of nonlinear equations is a well established area of practical optimization, see e.g., [4, 19]. However, most of the methods proposed in literature are designed for well-posed systems and are unsuited in the context of the inverse problems. In fact, the nonlinear systems modeling inverse problems are typically ill-posed, in the sense that their solutions do not depend continuously on the data and their data are affected from noise [5, 11, 22].

Let

\[ F(x) = y, \]  

(1.1)

with \( F : \mathbb{R}^n \to \mathbb{R}^n \) continuously differentiable, be obtained from the discretization of a problem modeling an inverse problem. It is realistic to have only noisy data \( y^\delta \) at disposal, satisfying

\[ \|y - y^\delta\| \leq \delta, \]  

(1.2)

for some positive \( \delta \). Thus, in practice it is necessary to solve a problem of the form

\[ F(x) = y^\delta, \]  

(1.3)

and, due to ill-posedness, possible solutions may be arbitrarily far from those of the original problem.

In the seminal papers [7, 8], Hanke supposed that an initial guess, close enough to some solution \( x^\dagger \) of (1.1), is available. Then, he proposed a regularizing Levenberg-Marquardt procedure which is capable to compute a stable approximation \( x^\delta_k \) to \( x^\dagger \) or to some other solution of the unperturbed problem (1.1) close to \( x^\dagger \). Such an approximation is found by using a nonlinear stepsize control in the Levenberg-Marquardt procedure and the discrepancy principle as the stopping criterion. In practice, the iterative process is stopped at the iteration \( k^* \) such that

\[ \|y^\delta - F(x^\delta_k)\|_2 \leq \tau \delta < \|y^\delta - F(x^\delta_k)\|_2, \quad 0 \leq k < k^*, \]  

(1.4)

where \( \tau > 1 \) is an appropriately chosen positive scalar [17]. Remarkably \( x^\delta_k \) converges to a solution of (1.1) as \( \delta \) tends to zero.

†Dipartimento di Ingegneria Industriale, Università di Firenze, viale G.B. Morgagni 40, 50134 Firenze, Italia, stefania.bellavia@unifi.it, benedetta.morini@unifi.it

*Work partially supported by INdAM-GNCS, under the 2014 Projects “Metodi di regolarizzazione per problemi di ottimizzazione vincolata”
Further regularizing iterative methods have been proposed, including first-order methods and Newton-type methods. They share the same regularizing properties as the Levenberg-Marquardt procedure proposed by Hanke, and we refer to the book [11] for their description and analysis.

The above mentioned regularizing Levenberg-Marquardt method belongs to the unifying framework of nonlinear stepsize control algorithms for unconstrained optimization developed by Toint [20]. Elaborating on original ideas by Hanke and on procedures from this framework, we introduce and analyze regularizing variants of the Levenberg-Marquardt and two further methods: trust-region [4] and Adaptive Quadratic Regularization (ARQ) [1, 2] methods. In particular, we introduce regularizing trust-region and ARQ methods which may approach a solution of (1.1) even if an accurate initial guess is not available. Our procedures applied to (1.3) generate iterates $x^\delta_k$ such that the value of the residual $\| F(x^\delta_k) - y^\delta \|_2$ monotonically decreases. Their convergence properties are enhanced with respect to the Levenberg-Marquardt procedure existing in literature in the following respects. With exact data, if there exists an accumulation point of the iterates which solves (1.1), then any accumulation point of the sequence solves (1.1). With noisy data, the methods have the potential to satisfy the discrepancy principle (1.4). In order to obtain these properties, the closeness of the initial guess to a solution of (1.1) is not crucial.

Our contributions cover theoretical and practical aspects of the methods considered. First, we have made an attempt toward global convergent methods for ill-posed problems; in this context we are aware only of a multilevel strategy proposed by Kaltenbacher in [10]. Second, we have carried out local convergence analysis without the assumption (commonly made in literature) on the boundness of the inverse of the Jacobian $J$ of $F$, since it may not be fulfilled in the situation of ill-posedness. Taking into account that the standard convergence analysis of both trust-region and ARQ methods always requires the invertibility of $J$, our results represent a progress in the theoretical investigation of these approaches. Third, we have compared numerically the methods discussed and tested their ability to approximate a solution of (1.1). We mention that trust-region methods have been applied to ill-posed nonlinear systems in the papers [12, 13, 23, 24], but issues on global convergence have not been addressed.

The paper is organized as follows. In §2 we describe the basic features of the Levenberg-Marquardt, trust-region and ARQ methods for well-posed systems. In §3 we analyze both the regularizing Levenberg-Marquardt method by Hanke and one possible variant. In §4 we introduce the regularizing versions of the trust-region and ARQ methods, then in §5 we study their local convergence properties. A comparative numerical analysis of all the procedures studied is done in §6.

Notations. We indicate the iterates of the procedures analyzed as $x^\delta_k$; if the data are exact, $x_k$ may be used in alternative to $x^\delta_k$. By $x^\delta_0 = x_0$ we denote an initial guess which may incorporate a-priori knowledge of an exact solution. The symbol $\| \cdot \|$ indicates the Euclidean norm. The Jacobian matrix of $F$ is denoted as $J$.

2. Preliminaries on the solution of nonlinear systems. In this section we describe some Newton-type methods for solving nonlinear systems of the form (1.3). We focus on two globally convergent approaches which are closely related to the Levenberg-Marquardt method: trust-region methods, see e.g., [4], and quadratic regularization methods [1, 18]. Starting from an arbitrary initial guess, these methods generate a sequence of iterates such that the $2$-norm of the nonlinear residual $y^\delta - F(x)$ is monotonically decreasing and this feature is enforced by a nonlinear control mechanism on the stepsize between two successive iterates.
The Levenberg-Marquardt method is a well-known iterative method for solving nonlinear least-squares problems and was originally proposed by Levenberg and Marquardt. At a generic iteration \( k \), given \( x_k^\delta \) and \( \lambda_k > 0 \), the step \( p_k \) taken minimizes the following quadratic model

\[
m_k^{LM}(p) = \frac{1}{2} \| F(x_k^\delta) - y^\delta + J(x_k^\delta)p \|^2_2 + \frac{1}{2} \lambda_k \| p \|^2,
\]

around \( x_k^\delta \), for the function

\[
\Phi(x) = \frac{1}{2} \| y^\delta - F(x) \|^2.
\]

Letting

\[
(B_k + \lambda I)p(\lambda) = -g_k,
\]

where \( B_k = J(x_k^\delta)^T J(x_k^\delta) \) and \( g_k = J(x_k^\delta)^T (F(x_k^\delta) - y^\delta) \), it is easy to see that \( p_k = p(\lambda_k) \); then the new iterate is given by \( x_{k+1}^\delta = x_k^\delta + p_k \). Moré proposed a robust and efficient implementation where \( \lambda_k \) depends on a maximum prefixed length of the step \( p_k \). Such a work highlighted a strict connection of the Levenberg-Marquardt method and trust-region methods. In fact, the latter methods, at a generic iteration \( k \), compute the step \( p_k \) solving

\[
\min_{p} m_k^{TR}(p) = \| F(x_k^\delta) - y^\delta + J(x_k^\delta)p \|^2,
\]

s.t. \( \| p \| \leq \Delta_k \),

where \( \Delta_k \) is a given positive trust-region radius.

If \( \| g_k \| \neq 0 \) then \( p_k \) solves (2.4) if and only if it satisfies

\[
\lambda_k (\| p_k \| - \Delta_k) = 0.
\]

Whenever the minimum norm solution \( p^+ \) of

\[
B_k p = -g_k,
\]

satisfies \( \| p^+ \| \leq \Delta_k \), the scalar \( \lambda_k \) is null and \( p_k = p(0) \) solves (2.4). Otherwise, the step \( p_k = p(\lambda_k) \) is a Levenberg-Marquardt step. If \( \| p_k \| = \Delta_k \), then the trust-region is said to be active. The trust-region radius \( \Delta_k \) is chosen adaptively in order to reduce the value of \( \| y^\delta - F(x) \|^2 \). Once the minimizer \( p_k \) has been computed, it is tested on the basis of the ratio between the achieved and predicted reduction

\[
\rho_k = \frac{\text{ared}(p_k)}{\text{pred}(p_k)},
\]

where

\[
\text{ared}(p_k) = \| F(x_k^\delta) - y^\delta \|^2 - \| F(x_k^\delta + p_k) - y^\delta \|^2,
\]

\[
\text{pred}(p_k) = \| F(x_k^\delta) - y^\delta \|^2 - m_k^{TR}(p_k).
\]

Specifically, the trust region radius is reduced if \( \rho_k \) is below some small positive threshold, otherwise it is left unchanged or enlarged.
A further method in the nonlinear stepsize control framework [20] is the Adaptive Quadratic Regularization (AQ) method proposed in [1, 18]. At a generic iteration \( k \), the step attempted solves the minimization problem

\[
\min_{p \in \mathbb{R}^n} m_k^{\text{ARQ}}(p) = \sqrt{\| F(x_k^\delta) - y^\delta + J(x_k^\delta)p\|^2 + \mu \|p\|^2 + \sigma_k \|p\|^2},
\]

where \( \sigma_k \) and \( \mu \) are given strictly positive parameters. The model \( m_k^{\text{ARQ}}(p) \) is strictly convex [1, Lemma 2.1], and continuously differentiable [1, p. 4]. The minimizer \( p_k \) of \( m_k^{\text{ARQ}}(p) \) solves (2.13), i.e. it has the form \( p_k = p(\lambda_k) \) and \( \lambda_k \) is the strictly positive parameter

\[
\lambda_k = \mu + 2\sigma_k \sqrt{\| F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k\|^2 + \mu \|p_k\|^2}.
\]

The couple \((p_k, \lambda_k)\) is unique [1, Lemma 4.1].

The role of the regularization term \( \sigma_k \) is to promote converge of the procedure. Once the approximate minimizer \( p_k \) has been computed, it is tested on the basis of the ratio (2.10) where \( \text{ared}(p_k) \) and \( \text{pred}(p_k) \) are defined as

\[
\text{ared}(p_k) = \| F(x_k^\delta) - y^\delta \| - \| F(x_k^\delta + p_k) - y^\delta \|,
\]

\[
\text{pred}(p_k) = \| F(x_k^\delta) - y^\delta \| - m_k^{\text{ARQ}}(p_k).
\]

The scalar \( \sigma_k \) is increased if \( \rho_k \) is below some small threshold; otherwise it is left unchanged or decreased.

In the three methods considered, the steps satisfy (2.3). For the subsequent analysis it is useful to establish relations between \( \lambda_k \), \( \|p(\lambda)\| \), and \( \| F(x_k^\delta) - y^\delta + J(x_k^\delta)p(\lambda)\| \).

**Lemma 2.1.** [1, Lemma 4.2] Suppose \( \|g_k\| \neq 0 \) and let \( p(\lambda) \) be the minimum norm solution of (2.3) with \( \lambda \geq 0 \). Suppose furthermore that \( J(x_k^\delta) \) is of rank \( \ell \) and its singular-value decomposition is given by \( U_k \Sigma_k V_k^T \) where \( \Sigma_k \) is the diagonal matrix with entries \( \gamma_1, \ldots, \gamma_n \) on the diagonal. Then, denoting \( r = U_k^T (F(x_k^\delta) - y^\delta) \), we have that

\[
\|p(\lambda)\|^2 = \sum_{i=1}^\ell \frac{\gamma_i^2 r_i^2}{(\gamma_i^2 + \lambda)^2},
\]

\[
\| F(x_k^\delta) - y^\delta + J(x_k^\delta)p(\lambda)\|^2 = \sum_{i=1}^\ell \frac{\lambda^2 r_i^2}{(\gamma_i^2 + \lambda)^2} + \sum_{i=\ell+1}^n r_i^2.
\]

The original proposals of the above methods and most of the existing literature on them concerns the solution of well-posed problems with exact data. Due to their relevant similarities in the form of the step, a single unifying framework can be used for their description and theoretical analysis [6, 20]. However, when the problem (1.4) is ill-posed and the data are noisy, the solution of (1.4) may be significantly misinterpreted if the regularizations are limited to promote convergence. In this case, specific procedures, named regularizing methods, are needed to approximate solutions of (1.4) [5, 7, 11].

Given an initial guess \( x_0 \), close enough to some solution \( x^\dagger \) of (1.4), regularizing methods aim at computing a (stable) approximation of \( x^\dagger \). Therefore, instead
of promoting convergence to a solution of \((3.1)\), these schemes provide approximations of increasing accuracy to \(x^1\) or some other solution of the unperturbed problem \((1.1)\) until \((1.4)\) is met. Remarkably, the approximations which satisfy \((1.4)\) converge to a solution of \((1.1)\) as \(\delta\) tends to zero. In the rest of the paper, elaborating on original ideas by Hanke \cite{Hanke2013} we will introduce and analyze regularizing variants of the Levenberg-Marquardt, trust-region and ARQ methods for handling ill-posed problems, possibly with noisy data.

3. Regularizing Levenberg-Marquardt method for ill-posed problems.

In this section we introduce the regularizing Levenberg-Marquardt method proposed in \cite{Hanke2013} and based on a specific choice of the regularization parameter \(\lambda_k\) in \((2.1)\). This parameter is selected as the solution \(\lambda_k^q\) of the nonlinear scalar equation

\[
\| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) \| = q\| F(x_k^q) - y^\delta \|,
\]

for some fixed \(q \in (0, 1)\).

We start discussing when equation \((3.1)\) admits a positive solution and how to find it in a reliable and efficient way. Let \(R(J(x_k^q))^\perp\) be the orthogonal complement of the range \(R(J(x_k^q))\) of \(J(x_k^q)\). Then if \(P_k^\delta\) is the orthogonal projector onto \(R(J(x_k^q))^\perp\), from Lemma \(2.1\) we get

\[
\lim_{\lambda \to 0} \| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) \| = \| P_k^\delta(F(x_k^q) - y^\delta) \|, \\
\lim_{\lambda \to \infty} \| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) \| = \| F(x_k^q) - y^\delta \|,
\]

see also \cite{Hanke2013} p. 65]. These two relations characterize the solution of \((3.1)\).

Lemma 3.1. Suppose \(\|g_k\| \neq 0\). Let \(p(\lambda)\) be the minimum norm solution of \((2.3)\) with \(\lambda \geq 0\) and \(P_k^\delta\) be the orthogonal projector onto \(R(J(x_k^q))^\perp\). Then

(i) Equation \((3.1)\) is not solvable if \(\| P_k^\delta(F(x_k^q) - y^\delta)\| > q\| F(x_k^q) - y^\delta \|\).

(ii) If \(\| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) \| \leq \frac{q}{\theta_k}\| F(x_k^q) - y^\delta \|\), for some \(\theta_k > 1\), then equation \((3.1)\) has a unique solution \(\lambda_k^q\) such that

\[
\lambda_k^q \in \left[ 0, \frac{q}{1-q}\| J(x_k^q)\|^2 \right].
\]

Proof. (i) By \((3.2)\), \((3.3)\) and the fact that \(\| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) \|\) is monotonically increasing as a function of \(\lambda\), we conclude that \((3.1)\) does not admit solution if

\[
\| P_k^\delta(F(x_k^q) - y^\delta)\| > q\| F(x_k^q) - y^\delta \|\]

(ii) Trivially \(\| P_k^\delta(F(x_k^q) - y^\delta)\| \leq \| F(x_k^q) - y^\delta - J(x_k^q)(x - x_k^q)\|\), for any \(x\). Hence, for the monotonicity of \(\| F(x_k^q) - y^\delta + J(x_k^q)p(\lambda)\|\), if \((3.4)\) holds, then equation \((3.1)\) admits a solution which is positive and unique. Finally,

\[
F(x_k^q) - y^\delta + J(x_k^q)p(\lambda) = \lambda(J(x_k^q)J(x_k^q)^T + \lambda I)^{-1}(F(x_k^q) - y^\delta),
\]

see e.g., \cite{Hanke2013} Proposition 2.1, and by \((3.1)\)

\[
q\| F(x_k^q) - y^\delta \| = \lambda_k^q\| J(x_k^q)J(x_k^q)^T + \lambda_k^q I \|^{-1}(F(x_k^q) - y^\delta)\|
\geq \frac{\lambda_k^q}{\| J(x_k^q)\|^2 + \lambda_k^q}\| F(x_k^q) - y^\delta \|,
\]
which yields \(3.6\).

Let now suppose that equation \(3.1\) has a solution and discuss how to solve it; to our knowledge, in literature this issue has not been addressed. Newton method is the most efficient procedure but in general it requires the knowledge of an accurate approximation to the solution. On the other hand, nonlinear equations with monotone and convex (or concave) functions on some interval containing the root are particularly suited to an application of Newton method, see e.g. [9, Theorem 4.8]. Equation \(3.1\) does not have such properties but it can be replaced by an equivalent equation with strictly decreasing and concave function on \([\lambda_k^q, \infty)\); thus, Newton method converges globally to \(\lambda_k^q\) whenever the initial guess overestimates such a root.

**Lemma 3.2.** Suppose \(\|F(x_k^q) - y^k\| \neq 0\), and that \(3.1\) has positive solution \(\lambda_k^q\). Let

\[
\psi(\lambda) = \frac{\lambda}{\|F(x_k^q) - y^k + J(x_k^q)p(\lambda)\|} - \frac{\lambda}{q\|F(x_k^q) - y^k\|} = 0.
\]

Then Newton method applied to \(3.6\) converges monotonically and globally to the root \(\lambda_k^q\) of \(3.7\) for any initial guess in the interval \([\lambda_k^q, \infty)\).

**Proof.** Trivially, solving \(3.1\) is equivalent to finding the positive root of the equation \(3.6\). We now show that \(\psi(\lambda)\) is strictly decreasing in \([\lambda_k^q, \infty)\) and concave on \((0, \infty)\). By \(2.14\),

\[
(3.7) \quad \frac{\lambda}{\|F(x_k^q) - y^k + J(x_k^q)p(\lambda)\|} = \left(\sum_{i=1}^{l} \left(\frac{r_i}{\zeta_i^2 + \lambda_k^q}\right)^2 + \sum_{i=l+1}^{n} \left(\frac{r_i}{\lambda_k^q}\right)^2\right)^{-1},
\]

and this function is concave on \((0, \infty)\), cfr. [3, Lemma 2.1]. Thus, \(\psi\) is concave on \((0, \infty)\) and trivially \(\psi'(\lambda)\) is strictly decreasing.

Now we show that \(\psi'(\lambda_k^q)\) is negative; thus, using the monotonicity of \(\psi'(\lambda)\), we get that \(\psi(\lambda)\) is strictly decreasing in \([\lambda_k^q, \infty)\). Differentiation of \(\psi(\lambda)\) and \(3.1\) give

\[
\psi'(\lambda_k^q) = \frac{(\lambda_k^q)^2}{\|F(x_k^q) - y^k + J(x_k^q)p(\lambda_k^q)\|^3} \left(\sum_{i=1}^{l} \frac{r_i^2}{(\zeta_i^2 + \lambda_k^q)^2} + \sum_{i=l+1}^{n} \frac{r_i^2}{\lambda_k^q}\right) - \frac{1}{q\|F(x_k^q) - y^k\|}.
\]

Moreover, using \(3.7\), it holds

\[
\psi'(\lambda_k^q) = -\frac{(\lambda_k^q)^2}{\|F(x_k^q) - y^k + J(x_k^q)p(\lambda_k^q)\|^3} \left(\sum_{i=1}^{l} \frac{r_i^2}{(\zeta_i^2 + \lambda_k^q)^2} + \sum_{i=1}^{l} \left(\frac{r_i}{\zeta_i^2 + \lambda_k^q}\right)^2\right)
\]

i.e. \(\psi'(\lambda_k^q)\) is negative.

The claimed convergence of Newton method follows from results on univariate concave functions given in [9, Theorem 4.8].

\[\Box\]
For the practical evaluation of \( \psi(\lambda) \) and \( \psi'(\lambda) \) we refer to [4, 15].

As shown in [7, 11], the Levenberg-Marquardt method combined with (3.1) is well defined and regularizing for (1.1) under the following assumptions on the solvability of the problem (1.1) and on the Taylor remainder of \( F \).

**Assumption 3.1.** Given an initial guess \( x_0 \), there exist positive \( \rho \) and \( c \) such that system (1.1) is solvable in \( B_\rho(x_0) \), and

\[
\| F(x) - F(\bar{x}) - J(x)(x - \bar{x}) \| \leq c\|x - \bar{x}\| \| F(x) - F(\bar{x}) \|, \quad x, \bar{x} \in B_{2\rho}(x_0).
\]

(3.8)

This assumption implies two relevant properties. First, there exists a unique \( x_0 \)-minimum-norm solution of (1.1) in \( B_\rho(x_0) \), see [11, Proposition 2.1]. Second, (3.4) is met for any \( x_k \) sufficiently close to a solution of (1.1) belonging to \( B_{2\rho}(x_0) \), see [7, Theorems 2.2, 2.3].

It remains an open problem how to preserve the properties of this Levenberg-Marquardt procedure in case the initial guess \( x_0 \) is a hint for some solution of (1.1) but it is not sufficiently close to guarantee the solvability of (3.1). As a first step, in order to allow more flexibility in the choice of \( \lambda_k \) we replace condition (3.1) with

\[
\| F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k \| \geq q\| F(x_k^\delta) - y^\delta \|,
\]

(3.9)

later denoted as the \( q \)-condition. This condition aims at coping with the case where (3.1) has not solution and tuning \( \lambda_k \) in view of global convergence.

We note that (3.9) was introduced in [7] as a viable alternative to (3.1) and used in [14] in the context of trust-region methods. However, a theoretical analysis of the Levenberg-Marquardt scheme employing (3.9) was not presented in [7]; for this reason in the following section we show its regularizing properties and point out some features that will be used thereafter.

### 3.1. Regularizing properties under the \( q \)-condition

We show that the regularizing properties of the Levenberg-Marquardt method are preserved under the \( q \)-condition provided that the initial guess \( x_0^\delta \) is sufficiently close to a solution of (1.1), and for any \( k \), \( \lambda_k \) is positive and bounded above by a scalar \( \bar{\lambda} \) independently of \( k \). As an intermediate result, we characterize the solutions of (3.9). The results given in this section are adaptations of those given in [7, 11]; the proofs are reported in Appendix for sake of completeness.

**Lemma 3.3.** Assume \( \| g_k \| \neq 0 \). Let \( p(\lambda) \) be the minimum norm solution of (2.3) with \( \lambda \geq 0 \) and \( P_k^\delta \) be the orthogonal projector onto \( \mathcal{R}(J(x_k^\delta))^\perp \). Then, equation (3.9) is satisfied for any \( \lambda \geq 0 \) whenever

\[
\| P_k^\delta(F(x_k^\delta) - y^\delta) \| \geq q\| F(x_k^\delta) - y^\delta \|.
\]

(3.10)

Otherwise, it is satisfied for any \( \lambda \geq \lambda_k^\delta \) where \( \lambda_k^\delta \) satisfies (3.5).

**Proof.** The claims easily follow from Lemma 3.1.

The next lemma shows the monotonic decay of the norm of the error, even in the presence of noise.

**Lemma 3.4.** Let Assumption 3.1 hold, \( x^\dagger \in B_\rho(x_0) \) be a solution of (1.1), \( x_k^\delta \) and \( x_{k+1}^\delta = x_k^\delta + p_k \) be two consecutive iterates of a Levenberg-Marquardt method, \( x_k^\delta \) and \( p_k \) satisfy (3.9). For noisy data, suppose \( k < k_* \) where \( k_* \) is defined in (1.3) and \( \tau > 1/q \).
Then, if \( x_k^\delta \) is sufficiently close to \( x^\dagger \) and belongs to \( B_{2\rho}(x_0) \), (3.4) holds for some \( \theta_k > 1 \), the step \( p_k = p(\lambda_k) \) corresponds to a strictly positive \( \lambda_k \), and

\[
\|x_k^\delta - x^\dagger\|^2 - \|x_{k+1}^\delta - x^\dagger\|^2 \geq \frac{2(\theta_k - 1)}{\theta_k \lambda_k} \|F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k\|^2.
\]

Now we can prove that the Levenberg-Marquardt method converges locally when applied to the unperturbed problem (1.1), while it finds a point satisfying (1.4) if the data are noisy. The following assumption on the initial guess is required.

**Assumption 3.2.** Let \( \rho \) as in Assumption 3.1, \( \tau > 1/q \) if the data are noisy, \( x^\dagger \) be a solution of (1.1) and \( x_0^\delta = x_0 \) satisfy

\[
\|x_0 - x^\dagger\| \leq \min\left\{ \frac{q}{c}, \rho \right\}, \quad \text{if} \quad \delta = 0,
\]

\[
\|x_0 - x^\dagger\| \leq \min\left\{ \frac{q\tau - 1}{c(1 + \tau)}, \rho \right\}, \quad \text{if} \quad \delta > 0.
\]

**Theorem 3.5.** Let Assumptions 3.1 and 3.2 hold and \( x_k^\delta \) be the iterates generated by a Levenberg-Marquardt method where (3.4) is satisfied. For noisy data, suppose \( k < k_* \) where \( k_* \) is defined in (1.4). Then, any iterate \( x_k^\delta \) belongs to \( B_{2\rho}(x_0) \).

Moreover, suppose that there exists \( \bar{\lambda} > 0 \) such that \( \lambda_k \in (0, \bar{\lambda}) \) at each iteration. With exact data, the sequence \( \{x_k\} \) converges to a solution of (1.1). With noisy data, the stopping criterion (1.4) is satisfied after a finite number \( k_* \) of iterations and \( \{x_k^\delta\} \) converges to a solution of (1.1) as \( \delta \) goes to zero.

### 4. Globally convergent procedures.

The goal of this paper is the construction of regularizing procedures that use the potential to approach a solution of (1.1) even if an accurate initial guess is not available. The trust-region and ARQ methods proposed in this section for problem (1.1) fulfill this requirement in the sense that,

- the value of \( \|F(x_k^\delta) - y^\delta\| \) is monotonically decreasing;
- with exact data, the sequence \( \{\|F(x_k) - y\|\} \) is convergent. If there exists an accumulation point of \( \{x_k\} \) which solves (1.1), then \( \lim_{k \to \infty} \|F(x_k) - y\| = 0 \), i.e. any accumulation point of \( \{x_k\} \) is a solution of (1.1);
- with noisy data, hopefully there exists a finite \( k_* \) such that discrepancy principle (1.4) is satisfied.

Algorithms 4.1 and 4.2 given below are regularizing versions of the trust-region and ARQ methods discussed in §2 and sketch a generic iteration \( k \). The standard procedures described in literature are modified so that the nonlinear stepsize control enforces both a reduction of \( \|F(x_k^\delta) - y^\delta\| \) and condition (3.9). This feature is obtained by suitable adaptations of the parameter \( \Delta_k \) in Algorithm 4.1 and \( \sigma_k \) in Algorithm 4.2. For the practical computation of steps we refer to [4] and [1] respectively.

We start our analysis of the regularizing procedures by verifying that the Algorithms 4.1, 4.2 are well-defined, i.e. the step \( p_k \) is found within a finite number of attempts. These results are obtained under the following assumption.

**Assumption 4.1.** There exists a positive constant \( \kappa_J \) such that

\[
\|J(x)\| \leq \kappa_J,
\]

for any \( x \) belonging to the level set \( \mathcal{L} = \{x \in \mathbb{R}^n \ s.t. \ \Phi(x) \leq \Phi(x_0)\} \).

The next two results show that the procedures given are well-defined.


Algorithm 4.1: Regularizing Trust-Region method for problem (1.3)

Given an initial point $x_0$ and $\Delta_0, \Delta_{\text{max}} > 0$, $1 > \eta_2 > \eta_1 > 0$, $\gamma_3 \in (0, 1)$.

Exact-data: given $\delta = 0$, $q \in (0, 1)$, $\gamma_2 > \gamma_1 > 1$.

Noisy-data: given $\delta > 0$, $\tau > 1$, $\gamma_1 \in (1, \tau)$, $\gamma_2 > \gamma_1$.

1. If $\delta > 0$ then $q = \gamma_1 \tau$.
2. Repeat
   2.1 Compute the solution $p_k$ of the trust-region problem (2.4).
   2.2 Compute $\rho_k$ given in (2.6), (2.7), (2.8), and $\zeta_k = \| F(x_\delta^k) - y^\delta + J(x_\delta^k)p_k \| / \| F(x_\delta^k) - y^\delta \|$
   2.3 If $\rho_k < \eta_1$ or $\zeta_k < q$ then set $\Delta_k = \gamma_3 \Delta_k$.

Until $\rho_k \geq \eta_1$ and $\zeta_k \geq q$.

3. Set $x_{k+1} = x_k + p_k$.
4. If $\rho_k \geq \eta_2$ and $\zeta_k \geq \gamma_2 q$ set $\Delta_{k+1} \in [\Delta_k, \Delta_{\text{max}}]$.
   else set $\Delta_{k+1} = \Delta_k$.

Lemma 4.1. Suppose that Assumption 4.1 holds. If $\| g_k \| \neq 0$ then the repeat-loop at Step 2 of Algorithm 4.1 terminates.

Proof. Let first enforce the $q$-condition. It is trivially satisfied if (3.10) is met. Otherwise, by Lemma 3.3 the monotonic decrease of $\| p(\lambda) \|$ and Assumption 4.1 it holds whenever $\Delta_k \leq \rho \left( \frac{q}{1 - q} \right)^\frac{\Delta_0}{\| g_k \|}$, i.e. after a finite number of reductions of $\Delta_k$.

The proof is completed since it is well-known that condition $\rho_k \geq \eta_1$ is enforced if the trust-region radius is small enough, see e.g. [4].

Lemma 4.2. Suppose that Assumption 4.1 hold. If $\| g_k \| \neq 0$ then the repeat-loop at Step 2 of Algorithm 4.1 terminates.

Proof. By Lemma 3.3 the condition $\zeta_k \geq q$ is satisfied within a finite number of steps. In fact, it is satisfied either for all $\lambda \geq 0$ or for $\lambda \geq \lambda_k^q$ with $\lambda_k^q$ in (3.5). By [1] Lemma 3.4

$$\| p_k \| \leq \frac{2\| g_k \|}{\sigma_k \| F(x_\delta^k) - y^\delta \|},$$

and since $\| p(\lambda) \|$ is monotonic decreasing, we get that $\lambda \geq \lambda_k^q$ holds whenever

$$\sigma_k \geq \frac{2\| g_k \|}{\| F(x_\delta^k) - y^\delta \| \| p(\lambda_k^q) \|}.$$

This implies that the $q$-condition is met within a finite number of increases of the parameter $\sigma_k$.

The proof that condition $\rho_k \geq \eta_1$ is enforced in a finite number of repetitions is given in [1] Theorem 3.6.
Algorithm 4.2: Regularizing ARQ method for problem (1.3)

Given an initial point \( x_0, \mu > 0, \sigma_0 > 0, \delta \geq 0, 1 > \eta_2 > \eta_1 > 0 \).

1. If \( \delta > 0 \), then \( q = \frac{\gamma_1}{\tau} \).
2. Repeat
   2.1 Compute the solution \( p_k \) of (2.9).
   2.2 Compute \( \rho_k \) given in (2.6), (2.11), (2.12), and \( \zeta_k = \| F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k \| \) divided by \( \| F(x_k^\delta) - y^\delta \| \).
   2.3 If \( \rho_k < \eta_1 \) or \( \zeta_k < q \), set \( \sigma_k = \gamma_1 \sigma_k \).

Until \( \rho_k \geq \eta_1 \) and \( \zeta_k \geq q \).
3. Set \( x_{k+1} = x_k + p_k \)
4. If \( \rho_k \geq \eta_2 \) and \( \zeta_k \geq \gamma_2 q \)
   set \( \sigma_{k+1} \in (0, \sigma_k] \),
   else
   set \( \sigma_{k+1} = \sigma_k \).

Global convergence results of the trust-region and ARQ methods are given in the following theorem; we refer to [19, Theorem 11.9] and [1, Theorem 3.8] for the proof.

We observe that the Lipschitz continuity of \( J \) assumed below is used also in the paper [10].

**Theorem 4.3.** Suppose that Assumption 4.1 holds and \( J \) is Lipschitz continuous on \( \mathbb{R}^n \). Then, both the sequences \( \{ x^\delta_k \} \) generated by Algorithms 4.1 and 4.2 satisfy

\[
\lim_{k \to \infty} \nabla \Phi(x^\delta_k) = \lim_{k \to \infty} \| J(x^\delta_k)^T (F(x^\delta_k) - y^\delta) \| = 0.
\]

By construction, the sequence \( \| F(x^\delta_k) - y^\delta \| \) is monotonically decreasing and bounded below by zero; hence it is convergent. Equation (4.3) implies that any accumulation point of the sequence \( \{ x^\delta_k \} \) is a stationary point of \( \Phi \). As for exact data, we conclude that if there exists an accumulation point of \( \{ x_k \} \) solving (1.1), then any accumulation point of the sequence solves (1.1). In the case of noisy data, if the value of \( \Phi \) at some accumulation point of \( \{ x^\delta_k \} \) is below the scalar \( \tau \delta \), then there exists an iterate \( x^\delta_k \) such that the discrepancy principle is met.

It remains to show the behaviour of the sequences generated by Algorithms 4.1 and 4.2 when, for some \( k \), \( x^\delta_k \) is sufficiently close to a solution \( x^\dagger \) of (1.1). For instance, this occurs with exact data when the accumulation points of \( \{ x_k \} \) solve (1.1) and \( k \) is sufficiently large. In the next section we show that the trust-region and ARQ methods described in Algorithms 4.1 and 4.2 share the same local convergence properties as the regularizing Levenberg-Marquardt method.

5. **Local behaviour of the trust-region and ARQ methods.** We analyze the local properties of the trust-region and ARQ methods under the same assumptions made for the Levenberg-Marquardt method. Then, we suppose that, for some \( k \), there exists an iterate which plays the role of the initial guess \( x_0 \) in the Assumptions 3.1 and
To make the presentation uniform with [22] in the rest of the section we denote such an iterate as $x_0$. We remark that the assumptions used are weaker than those typically used in literature. In fact, to our knowledge, local properties of trust-region and ARQ strategies have been analyzed involving the inverse of $J$ and its upper bound in a neighbourhood of a solution, except for papers [14, 23] where the trust-region approach is investigated under Assumption 3.1.

The following theorem shows that locally the trust-region is active and Algorithm 4.1 is regularizing:

**Theorem 5.1.** Suppose that Assumptions 3.1, 3.2 and 4.1 hold. Let $x_k^\delta$ be an iterate generated by Algorithm 4.1 and for noisy data suppose that $k < k_*$, where $k_*$ is defined in (1.3).

(i) If $x_k^\delta$ is sufficiently close to $x^\dagger$ and belongs to $B_2p(x_0)$, then the step has the form $p_k = p(\lambda_k)$ with $\lambda_k > 0$.

(ii) If $x_k^\delta$ and $x_{k-1}^\delta$ belong to $B_2p(x_0)$, then there exists a constant $\bar{\lambda} > 0$ such that $\lambda_k \leq \bar{\lambda}$.

(iii) In the case of exact data, the sequence $\{x_k\}$ generated converges to a solution of (1.1).

(iv) In the case of noisy data, the stopping criterion (1.4) is satisfied after a finite number $k_*$ of iterations. Moreover the sequence $\{\bar{x}_k\}$ converges to a solution of (1.1) whenever $\delta$ goes to zero.

**Proof.** Item (i) follows directly from Lemma 3.4 (ii). Since the trust-region is active, by (2.8)

\[ \Delta_k = \|p_k\| = \|((B_k + \lambda_k I)^{-1} g_k\| \leq \|g_k\| \frac{\lambda_k}{r_k}. \]

Thus our claim follows if $\Delta_k/\|g_k\|$ is larger than a suitable threshold, independent from $k$. Let us provide such a bound.

We start analyzing for which values of $\Delta_k$, the $q$-condition (3.9) is satisfied. By Lemma 3.3 and the fact that $\|p(\lambda)\|$ is monotonically decreasing, (3.9) is satisfied when $\Delta_k = \|p_k\| \leq \|p(\lambda_k^2)\|$. Using (2.3), (3.5) and Assumption 4.1 we have

\[ \|p(\lambda_k^2)\| \geq \|g_k\| \frac{\|B_k + \lambda_k^2 I\|}{\kappa_J^2} \geq \frac{(1 - q)}{\kappa_J^2} \|g_k\|. \]

Then, the $q$-condition holds whenever $\Delta_k \leq \frac{(1 - q)}{\kappa_J^2} \|g_k\|$ and by the updating rule of $\Delta_k$ at Steps 2.3 and 4 of the algorithm, the repeat loop forces (3.9) with

\[ \Delta_k \geq \min \{\omega_1 \|g_k\|, \Delta_0\}, \]

where $\omega_1 = \gamma_1 \frac{1 - q}{\kappa_J^2}$.

Now, consider the condition $\rho_k > \eta_1$ and suppose that $\|F(x_k^\delta + p_k) - y^\delta\|^2 > m_k^{TR}(p_k)$, otherwise it holds $\rho_k > 1 > \eta_1$. Trivially,

\[ 1 - \rho_k = \frac{\|F(x_k^\delta + p_k) - y^\delta\|^2 - m_k^{TR}(p_k)}{\|F(x_k^\delta) - y^\delta\|^2 - m_k^{TR}(p_k)}, \]

and

\[ \|F(x_k^\delta + p_k) - y^\delta\|^2 - m_k^{TR}(p_k) \leq \|F(x_k^\delta + p_k) - F(x_k^\delta) - J(x_k^\delta)p_k\|^2 \]

\[ + 2\|F(x_k^\delta + p_k) - F(x_k^\delta) - J(x_k^\delta)p_k\| \|F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k\|. \]
By (3.8) and the mean value [19, Theorem 11.1], it holds
\[ \| F(x_\delta^k + p_k) - F(x_\delta^k) - J(x_\delta^k)p_k \| \leq c||p_k||\| F(x_\delta^k + p_k) - F(x_\delta^k) \| \leq c\kappa_J\|p_k\|^2, \]
and consequently
\[ \| F(x_\delta^k + p_k) - y^\delta \|^2 - m_k^{TR}(p_k) \leq c\kappa_J\Delta_k^2(\kappa_J\Delta_{\max}^2 + 2\| F(x_\delta^0) - y^\delta \|). \]

Theorem 6.31 in [4] shows that
\[ \| F(x_\delta^k) - y^\delta \|^2 - m_k^{TR}(p_k) \geq \frac{1}{2}\| g_k \| \min\{\Delta_k, \frac{\| g_k \|}{\| B_k \|}\}. \]

Then,
\[ \| F(x_\delta^k) - y^\delta \|^2 - m_k^{TR}(p_k) \geq \frac{1}{2}\Delta_k\| g_k \|, \]
whenever \( \Delta_k \leq \frac{\| g_k \|}{\kappa_J^2} \) and this implies
\[ 1 - \rho_k \leq \frac{2c\kappa_J\Delta_k(\kappa_J\Delta_{\max}^2 + 2\| F(x_\delta^0) - y^\delta \|)}{\| g_k \|}. \]

Namely, termination of the repeat loop occurs with
\[ \Delta_k \leq \| g_k \| \min\left\{ \frac{1}{\kappa_J^2}, \frac{1 - \eta_1}{2c\kappa_J(\kappa_J\Delta_{\max}^2 + 2\| F(x_\delta^0) - y^\delta \|)} \right\}. \]
and, again by the updating rule at Steps 2.3 and 4, and by (5.2), \( \Delta_k \) is guaranteed to satisfy
\[ \Delta_k \geq \min\{ \omega_1\| g_k \|, \omega_2\| g_k \|, \Delta_0 \}, \]
where \( \omega_2 = \gamma_3 \min\left\{ \frac{1}{\kappa_J}, \frac{1 - \eta_1}{c\kappa_J(\kappa_J\Delta_{\max}^2 + 2\| F(x_\delta^0) - y^\delta \|)} \right\}. \]
Finally, by (5.1)
\[ \lambda_k \leq \frac{\| g_k \|}{\Delta_k} \leq \max\left\{ \frac{1}{\omega_1}, \frac{1}{\omega_2}, \frac{\kappa_J\| F(x_\delta^0 - y^\delta \|)}{\Delta_0} \right\}, \]
which concludes the proof.

Items (iii) and (iv) are a direct consequence of Theorem 5.2.

We conclude our theoretical analysis with the study of ARQ method implemented as in Algorithm 4.2.

**THEOREM 5.2.** Suppose that Assumptions 3.1, 3.2 and 4.1 hold. Let \( x_\delta^k \) be an iterate generated by Algorithm 4.2 and for noisy data suppose that \( k < k_* \), where \( k_* \) is defined in (5.1). Then

(i) If \( x_\delta^k \) and \( x_\delta^{k+1} \) belong to \( B_{2\rho}(x_\delta^0) \), the step has the form \( p_k = p(\lambda_k) \) with \( \lambda_k \in (0, \bar{\lambda}] \), for some positive \( \lambda \);

(ii) in the case of exact data, the sequence \( \{x_k\} \) generated converges to a solution of (1.1).
(iii) in the case of noisy data, the stopping criterion (1.4) is satisfied after a finite number \(k_*\) of iterations. Moreover the sequence \(\{x_k^j\}\) converges to a solution of (1.1) whenever \(\delta\) goes to zero.

**Proof.** (i) We know that the minimizer \(p_k = p(\lambda_k)\) of (2.9) solves (2.3) and (5.4)

\[
\lambda_k \in [\mu, \mu + 2\sigma_k\|F(x_k^j) - y^\delta\|],
\]

[Lemma 4.3]. Hence, the scalars \(\lambda_k\) are positive and for their uniform boundness, it suffices to show that \(\sigma_k\|F(x_k^j) - y^\delta\|\) is bounded from above. We now prove this fact.

Using (4.2), the updating rule at Steps 2.3 and 4 guarantees the \(q\)-condition when

\[
\sigma_k\|F(x_k^j) - y^\delta\| \leq \max \left\{2\gamma_1 \frac{|g_k|}{\|p(\lambda_k^j)\|}, \sigma_0 \right\}.
\]

Hence by (2.8), (5.5) and Assumption 4.1

\[
\frac{|g_k|}{\|p(\lambda_k^j)\|} \leq \|B_k + \lambda_k^j I\| \leq \frac{1}{1 - q\kappa^j},
\]

and

\[
\sigma_k\|F(x_k^j) - y^\delta\| \leq \max \left\{\gamma_1 \frac{2\kappa_j^2}{1 - q}, \sigma_0 \right\}.
\]

Concerning condition \(\rho_k \geq \eta_1\), trivially it is satisfied if \(\|F(x_k^j + p_k) - y^\delta\| \leq \|F(x_k^j) - y^\delta\| + J(x_k^j)p_k\) as this implies \(\|F(x_k^j + p_k) - y^\delta\| \leq m_k^{\text{ARQ}}(p_k).\) Otherwise,

\[
\|F(x_k^j + p_k) - y^\delta\| \leq \|F(x_k^j + p_k) - F(x_k^j) - J(x_k^j)p_k\| + \|F(x_k^j) - y^\delta\| + J(x_k^j)p_k\|
\]

and trivially

\[
\|F(x_k^j + p_k) - y^\delta\| - m_k^{\text{ARQ}}(p_k) \leq \|F(x_k^j) - F(x_k^j) - J(x_k^j)p_k\|.
\]

Moreover, by (5.5) and (4.1)

\[
\|F(x_k^j + p_k) - y^\delta\| - m_k^{\text{ARQ}}(p_k) \leq \frac{4\kappa_1\|g_k\|^2}{\sigma_k^2\|F(x_k^j) - y^\delta\|^2}.
\]

Lemma 3.3 of [1] gives

\[
\|F(x_k^j) - y^\delta\| - m_k^{\text{ARQ}}(p_k) \geq \frac{|g_k|^2}{4\|F(x_k^j) - y^\delta\|} \min \left\{\frac{1}{2\sigma_k\|F(x_k^j) - y^\delta\|}, \frac{1}{\|B_k + \mu I\|}\right\}
\]

Then, we get

\[
\|F(x_k^j) - y^\delta\| - m_k^{\text{ARQ}}(p_k) \geq \frac{|g_k|^2}{8\sigma_k\|F(x_k^j) - y^\delta\|^2},
\]

as soon as

\[
\sigma_k \geq \frac{\|B_k + \mu I\|}{2\|F(x_k^j) - y^\delta\|}
\]

(5.7)
Combining the above inequalities, we obtain
\[ 1 - \rho_k = \frac{\|F(x_k^\delta + p_k) - y^\delta\| - m^\text{ARQ}_k(p_k)}{\|F(x_k^\delta) - y^\delta\| - m^\text{ARQ}_k(p_k)} \leq \frac{32\kappa J}{\sigma_k}, \]
whenever (5.7) holds. Then, condition \( \rho_k > \eta_1 \) is met whenever
\[ \sigma_k \geq \max \left\{ \frac{\|B_k + \mu I\|}{2\|F(x_k^\delta) - y^\delta\|}, \frac{32\kappa J}{1 - \eta_1}, \sigma_0 \right\}. \]

Applying the updating rule for \( \gamma_k \) at Steps 2.3 and 4 we have
\[ \sigma_k \leq \max \left\{ \gamma_1 \frac{\|B_k + \mu I\|}{2\|F(x_k^\delta) - y^\delta\|}, \gamma_1 \frac{32\kappa J}{1 - \eta_1}, \sigma_0 \right\}, \]
and by (5.6) and the monotonic decrease of \( \|F(x_k^\delta) - y^\delta\| \),
\[ \sigma_k \|F(x_k^\delta) - y^\delta\| \leq \max \left\{ \gamma_1 \frac{2\kappa_J^2}{1 - q}, \gamma_1 \frac{\kappa_J^2 + \mu}{2}, \max \left\{ \gamma_1 \frac{32\kappa J}{1 - \eta_1}, \sigma_0 \right\} \right\} \|F(x_0) - y^\delta\|. \]

This completes the proof.

Items (ii) and (iii). The regularizing properties of ARQ readily follow from Theorem 3.5.

6. Numerical results. In this section we report on the comparison between the regularizing Levenberg-Marquardt, trust-region and ARQ algorithms applied to ill-posed problems both with exact and noisy data.

Concerning the test problems, we have selected four nonlinear systems arising from the discretization of nonlinear Fredholm integral equations of the first kind
\[ \int_0^1 k(t, s, x(s))ds = y(t), \quad t \in [0, 1]. \]

The integral equations considered model inverse problems from groundwater hydrology and geophysics. Their kernel is of the form
\[ k(t, s, x(s)) = \log \left( \frac{(t - s)^2 + H^2}{(t - s)^2 + (H - x(s))^2} \right), \]
see [21, §3], or
\[ k(t, s, x(s)) = \frac{1}{\sqrt{1 + (t - s)^2 + x(s)^2}}. \]

see [10, §6]. The interval [0, 1] was discretized with \( n = 20 \) equidistant grid points \( t_i = (i - 1)h, \quad h = 1/(n - 1), \quad i = 1, \ldots, n \). Function \( x(s) \) was approximated from the \( n \)-dimensional subspace of \( H_0^1(0, 1) \) spanned by standard piecewise linear functions.

Specifically, we let \( s_j = (j - 1)h, \quad h = 1/(n - 1), \quad j = 1, \ldots, n, \) and looked for an approximation \( \hat{x}(s) = \sum_{j=1}^n \hat{x}_j \phi_j(s) \) where
\[ \phi_1(s) = \begin{cases} \frac{s_2 - s}{h} & \text{if } s_1 \leq s \leq s_2, \\ 0 & \text{otherwise} \end{cases}, \quad \phi_n(s) = \begin{cases} 0 & \text{if } s_{n-1} \leq s \leq s_n, \\ \frac{s - s_{n-1}}{h} & \text{otherwise} \end{cases}, \]

\[ \phi_{n-1}(s) = \begin{cases} \frac{s_2 - s}{h} & \text{if } s_1 \leq s \leq s_2, \\ 0 & \text{otherwise} \end{cases}. \]
components are the value of the true solution at \( s \) in systems arising from their discretizations are denoted as [10, p. 660], and the discontinuous function in [10, p. 662] respectively; the nonlinear function \( F \) were implemented according to Algorithms 4.1 and 4.2.\]

(3.9) in order to avoid declaring a breakdown. The trust-region and ARQ procedure \( \lambda \) did not exist, we set \( \tau \) in case of exact data, and the discrepancy principle (1.4) with (6.4) was set equal to \( 10^{-1} \).\]

As for the Levenberg-Marquardt approach, we implemented the strategy proposed by Hanke [7] and described in §3. In particular, \( \Delta_k \) was decreased when \( \Delta_k \geq 2 \). The discretized nonlinear systems associated to the above problems are named \( P_1 \) and \( P_2 \) respectively. Similarly, two problems with kernel (6.3) were generated so that the solution is the constant function \( x(s) = 1 \), \( s \in [0, 1] \), see [10] p. 660, and the discontinuous function in [10] p. 662 respectively; the nonlinear systems arising from their discretizations are denoted as \( P_3 \) and \( P_4 \). The vector whose components are the value of the true solution at \( s_j \), \( 1 \leq j \leq n \), will be later denoted as \( x^\dagger \).

All procedures were implemented in MATLAB and run using MATLAB 2013B on an Intel Core(TM) i5-24667 1.6 GHz, 4 GB RAM; the machine precision is \( \epsilon_m \sim 2 \cdot 10^{-16} \). As for the Levenberg-Marquardt approach, we implemented the strategy proposed by Hanke [7] and described in §3, but if, at a certain iteration \( k \), the solution of (3.1) did not exist, we set \( \lambda_k = \lambda_{k-1} \). This amounted to replace (3.1) with \( q \)-condition (3.9) in order to avoid declaring a breakdown. The trust-region and ARQ procedure were implemented according to Algorithms 4.1 and 4.2.

The major implementation issues are as follows. The Jacobian of the nonlinear function \( F \) was computed by finite differences. The parameter \( q \) used in (3.1) and in (3.9) was set equal to \( 10^{-1} \) if the data are exact and equal to \( 1/\tau \) otherwise. As for the stopping criterion, we used (6.4) in case of exact data, and the discrepancy principle (1.4) with \( \tau = 1.5 \) for noisy problems. We chose \( \text{to1} = 10^{-3} \) for problems \( P_1 \) and \( P_2 \) and \( \text{to1} = 10^{-6} \) for problems \( P_3 \) and \( P_4 \); a tighter tolerance is needed in the latter problems in order to obtain a sufficiently accurate approximation to the solution. A maximum number of 200 iterations was allowed and a failure (below indicated as \( F_{H} \)) was declared when this limit was exceeded. In case of noisy problems, given the error level \( \delta \), the exact data \( y \) was perturbed by normally distributed values with mean 0 and variance \( \delta \) using the MATLAB function \texttt{randn}.

Algorithms 4.1 was run setting \( \eta_1 = 10^{-1} \), \( \eta_2 = 0.75 \), \( \gamma_2 = 1.1 \), \( \Delta_0 = 10^{-1} \), \( \Delta_{\text{max}} = 10^{4} \). The updating rules for the trust-region radius are based on standard technicalities, see [11][§6.1] and [19][§11.2]. In particular, \( \Delta_k \) was decreased by setting \( \Delta_k = \min\{\Delta_k/2, \|p_k\|/2\} \), and increased by setting \( \Delta_{k+1} = 1.75\Delta_k \). A failure of the algorithm was declared when \( \Delta_k \) reduces below \( 10^{-12} \) (below this failure is indicated as \( F_{\Delta} \)).

Algorithms 4.2 was run setting \( \mu = 10^{-7} \), \( \eta_1 = 0.1 \), \( \eta_2 = 0.9 \), \( \gamma_1 = 2 \). The parameter \( \sigma_k \) is decreased when \( p_k \geq \eta_2 \) and \( \zeta_k > q \) by means of the rule \( \sigma_{k+1} = \max\{\sigma_k/2, \sqrt{\epsilon_m}\} \).
Concerning the computation of the parameter \( \lambda_k \), it amounts to the solution of a suitable scalar equation, say \( \psi(\lambda) = 0 \), and we applied Newton method. Specifically, \( \psi \) takes the form (3.6) in the Levenberg-Marquardt approach, it is defined as

\[
\psi(\lambda) = \| p(\lambda) \| - \Delta_k,
\]

in the trust-region method, and it is the following convex reformulation of (2.10)

\[
\psi(\lambda) = \mu \lambda \lambda + 2 \sigma k \sqrt{\| F(x_k^0) - y^k + J(x_k^0)p_k \|^2 + \mu \| p_k \|^2} - 1,
\]

for the ARQ method [1, Lemma 4.4]. It is well known [4] that each Newton iteration applied to \( \psi(\lambda) = 0 \) requires the Cholesky factorization of a shifted matrix of the form \( B_k + \lambda I \), and typically high accuracy in the solution of the above scalar equations is not needed [1, 4]. Hence, we terminated the Newton process as soon as the absolute value of function \( \psi \) is below \( 10^{-1} \).

Our first set of experiments was made with exact data and tested the ability of the algorithms to converge from varying initial guesses, possibly far from the true solution. Table 6.1 displays the results obtained with five different initial guesses for each problem; here LM indicates the Levenberg-Marquardt procedure, TR indicates the trust-region algorithm, \( e = (1, \ldots, 1)^T \), and \( e_0 = \| x_0 - x^\dagger \| \) measures the distance of \( x_0 \) from the true solution \( x^\dagger \). Runs were classified as successful when (6.4) was satisfied and the number of iterations \( \text{iter} \) performed is shown. Moreover, we report the distance \( e_T = \| x_k - x^\dagger \| \) between the final \( x_k \) and the true solution \( x^\dagger \). The results are reported in italic when the sequence generated approaches the true solution while typewriter fonts are used when \( e_T \) appears to be large.

We first observe that the Levenberg-Marquardt failed on 12 tests out of 20, as well as the trust-region method, while ARQ never failed. Thus, the convergence of the ARQ method is remarkably enhanced by the globalization strategy. Among the successful runs, the true solution was not approached three times in the Levenberg-Marquardt and ARQ methods, twice in the trust-region method. As long as the Levenberg-Marquardt is successful, it is typically faster than the other two methods; this is reasonable as it succeeded for initial guesses close to the solution without the need to impose a monotonic decrease on \( \| F(x) - y \| \).

It is evident that the regularization induced by ARQ is very effective and the method outperforms the trust-region algorithm. All failures in the trust-region method occurred as the trust-region radius became excessively small and depend on the approximate computation of \( \lambda_k \) as a root of the nonlinear equation (6.5). In order to get more insight, in Table 6.2 we report the results obtained increasing the accuracy in such a computation, i.e. using the tighter tolerance \( 10^{-5} \) in the Newton procedure for (6.5); obviously, this increases the computational cost needed to compute each \( \lambda_k \). It can be observed that the number of failures drastically reduces to 2 but in 8 runs out of 18 successfully solved, the error \( e_T \) appears to be large. These failures occurred in correspondence to the initial guesses farther to the true solution. This occurrence can be explained as follows. The larger \( \Delta_k \) is, the smaller \( \lambda_k \) becomes, and the change of \( \lambda_k \) from one iteration to the other is not predictable even if the trust-region radius does not vary or changes slowly. As an example, in Figure 6.1 we report the values of \( \lambda_k \) at the first 10 nonlinear iterations of the trust-region and ARQ methods applied to problem P2 with \( x_0 = 2e \). We observe that in the ARQ approach the values of \( \lambda_k \) vary slowly, while in the trust-region method there is a drastic reduction of \( \lambda_k \) at
iteration 8. In particular at the end of the seventh iteration, the trust-region radius is increased from $\Delta_7 = 2.8723$ to $\Delta_8 = 5.0265$, and correspondingly $\lambda_k$ drops from order $10^0$ to $10^{-7}$. At such a stage, the regularizing effect of the trust-region is lost and the iterates do not further approach the true solution; in particular, the iterates seem to converge in the beginning of the iterative process and at the eight iteration they turn to diverge. Such an occurrence has not been detected in runs with the ARQ method and this is justified by (2.10) and (3.9) which yield $\lambda_k \geq \mu + 2\sigma_k q \|F(x) - y\|$. Therefore, the nonlinear residual affects the values of $\lambda_k$ and prevents small values in the initial phase of the iterative process.

For sake of completeness, in Figure 6.2 we plot the true solution and the solution computed by ARQ for all the four problems considered and with $\delta = 0$. The initial guess used is $x_0 = -2e$ for P1, $x_0 = 2e$ for P2, and $x_0 = 5e$ for P3 and P4.

In the rest of this section we focus on the case of noisy data and on the performance of the Levenberg-Marquardt and ARQ methods. In Table 6.3 we summarize the results obtained with $\delta = 10^{-2}$ and the same initial guesses as in Table 6.1. Besides the number of iterations $\text{iter}$ performed and the final error $e_T = \|x_k^\delta - x^\dagger\|$, we report the distance $e_I$ between the final iterate $x_k^\delta$ and $x^\dagger$ at the inner points $s_2, \ldots, s_{n-1}$, i.e. $e_I = \|x_k^\delta(2 : n-1) - x^\dagger(2 : n-1)\|$. The results on problem P4 are not reported because both methods succeeded from all the starting guesses, but the computed solutions were not approximations to $x^\dagger$.

We observe that both the errors $e_T$ and $e_I$ obtained with the Levenberg-Marquardt

| Problem | $x_0$ | $e_0$ | $\text{iter}$ | $e_T$ | $\text{iter}$ | $e_T$ | $\text{iter}$ | $e_T$ |
|---------|-------|-------|---------------|-------|---------------|-------|---------------|-------|
| P1      | 0 e   | 5.7e-2| 8             | 6.1e-3| 4             | 1.0e-2| 21            | 6.2e-3|
|         | -0.5 e| 4.6e-1| 15            | 1.8e-2| F\_\_\_       | 27    | 1.4e-2        |       |
|         | -1 e  | 9.6e-1| 15            | 2.1e+0| F\_\_\_       | 45    | 8.5e-3        |       |
|         | -2 e  | 2.0e+0| 37            | 5.6e-1| F\_\_\_       | 63    | 1.1e-2        |       |
|         | -3 e  | 3.0e+0| F\_\_\_       | 32    | 7.3e-1        | 101   | 8.4e-3        |       |
| P2      | 0 e   | 4.3e-1| 8             | 2.9e+0| 39            | 2.9e+0| 19            | 2.8e+0|
|         | 0.5 e | 1.3e-1| 13            | 9.0e-3| F\_\_\_       | 26    | 8.4e-3        |       |
|         | 1 e   | 6.0e-1| 21            | 4.6e-1| F\_\_\_       | 35    | 1.3e-2        |       |
|         | 2 e   | 1.6e+0| 16            | 2.9e+0| F\_\_\_       | 53    | 7.7e-3        |       |
|         | 3 e   | 2.6e+0| F\_\_\_       | 28    | 1.6e+0        | 74    | 2.0e+0        |       |
| P3      | 2 e   | 2.6e-1| F\_\_\_       | 47    | 1.4e-1        | 116   | 1.4e-1        |       |
|         | 3 e   | 3.9e-1| F\_\_\_       | 54    | 1.4e-1        | 154   | 1.4e-1        |       |
|         | 4 e   | 5.2e-1| F\_\_\_       | 107   | 1.4e-1        |       |               |       |
|         | 5 e   | 6.6e-1| F\_\_\_       | 123   | 1.4e-1        |       |               |       |
|         | 6 e   | 7.8e-1| F\_\_\_       | 130   | 8.5e+0        |       |               |       |
| P4      | 2 e   | 1.6e+0| F\_\_\_       | 24    | 4.0e-1        | 138   | 3.0e-1        |       |
|         | 3 e   | 2.5e+0| F\_\_\_       | 133   | 2.2e-1        |       |               |       |
|         | 4 e   | 3.5e+0| F\_\_\_       | 29    | 2.9e-1        | 140   | 1.4e-1        |       |
|         | 5 e   | 4.5e+0| F\_\_\_       | 164   | 1.4e-1        |       |               |       |
|         | 6 e   | 5.5e+0| F\_\_\_       | 179   | 1.4e-1        |       |               |       |

Table 6.1
Results obtained with exact data and varying initial guesses

17
Problem P1
\[ x_0 = 0 \varepsilon -0.5 \varepsilon - \varepsilon -2 \varepsilon -3 \varepsilon \]
iter 5 37 37 61 70
\[ e_T = 7.3 \varepsilon^{-3} 1.0 \varepsilon+0 8.4 \varepsilon^{-1} 1.2 \varepsilon+0 1.2 \varepsilon+0 \]

Problem P2
\[ x_0 = 0 \varepsilon 0.5 \varepsilon \varepsilon 2 \varepsilon 3 \varepsilon \]
iter 25 15 25 42 97
\[ e_T = 2.7 \varepsilon+0 2.2 \varepsilon-2 6.9 \varepsilon-1 1.2 \varepsilon+0 1.1 \varepsilon+0 \]

Problem P3
\[ x_0 = 2 \varepsilon 3 \varepsilon 4 \varepsilon 5 \varepsilon 6 \varepsilon \]
iter 50 84 94 \[ \tilde{F}_\text{it} \]
\[ e_T = 1.4 \varepsilon-1 3.4 \varepsilon+0 1.4 \varepsilon-1 \]

Problem P4
\[ x_0 = 2 \varepsilon 3 \varepsilon 4 \varepsilon 5 \varepsilon 6 \varepsilon \]
iter 25 26 59 148 132
\[ e_T = 3.3 \varepsilon-1 2.7 \varepsilon-1 2.1 \varepsilon-1 1.4 \varepsilon-1 4.0 \varepsilon+0 \]

Table 6.2

Trust-region method run with tighter tolerance $10^{-5}$ in the Newton procedure for (6.5).

Fig. 6.1. Behaviour of $\lambda_k$ in the first 10 iterations of trust-region and ARQ applied to Problem 2, with $\delta = 0$ and $x_0 = (2, 2, \ldots, 2)^T$. The trust-region method was run with tolerance $10^{-5}$ in the Newton procedure for (6.5).

method are large in six runs, while such an occurrence takes place once with ARQ method (problem P2, $x_0 = 0$). Except for this latter run, the solutions computed by ARQ method are accurate in the internal point $s_j$, i.e. in the interval $[h, 1-h]$. Concerning the error $e_T$, it is satisfactory in several runs and overall indicates that, if a major discrepancy between $x^\delta_k$ and $x^\dagger$ is present, then it is located near to the endpoints of the interval $[0, 1]$.

We conclude giving more details on some of the previous runs with ARQ method. For problem P2 with $\delta = 10^{-2}$, and $x^\delta_0 = 0.5 \varepsilon$ close to $x^\dagger$, Figure 6.3 displays the parameters $\lambda_k$ (on the left) and the norm of the errors $\|x^\delta_k - x^\dagger\|$ (on the right) versus the nonlinear iterations. We observe that the value of $\lambda$ varies slowly through the iterations and remains bounded below by $10^{-3}$ while the error monotonically
decays in accordance with the theoretical results in Theorem 5.2. The regularization properties of ARQ method are further supported in Figure 6.4 where, for problem P1 and $x_0 = -0.5e$, we display the computed solution and the true solution for decreasing level of noise $\delta = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$. As expected, the accuracy in the approximation increases as the noise level decreases. When $\delta = 10^{-2}, 10^{-3}$, the major discrepancy between the computed and true solution occurs at both ends of the interval where the solution is defined. It appears clearly that, as the noise decreases, the computed solution approaches the true one and for $\delta = 10^{-5}$ the two graphs are undistinguishable. A similar behaviour is observed with the trust-region algorithm when it is applied successfully.

7. Conclusions. We have presented regularized versions of trust-region and ARQ methods for nonlinear ill-posed systems, possibly with noisy data. They share the same local convergence properties as the regularizing Levenberg-Marquardt method proposed by Hanke in [7] but have enhanced convergence properties when the starting guess is not sufficiently close to a solution of the original system.

The numerical experience presented indicates that ARQ method outperforms both the regularizing Levenberg-Marquardt method and trust-region method. In particular, it regularizes the solution of the problems and may be successful from arbitrary starting guesses. On the contrary, we claim that a standard implementation of the trust-region method seems to be unreliable on the problems tested and specific updating rules for the trust-region radius should be devised in order to enforce the properties
| Problem | $x_0$ | iter | $e_T$ | $e_I$ | iter | $e_T$ | $e_I$ |
|---------|-------|------|-------|------|------|-------|------|
| P1      | 0 e   | 12   | 4.0e−2| 4.0e−2| 32   | 4.1e−2| 4.1e−2|
|         | -0.5 e| 16   | 4.3e−1| 9.1e−2| 43   | 4.3e−1| 9.1e−2|
|         | -1 e  | 19   | 6.8e−1| 1.5e−2| 51   | 8.3e−1| 1.5e−1|
|         | -2 e  | 19   | 2.1e+0| 2.1e+0| 78   | 6.3e−1| 1.4e−2|
|         | -3 e  | 19   | 2.3e+0| 2.1e+0| 111  | 8.0e−1| 1.5e−1|
| P2      | 0 e   | 16   | 2.9e+0| 2.9e+0| 39   | 2.9e+0| 2.9e+0|
|         | 0.5 e | 16   | 7.4e−2| 7.3e−2| 32   | 6.3e−2| 6.2e−2|
|         | 1 e   | 19   | 1.1e−1| 1.1e−1| 52   | 9.1e−2| 8.4e−2|
|         | 2 e   | 19   | 2.9e+0| 2.9e+0| 63   | 9.4e−2| 8.6e−3|
|         | 3 e   | 19   | 2.9e+0| 2.9e+0| 104  | 8.5e−2| 8.0e−2|
| P3      | 2 e   | 12   | 8.2e−1| 9.1e−1| 40   | 8.5e−1| 1.7e−1|
|         | 3 e   | 12   | 1.7e+0| 1.6e−1| 49   | 1.8e+0| 3.0e−1|
|         | 4 e   | 10   | 2.3e+0| 2.4e−1| 51   | 2.8e+0| 3.9e−1|
|         | 5 e   | 13   | 2.7e+0| 2.5e−1| 49   | 3.7e+0| 3.7e−1|
|         | 6 e   | 8    | 8.5e+0| 8.2e+0| 49   | 4.7e+0| 4.2e−1|

Table 6.3

Results obtained with noisy data, $\delta = 10^{-2}$, for varying initial guesses.

Fig. 6.3. Problem $P_2$, $\delta = 10^{-2}$, $x_0^\delta = (0.5, \ldots, 0.5)^T$. ARQ method: semilogarithmic plot of $\lambda_k$ vs iterations (left); error $\|x_k - x^\dagger\|$ vs iterations (right).

theoretically demonstrated.

Appendix. In this Appendix we prove Lemma 3.4 and Theorem 3.5.

Proof of Lemma 3.4. In the noise-free case suppose that $\|x^\dagger - x_k\| < \min \left\{ \frac{q}{c}, \rho \right\}$. Then, (3.4) yields $\theta_k = \frac{q}{c\|x^\dagger - x_k\|}$.

In presence of noisy data, let $\|x^\dagger - x_k^\delta\| < \min \left\{ \frac{q^\tau - 1}{c(1 + \tau)}, \rho \right\}$. By (3.8) and (1.2) we get

\[
\|y^\delta - F(x_k^\delta) - J(x_k^\delta)(x^\dagger - x_k^\delta)\| \leq \delta + \|y - F(x_k^\delta) - J(x_k^\delta)(x^\dagger - x_k^\delta)\| \\
\leq \delta + c\|x^\dagger - x_k^\delta\| \|y - F(x_k^\delta)\|
\]
Then, at any iteration $k < k^*$, condition (1.4) gives

$$
\| y^\delta - F(x^\delta_k) - J(x^\delta_k)(x^\dagger - x^\delta_k) \| \leq \left( 1 + c \| x^\dagger - x^\delta_k \| \right) \| y^\delta - F(x^\delta_k) \|.
$$

and (3.9) yields to (3.4) with 

$$
\theta_k = \frac{q}{1 + c(1 + \tau) \| x^\dagger - x^\delta_k \|}
$$

Thus, by Lemma 3.3 the step is $p_k = p(\lambda_k)$ for some strictly positive $\lambda_k$. Finally, the proof of (3.11) is a straightforward adaptation of [11, Proposition 4.1] for equation (4.6) therein.

**Proof of Theorem 3.5.** First suppose that the data are exact. By Assumption 3.2, Lemma 3.4 implies that condition (3.4) is satisfied at $k = 0$ with $\theta_0 = \frac{q}{c \| x_0 - x^\dagger \|}$. Further, it holds $\| x_1 - x^\dagger \| < \| x_0 - x^\dagger \|$ and consequently $x_1 \in B_{2\rho}(x_0)$. Repeating the above arguments, by Lemma 3.4 condition (3.4) holds at $k = 1$ with 

$$
\theta_1 = \frac{q}{c \| x_1 - x^\dagger \|} > \theta_0,
$$

and consequently $\| x_2 - x^\dagger \| < \| x_1 - x^\dagger \|$. Thus, by induction, $\{ \| x_k - x^\dagger \| \}$ is monotonic decreasing and $\{ \theta_k \}$ is monotonic increasing.

Since the function $(\theta - 1)/\theta$ is monotonic increasing it follows $(\theta_k - 1)/\theta_k >
Therefore, following the lines of the proof of Theorem 4.2 in [11], we can conclude that \( \{x_k\} \) is a Cauchy sequence, i.e., it is convergent.

Finally, by (7.1), \( \lambda_k < \bar{\lambda} \) and (3.11)

\[
\|x_k^\delta - x^\dagger\|^2 - \|x_{k+1}^\delta - x^\dagger\|^2 \geq \frac{2(\theta_0 - 1)q^2}{\theta_0}\|F(x_k^\delta) - y^\delta + J(x_k^\delta)p_k\|^2.
\]

Hence \( \|F(x_k) - y\| \) tends to zero and the limit of \( x_k \) has to be a solution of (1.1).

Let now consider noisy data. By using the above induction argument, we get that \( \|x_k^\delta - x^\dagger\| \) is strictly decreasing and \( \theta_k \) is strictly increasing for \( k \leq k_* \). Summing up from 0 to \( k_* - 1 \), by (3.9) and (3.11) it follows

\[
k_* \tau^2 \delta^2 \leq \sum_{k=0}^{k_*-1} \|F(x_k^\delta) - y^\delta\|^2 \leq \frac{\theta_0 \bar{\lambda}}{2(\theta_0 - 1)q^2}\|x_0 - x^\dagger\|^2.
\]

Thus, \( k_* \) is finite for \( \delta > 0 \), and the claim follows from [7, Theorem 2.3] (see also [11, Theorem 4.3]). □

REFERENCES

[1] S. Bellavia, C. Cartis, N. I. M. Gould, B. Morini, and Ph. L. Toint, Convergence of a Regularized Euclidean Residual Algorithm for Nonlinear Least-Squares, SIAM Journal on Numerical Analysis, 48, pp. 1–29, 2010
[2] S. Bellavia and B. Morini, Strong local convergence properties of adaptive regularized methods for nonlinear least-squares, IMA Journal on Numerical Analysis, to appear.
[3] C. Cartis, N. I. M. Gould, and Ph. L. Toint, Trust-region and other regularizations of linear least-squares problems, BIT, 49, pp. 21–53, 2009.
[4] A.R. Conn, N.I.M. Gould, Ph.L. Toint, Trust-region methods, SMPS/SIAM Series on Optimization, 2000.
[5] C.V. Groetsch, The theory of Tikhonov regularization for Fredholm equations of the first kind, Pitman Advanced Publishing Program, Boston, 1984.
[6] G.N. Grapiglia, J. Yuan and Y. Yuan, On the convergence and worst-case complexity of trust-region and regularization methods for unconstrained optimization, Mathematical Programming, Series A, to appear.
[7] M. Hanke, Regularizing Levenberg-Marquardt scheme, with applications to inverse groundwater filtration problems, Inverse Problems, 13, pp. 79–95, 1997.
[8] M. Hanke, The regularizing Levenberg-Marquardt scheme is of optimal order, I. Integral Equations Applications, 22, pp. 259–283, 2010.
[9] P. Henrici, Elements of Numerical Analysis, J. Wiley and Sons, Chicester and New York, 1964.
[10] B. Kaltenbacher, Toward global convergence for strongly nonlinear ill-posed problems via a regularizing multilevel approach, Numerical Functional Analysis and Optimization, 27, pp. 637–665, 2006.
[11] B. Kaltenbacher, A. Neubauer, O. Scherzer, Iterative Regularization Methods for nonlinear ill-posed problems, Walter de Gruyter, Berlin, 2008.
[12] A. Kleefeld, M. Rebe, The Levenberg-Marquardt method applied to a parameter estimation problem arising from electrical resistivity tomography, Applied Mathematics and Computation, 217, pp. 4490–4501, 2011.
[13] K. Levenberg, A method for the solution of certain nonlinear problems in least-squares, Quarterly Applied Mathematics, 2, pp. 164–168, 1944.
[14] G.D. Li, Y. Wang, A Regularizing Trust Region Algorithm for Nonlinear Ill-posed Problems, Inverse Problems in Science and Engineering, 14, pp. 859 – 872, 2006.
[15] D. Marquardt, An Algorithm for least-squares estimation of nonlinear parameters, SIAM Journal Applied Mathematics, 11, pp. 431–441, 1963.
[16] J.J. Moré, *The Levenberg-Marquardt algorithm: implementation and theory*, Proc. 7th Biennial Conf., Univ. Dundee, Dundee, 1977, pp. 105–116. Lecture Notes in Math., Vol. 630, Springer, Berlin, 1978.

[17] V.A. Morozov, *On the solution of functional equations by the method of regularization*, Soviet Mathematics Doklady, 7, pp. 414–417, 1966.

[18] Y. Nesterov, *Modified Gauss-Newton scheme with worst-case guarantees for global performance*, Optimization Methods and Software, 22, pp. 469–483, 2007.

[19] J. Nocedal, S.J. Wright, *Numerical Optimization*, Springer Series in Operations Research, 1999.

[20] Ph.L. Toint, *Nonlinear stepsize control, trust regions and regularizations for unconstrained optimization*, Optimization Methods and Software, 28, pp. 82–95, 2013.

[21] C. R. Vogel, *A constrained least squares regularization method for nonlinear ill-posed problems*, Siam J. Control and Optimization, 28, pp. 34–49, 1990.

[22] C. R. Vogel, *Computational methods for inverse problems*, SIAM, Frontiers in Applied Mathematics, Providence, 2002.

[23] Y. Wang, Y. Yuan, *Convergence and regularity of trust region methods for nonlinear ill-posed problems*, Inverse Problems, 21, pp. 821–838, 2005.

[24] Y. Wang, Y. Yuan, *On the regularity of trust region-CG algorithm for nonlinear ill-posed inverse problems with application to image deconvolution problem*, Science in China Ser.A, 46, pp. 312–325, 2003.