UNIVERSITÀ DEGLI STUDI DI FIRENZE

Dipartimento di Matematica “U. Dini”

Dottorato di Ricerca in Matematica

Ciclo XXII

Trust-region quadratic methods for bound-constrained nonlinear least-squares and nonlinear feasibility problems

Settore Disciplinare: MAT/08, Analisi Numerica

Dottorando:
Margherita Porcelli

Direttore di Ricerca:
Prof. Benedetta Morini

Coordinatore del Dottorato:
Prof. Mario Primicerio

2009

Contents

Chapter 1

Introduction

In this thesis we address the numerical solution of two types of problems which are closely related: bound-constrained nonlinear least-squares problems and nonlinear feasibility problems.

This chapter is devoted to an introduction to the problems of interest and to the various questions that arise in the computation of their solutions. We present the problems under study and discuss characterizations of nonlinear feasibility problems via nonlinear constrained or unconstrained optimization problems. Also, we give an overview of the methods proposed in literature in recent years. We close the chapter summarizing the contents of the thesis.

1.1 Problem overview

We shall be concerned with two classes of problems with smooth functions. The first problem of interest is the *bound-constrained nonlinear least-squares problem* given by

$$\min_{x\in\Omega} \theta(x) = \frac{1}{2}\|\Theta(x)\|_2^2, \tag{BCLS}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ is a given continuously differentiable mapping and Ω is the n-dimensional box

$$\Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}, \tag{1.1}$$

with $l \in (\mathbb{R} \cup -\infty)^n$, $u \in (\mathbb{R} \cup \infty)^n$, $l < u$. The constraints (1.1) are called *box constraints*, *simple bounds* or *bounds*.

Provided that problem (BCLS) has a solution at which Θ is null, then such solution solves the nonlinear system of equations $\Theta(x) = 0$, $x \in \Omega$. Vice versa, the bound-constrained and possibly nonsquare system of nonlinear equations

$$F(x) = 0, \quad x \in \Omega, \tag{NE}$$

with $F : \mathbb{R}^n \to \mathbb{R}^m$, can be transformed into problem (BCLS) letting $\Theta = F$.

The second problem considered is the nonlinear *feasibility problem* stated as a system of nonlinear equalities and inequalities of the form

$$\begin{aligned} C_E(v) &= 0, \\ C_I(v) &\le 0, \\ v_l \le v &\le v_u, \end{aligned} \tag{FP}$$

where the vector functions $C_E : \mathbb{R}^p \to \mathbb{R}^{m_E}$ and $C_I : \mathbb{R}^p \to \mathbb{R}^{m_I}$ are continuously differentiable, $v_l \in (\mathbb{R} \cup -\infty)^p$, $v_u \in (\mathbb{R} \cup \infty)^p$. If $m_I = 0$, the problem (FP) reduces to a bound-constrained system of nonlinear equations.

These problems appear frequently in practice. They occur in many contexts such as model formulation design, parameter identification problems, statement of Karush-Kuhn-Tucker conditions and detection of approximately feasible points in nonlinear programming in which the constraints are a mixture of general and box constraints, see e.g. [5, 16, 24, 25, 33, 41]. Problem (FP) may also occur as a subproblem in the "restoration" phase arising in filter methods for nonlinear programming problems, see e.g. [21, 34, 65, 67].

Taking into account the variety of applications yielding the problems considered, we allow any relationship between the dimensions m and n in (BCLS) and (NE) and the dimensions m_E, m_I and p in (FP).

We assume the presence of simple bounds in (BCLS), (NE) and (FP) as it is fairly common to have restrictions on the expected size of each variable, [24, 29]. In some situations, the presence of box constraints can specify either the domain of the mappings or prevent the computation of solutions which have no physical meaning. As an example, chemical equilibrium problems are modelled as problems (NE) and the concentrations of chemical species must be nonnegative, [64]. Furthermore, it is often helpful to introduce unnecessary but reasonable bounds on the variables when there is a good guess of the area where solutions are expected.

Solving problem (FP) consists in finding a vector $v \in \mathbb{R}^p$ which satisfies its equalities and inequalities. If such a point cannot be found, the goal is to minimize the sum of the constraint violations in (FP). Therefore, the solution of problem (FP) is typically attempted formulating it as a nonlinear least-squares problem.

The formulations proposed vary in the properties of the objective functions and in the possible presence of bounds on the variables. Specifically, one approach is to include all the equalities and inequalities in (FP) into the objective function and form the unconstrained least-squares problem

$$\min_{v \in \mathbb{R}^p} \frac{1}{2} \|\Phi(v)\|_2^2, \tag{1.2}$$

where $\Phi : \mathbb{R}^p \to \mathbb{R}^{m_E+m_I+r}$ and r, $0 < r \le 2p$ is a scalar that depends on the number of finite simple bounds, [16, 29, 34].

Alternatively, the original problem can be transformed into the bound-constrained nonlinear least-squares problem (BCLS) including the general inequalities $C_I(v) \le 0$ into the objective function and leaving the bounds in (1.1) unchanged; in this case the dimension n in (BCLS) is such that $n \ge p$ while $m = m_E + m_I$. Unlike problem (1.2), problem (BCLS) can be solved by means of bound-constrained algorithms where it is relatively straightforward to ensure that the objective function is computed at feasible iterates only. This may be considered a good feature of the reformulation as may serve as a check on the problem formulation and prevent evaluations at unreasonable or nonsensical points during the iterations.

1.2 Numerical methods

In this section we review the existing methods for solving bound-constrained nonlinear least-squares problems (BCLS), bound-constrained nonlinear systems (NE) and nonlinear feasibility problems (FP).

1.2.1 Bound-constrained least-squares problems and nonlinear systems

The major algorithms for solving bound-constrained nonlinear systems (NE) aim to solve them in a least-squares sense. Due to the close relation between these problems, numerical methods for (BCLS) have been proposed in the context of solving nonlinear and nonsquare systems (NE). In this section we describe procedures for problems (BCLS) and (NE) and pay particular attention to those that allow any relationship between the dimensions m and n of the mapping Θ and F.

The solution of an unconstrained square system of nonlinear equations, i.e. $\Omega = \mathbb{R}^n$ and $m = n$ in (NE), is a classical problem in mathematics for which many well-known solution techniques are available, see e.g. [17, 59]. On the other hand, the solution of constrained square systems of equations has not been the subject of intense research and we are currently aware of few papers that deal with such problems. Most of these papers appeared during the last few years; in particular the methods in [1, 2, 3, 4, 43, 44, 46, 50, 63, 71] handle box constraints while the methods given in [26, 55, 69] handle convex feasible sets. Note that papers [26, 46, 63, 71] consider nonsmooth, typically semismooth, equations.

In the remaining of the section, we focus on the papers for constrained nonsquare problems (NE) which are, as far as to our knowledge, [25, 47, 48, 73]. A common feature of these methods is that they attempt to solve (NE) by solving the optimization problem (BCLS). It follows the main characteristics of these methods.

Francisco et al. [25] designed a trust-region interior point method which adapts the method [1] for square systems to underdetermined nonlinear systems. The core of the method is the normal flow method for unconstrained underdetermined systems [70] which uses a Gauss-Newton model and the minimum norm step to generate a trial point. Such method is embedded into a trust-region strategy and the bounds are handled by using the Coleman-Li affine scaling matrix and a proper stepsize chopping rule [11]. Under full rank assumptions on the Jacobian matrix of F, local quadratic convergence is proved to interior points, i.e. points x^* such that $l < x^* < u$.

Kanzow et al. [48], Kanzow and Petra [47] and Zhu [73] proposed global Levenberg-Marquardt methods. In particular, the numerical solution of general nonsquare systems of nonlinear equations was addressed in [48, 73] while the paper [47] is focused on overdetermined nonsmooth systems of equations arising from a suitable reformulation of mixed complementarity problems. In [48] two Levenberg-Marquardt-type algorithms for nonlinear systems with convex constraints are presented. They differ in the way to compute the search directions: the first method solves a strictly convex minimization problem at each iteration, whereas the second one solves only one system of linear equations in each step. Both methods are shown to converge locally quadratically under an error bound assumption that is much weaker than the standard nonsingularity condition of the Jacobian matrix. A simple globalization strategy based on the projected gradient is employed. In [47] the Levenberg-Marquardt method is globalized by an affine trust-region procedure. The search direction is computed solving a trust-region subproblem where the constraint set is the box resulting from the intersection of the simple bounds and the current trust-region defined using the ∞-norm. Therefore, at each iteration a bound-constrained quadratic program is solved and the feasibility of the iterate is preserved. A variant of the Coleman-Li scaling matrix proposed in [71] is used in the globalization strategy. Moreover, a multidimensional filter technique is incorporated into the trust-region strategy to accept a full step more frequently. Global convergence is shown and local convergence is proved under an error bound assumption.

Finally, in [73] the Levenberg-Marquardt method is employed in association with the Coleman-Li scaling matrix [11] and a nonmonotone interior backtracking line search technique. A global convergence result is proved assuming that the function F in (NE) is semismooth and, under a local error bound condition, local fast convergence to non-degenerate solutions is achieved.

1.2.2 Nonlinear feasibility problems

The solution of systems of nonlinear equalities and inequalities (FP) was explicitly addressed in the papers [9, 13, 16, 21, 27, 34, 52, 53, 58]. In this section we discuss the methods proposed and give an insight into those based on trust-region strategies.

A Newton-type method for systems of mixed equalities and inequalities was given in [13], while global quadratic algorithms based on backtracking line search, were proposed in [9, 27]. The most recent methods are trust-region approaches [16, 21, 34, 52, 53, 58]. They are based on suitable transformations of the problem (FP) and vary widely from a computational point of view.

Fletcher and Leyffer [21] suggest to state the problem (FP) as a system of all inequalities by expressing the equalities $C_E(v) = 0$ as two separate inequalities $C_E(v) \leq 0$ and $-C_E(v) \leq 0$. Then they transformed the resulting problem into a bi-objective nonlinear programming problem. In particular, the inequalities are divided into two sets: the first set $J^\perp$ represents the inequalities which are close to being satisfied or for which the linearized inequality provides a good local model, the second set J is the complement of $J^\perp$. The nonlinear feasibility problem consists in the minimization of the constraint violations in the set J subject to the constraints in the set $J^\perp$. The problem obtained is solved by a filter trust-region SQP algorithm and the definition of the bi-objective problems is changed adaptively as the algorithm proceeds. The algorithm uses second order information and this fact yields fast local convergence and an efficient solution of locally infeasible problems. A proof of global convergence is provided.

Alternatively, all the inequalities in (FP) can be replaced by equalities and the problem takes the form of the least-squares problem (1.2) where the Euclidean norm of the constraint violations is minimized. In practice, the function $\Phi : \mathbb{R}^p \rightarrow \mathbb{R}^{m_E+m_I+2p}$ is given by

$$\Phi(v) = \begin{pmatrix} C_E(v) \\ \max\{C_I(v), 0\} \\ \max\{v_l - v, 0\} \\ \max\{v - v_u, 0\} \end{pmatrix},$$

where the maximum is taken componentwise; actually, note that some components of $\max\{v_l - v, 0\}$ and $\max\{v - v_u, 0\}$ can be eliminated if $v_l = -\infty$ or $v_u = \infty$. The mapping Φ is continuous but not continuously differentiable. This formulation of problem (FP) was adopted by Dennis et al. [16] and Gould and Toint [34].

In particular Dennis et al. [16] proposed two new trust-region algorithms: the first is a single model method, while the second is a multimodel algorithm where the Cauchy-point computation is a model selection procedure. A key ingredient of the methods is the use of an indicator matrix such that the problem can be transformed into one that possesses sufficient smoothness. This allows to develop algorithms that require differentiability. In practice the use of the indicator matrix gives rise to active-set type methods that try to identify the inequalities likely to be violated at a solution of (1.2). Global convergence for the two algorithms is proved.

Gould and Toint [34] considered the formulation (1.2) and proposed a filter trust-region method for finding a local minimizer of the problem. The procedure proposed is based on the algorithm given in [31] for nonlinear least-squares problems; such algorithm combines the efficiency of filter techniques and the robustness of the trust-region methods. In the trust-region strategy presented in [34] an adaptive model choice is used so that both the Gauss-Newton and the Full-Newton model are considered. Then the solution of the trust-region problem is computed approximately by using the Generalized Lanczos Trust-Region (GLTR) method [32] and results to be efficient independently of the dimension of the problem. The combination of the trust-region strategy and the filter strategy produces significant gains in reliability and efficiency compared to the standard trust-region approach. A Fortran 95 implementation of the proposed procedure is developed in the `FILTRANE` package.

Summarizing the properties of the methods given in [16, 21, 34], we point out that the methods in [16, 21] are globally convergent under appropriate assumptions while the technique used in [34] to handle the inequality constraints is heuristic and no theoretical guarantee of convergence can be provided for problems involving inequality constraints. None of these methods are supported by local convergence analysis.

Our contributions in solving problems (BCLS) and (FP) presented in this thesis are partly published in the papers [52, 53, 58].

1.3 Contents of the thesis

In this thesis we adopt problem (BCLS) as a unifying formulation for the problems under study. Therefore we develop and analyze trust-region methods with quadratic local convergence properties for solving the bound-constrained least-squares problem (BCLS). One of the algorithms proposed is implemented in a `Matlab` solver called `TRESNEI` that is robust and easy to use.

We adopt formulations of (FP) alternative to those proposed in the papers [16, 21, 34] discussed in Section 1.2.2. To transform the problem into a bound-constrained least-squares problem, we distinguish between the general inequalities $C_I(v) \leq 0$ and the simple bounds and replace the general inequalities by equalities. The resulting problem takes the form (BCLS) with a continuously differentiable function Θ.

The two new trust-region methods proposed offer global convergence properties combined with potentially fast local convergence. The procedures employ the Coleman-Li scaling matrix and differ in the quadratic model used during the iterations; one method employs a Gauss-Newton model while the other employs a regularized Gauss-Newton model. The implementations of the trust-region strategy involve a linear algebra phase; we discuss the use of matrix factorizations and CG-like methods for such phase.

From a computational point of view, the Gauss-Newton trust-region method resulted the most promising procedure. For this reason, it is coded into the `Matlab` implementation `TRESNEI`. The structure of `TRESNEI` handles the general statement of the nonlinear feasibility problem (FP) adopted in the CUTEr collection [33] and offers flexibility in the reformulation of problem (FP) as a bound-constrained least-squares problem.

The thesis is organized as follows. In Chapter 2 we give an overview on nonlinear least-squares problems. In Chapter 3 we present our statement of (FP) and introduce the new trust-region methods. Assuming that direct methods are used in the linear algebra phase, the trust-region methods are analyzed from both a theoretical and practical perspective; global and fast local convergence theory are provided and numerical performance is assessed. In Chapter 4 we provide a careful description of the solver `TRESNEI` and the results of the benchmarking process with functions from the `Matlab` Optimization Toolbox. In Chapter 5 an inexact paradigm for the Gauss-Newton trust-region method is proposed and its theoretical analysis is conducted. Finally, an Appendix is given where prerequisites to our study are summarized.

Notation

Throughout the thesis we use the following notation.

For a vector $x \in \mathbb{R}^n$ we denote by $(x)_i$ or x_i its i-th component and for any set of indices I, $[x]_I$ will denote the subvector of x with components x_i, $i \in I$. Similarly, the (i, j)-th element of a matrix $A \in \mathbb{R}^{m \times n}$ will be written as $(A)_{ij}$.

Given two vectors $x, y \in \mathbb{R}^n$ the inequalities $x \leq y$ and equalities $x = y$ are meant componentwise. Analogously, the vector $\max\{x, y\}$ is simply that whose i-th component is $\max\{x_i, y_i\}$.

Unless explicitly stated, $\|\cdot\|$ denotes the 2-norm and $B_\rho(y) = \{x : \|x - y\| < \rho\}$ denotes the open Euclidean ball of radius ρ around point $y \in \mathbb{R}^n$.

The symbols I_p represents the identity matrix of dimension p. A^+ denotes the Moore-Penrose pseudoinverse of the matrix A.

Given a sequence of vectors $\{x_k\}$, for any function f we let $f_k = f(x_k)$.

Given $\Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}$ we let $P_\Omega(x)$ be the projection of x onto Ω, i.e. $(P_\Omega(x))_i = \max\{l_i, \min\{x_i, u_i\}\}$, $i = 1, \ldots, n$.

We make frequent use of the Landau symbol $O(\cdot)$ which is defined as follows: given two sequences $\{\alpha_k\}$ and $\{\beta_k\}$ converging to zero as k tends to ∞, we write $\alpha_k = O(\beta_k)$ if $\text{limsup}_{k \to \infty} \alpha_k / \beta_k < \infty$.

Chapter 2

Nonlinear least-squares problems

In this chapter we review the main properties of nonlinear least-squares problems and discuss their numerical solution. The properties and algorithms presented will be the basis of the strategies proposed in this thesis for solving problems (BCLS) and (FP).

2.1 The unconstrained problem

Nonlinear least-squares problems belong to a special class of minimization problems where the function is a sum of squares of nonlinear functions

$$\min_{x\in\mathbb{R}^n} \theta(x) = \frac{1}{2}\|\Theta(x)\|^2, \tag{LS}$$

with $\theta : \mathbb{R}^n \to \mathbb{R}$ and $\Theta : \mathbb{R}^n \to \mathbb{R}^m$. We will refer to θ as the *objective function* and to Θ as the *residual function*. The vector function Θ is defined by

$$\Theta(x) = (\Theta_1(x), \Theta_2(x), \dots, \Theta_m(x))^T,$$

where the component functions $\Theta_i, i = 1, \dots, m$, are nonlinear real valued functions from $\mathbb{R}^n$ to $\mathbb{R}$. If $m > n$ we say the problem is *overdetermined*. If $m = n$ we have a *square* problem; if $m < n$ the problem is *underdetermined*.

We say that problem (LS) is unconstrained because no conditions are imposed on the variable x.

Let each $\Theta_i, i = 1, \dots, m$, be a smooth function. The Jacobian $J(x) \in \mathbb{R}^{m\times n}$ of the residual function $\Theta(x)$ is given by

$$(J(x))_{ij} = \frac{\partial\Theta_i(x)}{\partial x_j} \quad i = 1, \dots, m,\ j = 1, \dots, n,$$

and the Hessian $\nabla^2\Theta_i(x) \in \mathbb{R}^{n\times n}$ of Θ_i is given by

$$\left(\nabla^2\Theta_i(x)\right)_{jl} = \frac{\partial^2\Theta_i(x)}{\partial x_j\partial x_l}, \quad j, l = 1, \dots, n.$$

Then, the gradient and the Hessian of θ can be written as

$$\nabla\theta(x) = J(x)^T\Theta(x), \tag{2.1}$$

$$\nabla^2\theta(x) = J(x)^T J(x) + S(x), \quad S(x) = \sum_{i=1}^{m} \Theta_i(x)\nabla^2\Theta_i(x). \tag{2.2}$$

Solving problem (LS) means to find a *local minimizer* of θ, that is, a point x^* such that there exists a neighborhood $\mathcal{N}$ of x^* where $\theta(x^*) \leq \theta(x)$ for all $x \in \mathcal{N}$. Moreover, a point x^* is a *strict local minimizer* of θ if there exists a neighborhood $\mathcal{N}$ of x^* such that $\theta(x^*) < \theta(x)$ for all $x \in \mathcal{N}$ with $x \neq x^*$. In general local minimizers differ from *global minimizers* of θ, that is points where the function θ attains its least value over $\mathbb{R}^n$.

A local minimizer x^* for problem (LS) such that $\theta(x^*) = 0$ is a global minimizer of θ since $\theta(x) \geq 0$ for all $x \in \mathbb{R}^n$. It is denoted as a *zero-residual solution* and the problem (LS) is called a *zero-residual* problem. In this case, solving problem (LS) is equivalent to finding a solution of the nonlinear system $\Theta(x) = 0$. If $\theta(x^*)$ is small, the problem is called a *small-residual* problem. Otherwise one has a *large-residual* problem.

The following optimality conditions are the tools to recognize local minima.

Theorem 2.1 (First-Order Necessary Conditions) *([59]) If x^* is a local minimizer of θ and θ is continuously differentiable in an open neighborhood of x^*, then*

$$\nabla\theta(x^*) = J(x^*)^T \Theta(x^*) = 0. \tag{2.3}$$

We call x^* a *stationary point* for problem (LS) if it satisfies the equation (2.3). From Theorem 2.1, we have that a local minimizer must be a stationary point; vice versa, the converse is not true in general. To obtain further conditions for a local minimizer, we need stronger assumptions on the function θ.

Theorem 2.2 (Second-Order Necessary Conditions) *([59]) If x^* is a local minimizer of θ and $\nabla^2\theta$ is continuous in an open neighborhood of x^*, then x^* satisfies the equation (2.3) and $\nabla^2\theta(x^*)$ is positive semidefinite.*

It follows the sufficient conditions which guarantee that x^* is a local minimizer of θ.

Theorem 2.3 (Second-Order Sufficient Conditions) *([59]) Suppose that $\nabla^2\theta$ is continuous in an open neighborhood of x^*, x^* satisfies the equation (2.3) and $\nabla^2\theta(x^*)$ is positive definite. Then x^* is a strict local minimizer of θ.*

The optimality conditions suggest to find a local minimizer by finding a zero of equation (2.3).

2.2 Solving the problem

Numerical methods for solving nonlinear least-squares problems are iterative, i.e. they seek to find a sequence of iterates $\{x_k\}$ whose limit is a solution to (LS). By a *local convergence* method we mean one that requires that the initial iterate x_0 is close to a local minimizer x^*. On the contrary, by a *global convergence* method we mean a method with the property that for any initial iterate x_0, it converges to a solution or fails to do so in one of a small number of ways. In this section, first we will review the main features of locally convergent methods for (LS); second, we will concentrate on trust-region methods.

2.2.1 Local convergence methods

The Gauss-Newton method, the Full-Newton method and the Levenberg-Marquardt method are locally convergent procedures that form the basis of many important and successful methods for solving nonlinear least-squares problems (LS).

The basic idea of these methods is to use the Taylor's Theorem A.1 (see Appendix) to approximate the objective function θ around x_k by a quadratic model of the form

$$m_k(p) = \frac{1}{2}\|\Theta_k\|^2 + p^T J_k^T \Theta_k + \frac{1}{2} p^T B_k p, \tag{2.4}$$

where B_k is a symmetric approximation of the Hessian $\nabla^2\theta_k$ and use the minimizer p_k of m_k to modify x_k, i.e.

$$x_{k+1} = x_k + p_k, \qquad p_k = \operatorname*{argmin}_{p\in\mathbb{R}^n} m_k(p).$$

If B_k is the Hessian matrix $\nabla^2\theta_k$ in (2.2), the model (2.4) is called Full-Newton model and the resulting method is called *Full-Newton* method. In fact, it is the Newton's method applied to the nonlinear system (2.3) and the step p_k satisfies

$$(J_k^T J_k + S_k)\, p_k = -J_k^T \Theta_k.$$

Due to well-known local convergence properties of the Newton method [17, 59], this procedure is quadratically convergent to a solution x^* of (LS) as long as $\nabla^2\theta(x)$ is Lipschitz continuous around x^* and $\nabla^2\theta(x^*)$ is positive definite, see e.g. [17, §10.3]. On the other hand, a disadvantage of this method is that S_k is rarely available analytically at reasonable cost and it is very expensive to approximate by finite differences.

An alternative approach is the so-called *Gauss-Newton* method that consists in letting $B_k = J_k^T J_k$ in (2.4). This way the quadratic model (2.4) has the following special structure

$$m_k^{GN}(p) = \frac{1}{2}\|J_k\, p + \Theta_k\|^2, \tag{2.5}$$

and it is named Gauss-Newton model. In fact, the Gauss-Newton model is a linearization for the residual function. The step p_k satisfies

$$J_k^T J_k\, p_k = -J_k^T \Theta_k, \tag{2.6}$$

and the Gauss-Newton iteration

$$x_{k+1} = x_k - (J_k^T J_k)^{-1} J_k^T \Theta_k, \tag{2.7}$$

is well-defined if J_k has full column rank. This is the case when $m \geq n$ and J_k is full rank. Observe that if $m = n$, trivially the step p_k is the Newton step at x_k for the problem $\Theta(x) = 0$.

If $J_k^T J_k$ is singular, equation (2.6) has an infinite number of solutions and therefore an additional constraint must be imposed so that a unique step can be computed, [28]. One can choose the solution of (2.6) with minimum Euclidean norm, i.e. the step $(-{J_k}^+\,\Theta_k)$

where ${J_k}^+$ denotes the pseudoinverse of J_k, see Appendix A.4. Such a step will be called the *minimum norm* step and the iteration takes the form

$$x_{k+1} = x_k - J_k^+ \Theta_k. \tag{2.8}$$

If problem (LS) states an underdetermined system of equations, the resulting method is known as the *normal flow* method [70].

Regarding the local convergence properties of the Gauss-Newton approach, there are many situations in which the term $J(x^*)^T J(x^*)$ is much more significant than the term $S(x^*)$ in (2.2) and the Gauss-Newton method performs as well as the Full-Newton method. This happens, for instance, when x^* is a zero or small-residual solution. Let us consider assumptions on Θ which are standard in the analysis of the Newton method. Let assume that $J(x^*)$ is full column rank. Theorem A.3 states that the Gauss-Newton method is q-quadratically convergent if $S(x^*) = 0$, and that it is locally q-linearly convergent if $S(x^*)$ is small relative to $J(x^*)^T J(x^*)$. If $S(x^*)$ is too large, the Gauss-Newton method may not converge. On the other side, in underdetermined problems $J(x^*)^T J(x^*)$ is not full rank but Theorem A.4 establishes that the normal flow method locally converges q-quadratically to zero-residual solutions. To sum up, provided a good initial guess, the Gauss-Newton and the normal flow methods are especially suitable for zero and small-residual problems.

Finally, the *Levenberg-Marquardt* method is a further alternative approach for solving problem (LS). Given a positive parameter μ_k, the quadratic model considered is the sum of squares

$$m_k^{LM}(p) = \frac{1}{2}\|J_k\, p + \Theta_k\|^2 + \frac{1}{2}\mu_k \|p\|^2,$$

and it corresponds to (2.4) letting $B_k = J_k^T J_k + \mu_k I_n$. The minimizer p_k of m_k^{LM} solves the linear system

$$(J_k^T J_k + \mu_k I_n)\, p_k = -J_k^T \Theta_k. \tag{2.9}$$

Since μ_k is strictly positive, $J_k^T J_k + \mu_k I_n$ is a positive definite matrix and the system (2.9) has a unique solution. Therefore, the Levenberg-Marquardt method is well-defined even when J_k is not full column rank.

The local convergence properties of the Levenberg-Marquardt method are similar to those of the Gauss-Newton method if $J(x^*)$ is full column rank and μ_k is updated by an appropriate rule, see Theorem A.5. On the other hand, recently Levenberg-Marquardt methods that enjoy strong convergence properties to zero-residual solutions x^* with a weaker condition than the full column rank assumption of $J(x^*)$ have been proposed [14, 20, 48, 51, 52, 72, 73]. Such methods retain the fast local convergence properties in case $J(x^*)^T J(x^*)$ is singular, assuming that $\|\Theta(x)\|$ provides a local error bound for problem (LS) near x^* and the parameter μ_k is chosen as $\mu_k = O(\|\Theta_k\|^\delta)$, $\delta \in [1, 2]$. We will give details on such methods in Chapter 3 where we will develop a Levenberg-Marquardt method with strong local convergence properties for bound-constrained least-squares problems.

2.2.2 Trust-region methods

The locally convergent algorithms discussed in the previous section can and do fail when the initial iterate is not near to a solution. Globalization techniques improve the likelihood of convergence from initial approximations that may not be near to a solution. In this section we present the main features of a class of globalization algorithms known as *trust-region* methods. An exhaustive description and analysis of trust-region methods can be found in the book [12] by Conn, Gould and Toint.

The basic idea of a trust-region method is, at iteration k, defining around the current iterate x_k, a quadratic model for θ and a region within which the model is trusted to be an adequate representation of θ. The trial step is then computed, nearly exactly or approximately, minimizing this quadratic model inside the trust-region. Throughout the section, trust-region methods for problem (LS) will be discussed considering the Gauss-Newton model m_k^{GN} in (2.5).

At each iteration k, we let $\Delta_k > 0$ be the radius of the ball centered at x_k

$$\{x \in \mathbb{R}^n \, : \, x = x_k + p, \, \|p\| \leq \Delta_k\}, \tag{2.10}$$

in which the quadratic model m_k^{GN} is trusted to accurately represent the objective function θ in (LS). The scalar Δ_k is called the trust-region radius and the set (2.10) is called the trust-region. Then to obtain the trial step, we seek a solution p_{tr} of the trust-region problem

$$\min \{m_k^{GN}(p) = \frac{1}{2}\|J_k\, p + \Theta_k\|^2 \, : \, \|p\| \leq \Delta_k\}. \tag{2.11}$$

A solution of (2.11) is fully characterized by the following theorem.

Theorem 2.4 *([56]) The problem (2.11) is solved by $p_{tr} = p(\lambda)$ where*

$$p(\lambda) = -(J_k^T J_k + \lambda I_n)^{-1} \nabla \theta_k, \tag{2.12}$$

for some $\lambda \geq 0$, such that $\lambda \, (\|p(\lambda)\| - \Delta_k) = 0$. In case J_k is not full column rank, $p(0)$ is defined by the limiting process

$$p(0) = \lim_{\lambda \to 0^+} p(\lambda) = -J_k^+ \Theta_k.$$

Therefore, there are two possibilities: either $\lambda = 0$ and $\|p(0)\| \leq \Delta_k$, in which case $p_{tr} = -J_k^+ \Theta_k$ is the solution for which $\|p_{tr}\|$ is least, or $\lambda > 0$ and $\|p(\lambda)\| = \Delta_k$, and then $p(\lambda)$ is the unique solution p_{tr} to problem (2.11).

Having solved problem (2.11), one must decide whether to accept the trial step or to change the trust-region radius. Usually, the trust-region radius and the new point $x_k + p_{tr}$ are tested simultaneously and this test is centered on how well the quadratic model approximates the objective function inside the trust-region. We measure this using the ratio ρ_θ of the actual to the predicted reduction defined as

$$\rho_\theta(p_{tr}) = \frac{\theta(x_k) - \theta(x_k + p_{tr})}{m_k^{GN}(0) - m_k^{GN}(p_{tr})}. \tag{2.13}$$

Note that since the step p_{tr} is obtained by minimizing the model m_k^{GN} over a region that includes the step $p = 0$, the predicted reduction $(m_k^{GN}(0) - m_k^{GN}(p_{tr}))$ will always be nonnegative.

In practice, if $\rho_\theta(p_{tr})$ is close to 1, there is good agreement between the model m_k^{GN} and the function θ over this step, so $x_k + p_{tr}$ is accepted as the new iterate and it is safe to expand the trust region for the next iteration. If $\rho_\theta(p_{tr})$ is positive but not close to 1, $x_{k+1} = x_k + p_{tr}$ but the trust region is not altered. If $\rho_\theta(p_{tr})$ is close to zero or negative, the step p_{tr} is rejected and the trust region is shrunk. Algorithm 2.1 describes the process.

Algorithm 2.1 TRUST-REGION : k-TH ITERATION

```
Input: $0 < \Delta_k$, $0 < \eta_1 \leq \eta_2 < 1$ and $0 < \gamma_1 < 1 < \gamma_2$.
    1. Compute $p_{tr}$ by solving (2.11);
    2. Evaluate $\rho_\theta(p_{tr})$ from (2.13);
    3. If $\rho_\theta(p_{tr}) \geq \eta_1$, then set $x_{k+1} = x_k + p_{tr}$;
       Else set $\Delta_k = \gamma_1 \Delta_k$; go to Step 1.
    4. If $\rho_\theta(p_{tr}) \geq \eta_2$, then set $\Delta_{k+1} = \gamma_2 \Delta_k$;
       Else set $\Delta_{k+1} = \Delta_k$.
```

Now we focus on Step 1 of Algorithm 2.1, i.e. on solving the trust-region problem (2.11). Two different approaches may be considered. The former is based on the characterization in Theorem 2.4 and attempts to find a *nearly exact* solution of the trust-region problem; the latter seeks an *approximate* solution.

The method proposed by Moré and Sorensen in [57] follows the former approach. Suppose J_k is full column rank. If $\|(J_k^T J_k)^{-1} \nabla \theta_k\| \leq \Delta_k$, then the solution taken is $p_{tr} = (J_k^T J_k)^{-1} \nabla \theta_k$, otherwise $p_{tr} = p(\lambda)$ where $p(\lambda)$ is given in (2.12) and the value $\lambda > 0$ is sought so that

$$\|p(\lambda)\| = \Delta_k. \tag{2.14}$$

Problem (2.14) is a one-dimensional root-finding problem in the variable λ and can be solved by an efficient implementation of the Newton's method where the Cholesky factorization of $J_k^T J_k + \lambda I_n$ is performed at each iteration.

On the other side, if J_k is not full column rank the special structure of problem (LS) implies that the gradient $\nabla \theta_k$ is orthogonal to the eigenspace S_{min} associated to the smallest eigenvalue $\lambda_{min} = 0$ of $J_k^T J_k$. In fact, letting $S_{min} = \{z \in \mathbb{R}^n \, : \, J_k^T J_k z = 0, \; z \neq 0\}$, then for all $z \in S_{min}$ we have $\|J_k z\|^2 = z^T J_k^T J_k z = 0$ and then

$$\nabla \theta_k^T z = \Theta_k^T J_k z = 0.$$

This case is known as the *hard case*. In the hard case, the Moré and Sorensen procedure computes a step p_{tr} associated with $\lambda = 0$ and such that $\|p_{tr}\| = \Delta_k$. From a computational point of view, the algorithm finds an accurate approximation to the vector

$$p_{tr} = -J_k^+ \Theta_k + \tau z, \tag{2.15}$$

where $z \in S_{min}$ and τ is chosen so that $\| - J_k^+ \Theta_k + \tau z, \| = \Delta_k$, without the actual computation of the components indicated in (2.15), [57, §3]. It is important to point out that this approach steps to the boundary even if $\|J_k^+ \Theta_k\| < \Delta_k$, i.e. the minimum norm solution of problem (2.11) is in the interior of the trust-region. Global convergence properties for the resulting trust-region algorithm are proved in [57].

In this thesis we will use methods that solve the trust-region problem approximately. For global convergence purpose, it is enough to find an approximate solution p_{tr} that lies within the trust-region and gives a sufficient reduction in the model m_k^{GN}. The sufficient reduction can be quantified in term of the Cauchy point which is the minimizer of the model m_k^{GN} along the steepest descent direction $(-\nabla\theta_k)$ within the trust-region, i.e.

$$p_k^c = \text{argmin}\,\{m_k^{GN}(p)\ :\ p = -\tau\nabla\theta_k,\ \tau > 0,\ \|p\| \leq \Delta_k\}. \tag{2.16}$$

It is easy to see that the Cauchy point has a closed-form that is inexpensive to calculate [59, §4.1].

The Cauchy point is of crucial importance in deciding if an approximate solution of the trust-region subproblem is acceptable; specifically, a trust-region method is globally convergent if, at each iteration, the step p_{tr} taken attains a reduction in the model m_k^{GN} that is at least some fixed multiple of the decrease attained by the Cauchy step [59, Theorem 4.5-4.7]. The idea of the following methods is to generate approximate solutions to the trust-region problem (2.11) starting by computing the Cauchy point and then trying to improve on it. This way, global convergence is guaranteed.

The so-called *dogleg* method belongs to a special class of algorithms that approximate the solution of the trust region problem by minimizing m_k^{GN} along a piecewise linear path called dogleg because of its shape. Let us consider the case where $J_k^T J_k$ is positive definite. One may think of the dogleg path as a piecewise linear approximation to the path with parametric representation

$$\{\,-(J_k^T J_k + \lambda I_n)^{-1}\nabla\theta_k\,\}_{\lambda\geq 0}. \tag{2.17}$$

From Theorem 2.4, this is the path on which the exact solution of the trust-region problem lies. The classical dogleg path replaces the curved trajectory (2.17) with a path consisting of two line segments. The first line segment runs from the origin to the unconstrained minimizer along the steepest descent direction $(-\nabla\theta_k)$ given by

$$p_k^U = -\frac{\|\nabla\theta_k\|^2}{\|J_k\nabla\theta_k\|^2}\nabla\theta_k,$$

while the second line segment runs from p_k^U to the unique unconstrained minimizer p_k^N of m_k^{GN}. Formally, we denote this trajectory by $p(\tau)$, for $\tau \in [0, 2]$, where

$$p(\tau) = \begin{cases} \tau p_k^U, & \tau \in [0,1], \\ p_k^U + (\tau - 1)(p_k^N - p_k^U), & \tau \in [1,2]. \end{cases} \tag{2.18}$$

The dogleg method chooses p_{tr} to minimize the model m_k^{GN} along this path, subject to the trust-region bound. In fact, it is not even necessary to carry out a search, because it can be proved that $\|p(\tau)\|$ is an increasing function of τ and $m_k^{GN}(p(\tau))$ is a decreasing function of τ, see [59, Lemma 4.1]. This implies that as long as $\|p_k^N\| \geq \Delta_k$, the dogleg path (2.18) intersects the trust-region boundary at most once and this point is the approximate p_{tr} sought. In practice, we have

$$p_{tr} = \begin{cases} p_k^N & \text{if } \|p_k^N\| \leq \Delta_k, \\ \dfrac{\Delta_k}{\|p_k^U\|} p_k^U & \text{if } \|p_k^U\| > \Delta_k, \\ p_k^U + \tau(p_k^N - p_k^U), \text{ with } \tau \in [0,1) \text{ s.t.} & \\ \qquad \|p_k^U + \tau(p_k^N - p_k^U)\|^2 = \Delta_k^2 & \text{otherwise,} \end{cases}$$

and by the properties stated above, the value of the model at p_{tr} is guaranteed to be at least as good as the value at the Cauchy point p_k^c.

The dogleg strategy can be adapted to handle the case where $J_k^T J_k$ is positive semidefinite by choosing the minimum norm minimizer $p_k^N = -J_k^+ \Theta_k$ between the unconstrained minimizers of m_k^{GN}.

Clearly, the main cost of the dogleg strategy is the computation of p_k^N that involves the solution of the system (2.6), i.e. the factorization of J_k. When problem (LS) is large and J_k is dense, this operation may be quite costly. In order to avoid the high overhead of computing series of factorizations while approximating a solution of (2.11) along the iterations, there have been proposed techniques based on iterative methods for solving linear systems.

The Conjugate Gradient (CG) method is an iterative algorithm for solving linear systems with symmetric positive definite coefficient matrices [7, 39] and it is the core of most strategies for finding an approximate solution of (2.11) via iterative methods [10, 32, 66, 68].

For the moment we assume that $J_k^T J_k$ is positive definite. The idea is applying the CG method to the linear system (2.6), generating a sequence of approximations $\{p_k^{(j)}\}_{j\geq 0}$ of the unconstrained minimizer p_k^N of m_k^{GN} and then exploiting the special properties of the CG iterations to handle the trust-region constraint.

Let $p_k^{(0)} = 0$. Each $p_k^{(j)}$ with $j \geq 1$ is generated by the CG method minimizing m_k^{GN} over the j-th Krylov subspace

$$\mathcal{K}_j = span\left(\{(J_k^T J_k)^i J_k^T \Theta_k\}_{i=0}^{j-1}\right), \tag{2.19}$$

i.e. each $p_k^{(j)}$ solves the following subspace minimization problem

$$\min\{m_k^{GN}(p) \,:\, p \in \mathcal{K}_j\}.$$

The subspaces $\{\mathcal{K}_j\}$ satisfy the expansion property $\mathcal{K}_j \subset \mathcal{K}_{j+1}$ for $j \geq 1$ and this implies that CG computes the exact solution of the system (2.6) in at most l iterations, where $l \leq n$ is the number of distinct eigenvalues of $J_k^T J_k$ [38]. The CG method remains valid when $J_k^T J_k$ is positive semidefinite and in this event the algorithm terminates computing the minimum norm step $(-J_k^+\Theta_k)$ in at most l iterations, where l is the number of distinct nonzero singular values of J_k [7, 38].

A crucial property of the sequence $\{p_k^{(j)}\}$ is that the norm of the iterates is monotonically increasing, i.e. $\|p_k^{(j)}\| < \|p_k^{(j+1)}\|$ for all $j \geq 0$. This allows to extend CG to cope with the trust-region constraint. In fact, it is acceptable to stop iterating as soon as the trust-region boundary is encountered because no further iterates giving a lower value of m_k^{GN} will be inside the trust-region. Summarizing, either CG finds the unconstrained minimizer of m_k^{GN} in the interior of the trust-region or exits the trust-region, i.e. finds

$$\|p_k^{(j-1)}\| < \Delta_k \leq \|p_k^{(j)}\|, \tag{2.20}$$

for some $j \geq 1$. In the last case, a solution of the problem (2.11) must lie on the boundary and different techniques have been proposed to approximate it. The *Truncated Conjugate Gradient* method proposed independently by Steihaug [66] and Toint [68], computes the so-called Steihaug-Toint point p_k^{ST} that is

$$p_k^{ST} = p_k^{(j-1)} + \tau(p_k^{(j)} - p_k^{(j-1)}),\ \tau \in (0,1] \ \text{ such that } \ \|p_k^{ST}\| = \Delta_k. \tag{2.21}$$

The Steihaug-Toint point has the favorable property that the optimal decrease of m_k^{GN} at the exact solution of the trust-region problem (2.11), i.e. $(m_k^{GN}(0) - m_k^{GN}(p_{tr}))$, is no more than twice that achieved at p_k^{ST} [12, Theorem 7.5.9]. On the other hand, the computation of the Steihaug-Toint point does not allow the accuracy of the constrained solution to be specified.

Regarding the global convergence of the trust-region method based on CG, it is important to remark that since $p_k^{(0)} = 0$, the step $p_k^{(1)}$ coincides with the Cauchy point p_k^c in (2.16). This implies that the value of the model at each subsequent iterate will be lower than the value attained at the Cauchy point which ensures global convergence.

The algorithm described can be applied substituting CG method with CGLS method [39] or LSQR method [60, 61]. In fact, the methods CGLS and LSQR are mathematically equivalent to CG applied to the system (2.6) and generate exactly the same sequence of iterates. As a consequence, all the properties of CG iterates are retained. Nevertheless their implementations improve the numerical performance of CG if the system (2.6) is ill-conditioned. In particular, CGLS method is a slight modification of CG method derived by algebraic rearrangements that prevent the computation of the matrix $J_k^T J_k$ and allow the only use of matrix-vector products of the form $J_k v$ and $J_k^T v$. The LSQR procedure is based on the iterative bi-diagonalization algorithm due to Golub and Kahan [30] applied to J_k for generating an orthonormal basis for the Krylov subspace (2.19); in [60] it is shown that the numerical properties of LSQR are better than those of CGLS and more accurately reflect the conditioning of the problem.

Methods that allow a solution on the trust-region boundary to be calculated to any prescribed accuracy have been proposed in [10, 32]. The approach presented in these papers consists in finding an iterate for which (2.20) occurs and then solving a sequence of equality constrained subspace problems of the form

$$\min\{m_k^{GN}(p)\ :\ p \in \mathcal{K}_j,\ \|p\| = \Delta_k\},$$

until a prescribe accuracy is reached. The iterative methods used are the CG method in [32] and the LSQR method in [10].

2.3 The bound-constrained problem

Consider the bound-constrained least-squares problem

$$\min_{x\in\Omega} \theta(x) = \frac{1}{2}\|\Theta(x)\|^2, \tag{BCLS}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ and Ω is the n-dimensional box $\Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}$, $l \in (\mathbb{R} \cup -\infty)^n$, $u \in (\mathbb{R} \cup \infty)^n$, $l < u$.

The necessary conditions for optimality admit the possibility that the minimizer lies on the boundary of Ω; the first-order and second-order necessary conditions are expressed by the following theorems.

Theorem 2.5 (First-Order Necessary Conditions) *([49]) Let θ be continuously differentiable on Ω. If x^* is a solution to problem (BCLS), then x^* is such that*

$$\nabla\theta(x^*)^T(x - x^*) \geq 0, \qquad \textit{for all } x \in \Omega. \tag{2.22}$$

A point x^* satisfying (2.22) is called a *stationary point* for problem (BCLS).

Given $x^* \in \Omega$, let $\mathcal{F}^*$ denote the set of indices corresponding to free components of x^*, $\mathcal{F}^* = \{i : l_i < x_i^* < u_i\}$ and $(\nabla^2\theta(x^*))_{\mathcal{F}^*}$ denote the submatrix of $\nabla^2\theta(x^*)$ with indices in $\mathcal{F}^*$.

Theorem 2.6 (Second-Order Necessary Conditions) *([49]) If x^* is a solution to problem (BCLS) and θ is twice continuously differentiable in an open neighbourhood of x^*, then $(\nabla^2\theta(x^*))_{\mathcal{F}^*}$ is positive semidefinite.*

The concept of nondegenerate stationary point is used in the formulation of the sufficient conditions for (BCLS).

Definition 2.3.1 *([49]) A point $x^* \in \Omega$ is a nondegenerate stationary point for problem (BCLS) if x^* is stationary and*

$$(\nabla\theta(x^*))_i \neq 0, \qquad \textit{if } x_i^* = l_i \textit{ or } x_i^* = u_i,$$

for $i = 1, \ldots, n$.

For a nondegenerate stationary point the sufficient conditions are very similar to the unconstrained case.

Theorem 2.7 (Second-Order Sufficient Conditions) *([49]) Let $x^* \in \Omega$ be a nondegenerate stationary point for problem (BCLS). Let θ be twice continuously differentiable in an open neighbourhood of x^* and assume that the matrix $(\nabla\theta(x^*))_{\mathcal{F}^*}$ is positive definite. Then x^* is a solution to problem (BCLS).*

Let Θ be a continuously differentiable mapping. The first-order necessary conditions (2.22) for x^* to be a local minimizer can be formulated componentwise as

$$(\nabla\theta(x^*))_i \begin{cases} = 0 & \text{if} \quad l_i < x_i^* < u_i, \\ \leq 0 & \text{if} \quad x_i^* = u_i, \\ \geq 0 & \text{if} \quad x_i^* = l_i, \end{cases} \tag{2.23}$$

for $i = 1, \ldots, n$. Coleman and Li [11] noted that introducing a proper scaling matrix, (2.23) can be stated as a system of nonlinear equations which parallels the system (2.3) of the unconstrained case. Let D be the diagonal scaling matrix

$$D(x) = diag(|v_1(x)|, \ldots, |v_n(x)|), \tag{2.24}$$

and

$$v_i(x) = \begin{cases} x_i - u_i & \text{if } (\nabla\theta(x))_i < 0,\ u_i < \infty, \\ x_i - l_i & \text{if } (\nabla\theta(x))_i \geq 0,\ l_i > -\infty, \\ 1 & \text{if } (\nabla\theta(x))_i \geq 0,\ l_i = -\infty \ \text{ or } \ (\nabla\theta(x))_i < 0,\ u_i = \infty, \end{cases} \tag{2.25}$$

for $i = 1, \ldots, n$. Then conditions (2.23) are equivalent to the system of nonlinear equations

$$D(x)\nabla\theta(x) = 0, \tag{2.26}$$

for $x \in \Omega$. By (2.1), equation (2.26) can be written as $D(x)J(x)^T\Theta(x) = 0$.

The scaling matrix D has further nice properties. In particular, the scaled steepest descent direction of θ at x_k defined as

$$d_k \ = \ -D_k\nabla\theta_k, \tag{2.27}$$

is well-angled with respect to the bounds. In fact, for the components $(x_k)_i$ which are approaching the correct bounds, d_k becomes increasingly tangential to the bounds; hence, the bounds will not prevent a large stepsize along d_k. On the other hand, for the components $(x_k)_i$ which are approaching the incorrect bounds, d_k points away from these bounds in relatively large angles [8].

In recent years, a number of methods have exploited the fact the first-order necessary conditions of (BCLS) can be written as (2.26). Such methods generate feasible iterates and are named *affine scaling* methods. The methods proposed in [25, 47, 73] and introduced in Section 1.2.1 belong to this class along with the methods in [1, 2, 3, 4, 43, 46] for square problems.

Chapter 3

The affine scaling trust-region methods

Bound-constrained least-squares problems can be a unifying formulation for the problems addressed in this thesis. In this chapter we discuss two ways to transform the feasibility problem (FP) into the problem

$$\min_{x\in\Omega} \theta(x) = \frac{1}{2}\|\Theta(x)\|^2, \qquad \text{(BCLS)}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ is a given continuously differentiable mapping and Ω is the n-dimensional box $\Omega = \{x \in \mathbb{R}^n \mid l \le x \le u\}$, $l \in (\mathbb{R} \cup -\infty)^n$, $u \in (\mathbb{R} \cup \infty)^n$, $l < u$. Then, we present two new trust-region methods for solving problem (BCLS) which belong to the class of affine scaling methods. The quadratic models used in the trust-region strategy are a Gauss-Newton model and a regularized Gauss-Newton model. A linear algebra phase arises at each iteration of the methods; in this chapter we restrict to the case where matrix factorizations are used.

In Section 3.1 we discuss formulations of nonlinear feasibility problems as problem (BCLS) where the function Θ is continuously differentiable. In Section 3.2 we explore two approaches for bound-constrained least-squares problems which require differentiability of the residual function Θ. In Section 3.3 we show that the methods are globally and potentially q-quadratically convergent. A comparison between the numerical performance of the methods is presented in Section 3.4.

3.1 Nonlinear feasibility problems

Consider the problem

$$\begin{array}{c} C_E(v) = 0, \\ C_I(v) \le 0, \\ v_l \le v \le v_u, \end{array} \qquad \text{(FP)}$$

where the vector functions $C_E : \mathbb{R}^p \to \mathbb{R}^{m_E}$ and $C_I : \mathbb{R}^p \to \mathbb{R}^{m_I}$ are continuously differentiable, $v_l \in (\mathbb{R} \cup -\infty)^p$, $v_u \in (\mathbb{R} \cup \infty)^p$ and $v_l < v_u$. Now we discuss formulations of this problem which are alternative to those described in Section 1.2.2 and let $m = m_E + m_I$.

We convert the general inequalities, $C_I(v) \leq 0$, into equalities and transform the problem (FP) into problem (BCLS) where Θ is continuously differentiable. A possible transformation is to add a slack variable $s \in \mathbb{R}^{m_I}$ whose non-positivity is imposed by means of simple bounds. The problem then becomes the least-squares problem (BCLS) where

$$x = \begin{pmatrix} v \\ s \end{pmatrix}, \; \Theta(x) = \begin{pmatrix} C_E(v) \\ s - C_I(v) \end{pmatrix}, \; l = \begin{pmatrix} v_l \\ -\mathbf{Inf} \end{pmatrix}, \; u = \begin{pmatrix} v_u \\ \mathbf{0} \end{pmatrix}. \tag{3.1}$$

Here, $\Theta : \mathbb{R}^{p+m_I} \to \mathbb{R}^m$, the column vectors $-\mathbf{Inf}$ and $\mathbf{0}$ are in $\mathbb{R}^{m_I}$ and such that $(-\mathbf{Inf})_i = -\infty$, $(\mathbf{0})_i = 0$ for $i = 1, \ldots, m_I$.

Another transformation is to use the function

$$[t]_+ = \frac{1}{2} \max\{t, 0\}^2,$$

which is continuously differentiable. Then, the problem takes the form (BCLS) where

$$x = v, \; \Theta(x) = \begin{pmatrix} C_E(x) \\ [C_I(x)]_+ \end{pmatrix}, \; l = v_l, \; u = v_u. \tag{3.2}$$

Note that $\Theta : \mathbb{R}^p \to \mathbb{R}^m$ and $[C_I(x)]_+$ denotes the vector of infeasibilities at x.

The above formulations are such that any zero-residual solution to problem (BCLS) gives a solution to problem (FP). It is also important to note that by (2.1) any zero-residual solution to (BCLS) is a degenerate stationary point.

In both transformations, the number of components of the vector function Θ is m while the number of variables differs. The transformation (3.1) adds m_I extra variables and the problem (BCLS) is underdetermined if the system of equalities is underdetermined. On the contrary, the transformation (3.2) leaves the number of unknowns unchanged and Θ inherits the dimensions of the system of equalities and general inequalities.

For the functions Θ in (3.1) and (3.2), the Jacobian matrix J of Θ is given by

$$J(x) = \begin{pmatrix} C'_E(v) & \mathbf{0} \\ -C'_I(v) & I_{m_I} \end{pmatrix}, \tag{3.3}$$

and

$$J(x) = \begin{pmatrix} C'_E(x) \\ \max\{C_I(x), \mathbf{0}\}\, C'_I(x) \end{pmatrix}, \tag{3.4}$$

respectively, where C'_E and C'_I denote the Jacobian matrices of C_E and C_I, $\mathbf{0}$ denotes the null matrix in $\mathbb{R}^{m_E \times m_I}$ in (3.3) and the null vector in $\mathbb{R}^{m_I}$ in (3.4). Letting x^* be a solution to (FP), the matrix $J(x^*)$ given in (3.4) has full rank if the system of equalities $C_E(x) = 0$ is square or overdetermined and $C'_E(x^*)$ has full rank. Otherwise, $J(x^*)$ does not have full rank and the rate of convergence of a minimization method for smooth problems may be inhibited.

3.2 The methods and the algorithmic options

We present trust-region methods for solving the bound-constrained least-squares problem (BCLS). The sequence $\{x_k\}$ generated by the methods consists of feasible points, i.e. $x_k \in \Omega$, $k \geq 0$. Without loss of generality we assume that for all k, x_k is not a stationary point for the least-squares problem (BCLS).

At the k-th iteration, we define a quadratic model m_k for θ as

$$m_k(p) = \frac{1}{2}\|J_k\, p + \Theta_k\|^2 + \frac{1}{2}\mu_k\|p\|^2, \tag{3.5}$$

where μ_k is a nonnegative scalar. Then, given the trust-region radius $\Delta_k \geq 0$, we consider the trust-region problem

$$\min\{m_k(p)\ :\ \|p\| \leq \Delta_k\}. \tag{3.6}$$

We allow for two options in the choice of the parameter μ_k in (3.5). The first option is to let $\mu_k = 0$ so that the Gauss-Newton model is used. The second option is to choose a strictly positive scalar μ_k which depends on the value $\|\Theta_k\|$; this way, m_k can be interpreted as a regularized Gauss-Newton model.

In order to give the precise characterization of a solution to (3.6), we first note that the Gauss-Newton model is convex and that the global unconstrained minimizer is unique if J_k is full column rank. On the other hand, the regularized Gauss-Newton model is strictly convex and admits a unique unconstrained minimizer. Let p_k^N denote the minimum norm minimizer of m_k. If $\mu_k = 0$, the step p_k^N has the form

$$p_k^N = -{J_k}^+\,\Theta_k. \tag{3.7}$$

Otherwise p_k^N solves the linear system

$$({J_k}^T J_k + \mu_k I_n)\, p_k^N = -{J_k}^T\Theta_k. \tag{3.8}$$

Trivially, the model m_k in (3.5) can be written as

$$m_k(p) = \frac{1}{2}\left\|\begin{pmatrix} J_k \\ \sqrt{\mu_k}\, I_n \end{pmatrix} p + \begin{pmatrix} \Theta_k \\ 0 \end{pmatrix}\right\|^2. \tag{3.9}$$

Then, taking into account the trust-region constraint, by Theorem 2.4 we know that any solution p_{tr} to the trust-region problem (3.6) satisfies the equation

$$({J_k}^T J_k + (\mu_k + \lambda) I_n)\, p_{tr} = -{J_k}^T\Theta_k, \tag{3.10}$$

where $\lambda \geq 0$ and $\lambda(\|p_{tr}\| - \Delta_k) = 0$. Then we have the following cases.

(i) If ${J_k}^T J_k + (\mu_k + \lambda) I_n$ is positive definite, then p_{tr} is unique; clearly this is the case whenever $\mu_k > 0$ or $J_k^T J_k$ is nonsingular.

(ii) The vector p_k^N solves (3.10) with $\lambda = 0$. Hence there are two possibilities: either p_k^N is inside the trust-region, i.e. $p_{tr} = p_k^N$, or there are no unconstrained minimizers of m_k inside the trust-region.

(iii) If $\mu_k = 0$, ${J_k}^T J_k$ is rank deficient and p_k^N given in (3.7) is inside the trust-region, then $p_{tr} = p_k^N$ is the solution to (3.6) for which $\|p_{tr}\|$ is least.

Correspondingly to problem (3.6), we consider a generalized Cauchy step p_k^C which is typical of affine scaling trust-region methods. Such step is defined along the scaled steepest descent direction d_k given in (2.27) and minimizes m_k within the feasible trust-region, i.e.

$$p_k^C \quad = \quad \operatorname{argmin}\{m_k(p) \ : \ p \in span(d_k), \ \|p\| \le \Delta_k, \ x_k + p \in \Omega\}. \tag{3.11}$$

Now we outline the general form of the method which will be denoted as `TREBO` method (Trust-REgion method for BOund-constrained least-squares problems). It is the paradigm for our procedure and two different implementations are allowed. They correspond to the following possible choices of the sequence $\{\mu_k\}$. Fixed $\mu \ge 0$ and a small positive scalar $\hat{\mu}$, we let

$$\mu_k = \begin{cases} \min\{\hat{\mu}, \mu\} \, \|\Theta_0\|^2 & \text{if } k = 0, \\ \min\{\mu_{k-1}, \, \mu\|\Theta_k\|^2\} & \text{if } k > 0. \end{cases} \tag{3.12}$$

Choosing $\mu = 0$ we get $\mu_k = 0$ for all k; on the other hand if $\mu > 0$ then $\mu_k > 0$ for all k such that $\|\Theta_k\| \neq 0$.

The assignment (3.12) is in accordance to the proposal in [48]. The implementation with $\mu = 0$ is based on the Gauss-Newton model and will be denoted as `TREBO-GN`. The other implementation is obtained setting $\mu > 0$ and gives rise to a Levenberg-Marquardt method; therefore it will be denoted `TREBO-LM` method. We remark that the parameter $\hat{\mu}$ in (3.12) has effect only on the Levenberg-Marquardt procedure; it is assumed small enough so that the model m_k does not drift too far from the Gauss-Newton model when $\|\Theta_k\|$ is large.

Algorithm 3.1 describes the k-th iteration of the `TREBO` framework. We now analyze it in detail specifying the tasks that are not shared by the two implementations and discussing the most relevant algorithmic options.

The input parameters $\Delta_{min}, \beta_1, \beta_2, \hat{\mu}, \mu, \delta$ are independent of k. The initial trust-region radius Δ_k satisfies $\Delta_k \ge \Delta_{min}$. The parameter μ_k which characterizes the model m_k used is evaluated in Step 1. Then, in Step 2 we find the minimum norm minimizer p_k^N of m_k. For the `TREBO-GN` method the step p_k^N in (3.7) can be computed in a numerically reliable way from either the complete orthogonal factorization of J_k or the singular value decomposition of J_k, [28, §6.6.4, §6.6.5]. In the `TREBO-LM` method the step p_k^N solves the linear system (3.8) with positive definite matrix and there are two ways to compute it. The simplest way to obtain p_k^N is to use the Cholesky decomposition on the system (3.8). Alternatively, p_k^N can be computed applying the QR decomposition to the least-squares problem $\min_p m_k(p)$ with $m_k(p)$ stated as (3.9). A possible disadvantage of the first approach is that the condition number of $J_k^T J_k + \lambda I_n$ is the square of that of the matrix in (3.9).

Taking into account (2.27) and (3.11), Step 3 is performed letting

$$p_k^C = c_k \, d_k, \qquad c_k = \begin{cases} \hat{c}_k & \text{if} \quad x_k + \hat{c}_k d_k \in \Omega, \\ \lambda_k & \text{otherwise}, \end{cases} \tag{3.13}$$

where

$$\hat{c}_k = \min \left\{ \frac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|J_k d_k\|^2 + \mu_k\|d_k\|^2}, \frac{\Delta_k}{\|d_k\|} \right\}, \ \lambda_k = \operatorname{argmax}\{\lambda > 0, x_k + \lambda d_k \in \Omega\}. \quad (3.14)$$

It is easy to verify that λ_k has the form

$$\lambda_k = \min_{1\le i\le n} \Lambda_i \quad \text{where} \quad \Lambda_i = \begin{cases} \max\left\{\frac{l_i-(x_k)_i}{(d_k)_i}, \frac{u_i-(x_k)_i}{(d_k)_i}\right\} & \text{if } (d_k)_i \neq 0, \\ \infty & \text{if } (d_k)_i = 0. \end{cases} \quad (3.15)$$

Algorithm 3.1 TREBO: k-TH ITERATION

```
Input: $x_k \in \Omega$, $\Delta_k \ge \Delta_{min} > 0$, $\mu \ge 0$, $\hat{\mu}$, $\beta_1$, $\beta_2$, $\delta \in (0,1)$.
  1. Define $\mu_k$ according to (3.12).
  2. Compute the minimum norm solution $p_k^N$ to the problem $\min_{p\in\mathbb{R}^n} m_k(p)$.
  3. Compute the generalized Cauchy step $p_k^C$ based on (3.11).
  4. If $\|p_k^N\| \le \Delta_k$ set $p_{tr} = p_k^N$;
     Else find the dogleg step $p_{tr}$ for (3.6).
  5. Let $\bar{p}_{tr} = P_\Omega(x_k + p_{tr}) - x_k$.
  6. If
         $\rho_c(\bar{p}_{tr}) = \frac{m_k(0) - m_k(\bar{p}_{tr})}{m_k(0) - m_k(p_k^C)} \ge \beta_1,$     (3.16)
        Set $p_k = \bar{p}_{tr}$;
     Else find $p_k = t\,p_k^C + (1-t)\bar{p}_{tr}$, $t \in (0,1]$, such that (3.16) holds.
  7. If
         $\rho_\theta(p_k) = \frac{\theta(x_k) - \theta(x_k + p_k)}{m_k(0) - m_k(p_k)} \ge \beta_2,$     (3.17)
        Set $x_{k+1} = x_k + p_k$, choose $\Delta_{k+1} \ge \Delta_{min}$;
     Else reduce $\Delta_k$, $\Delta_k = \delta\Delta_k$, and go to Step 3.
```

In Step 4, the trust-region problem (3.6) is solved. We set $p_{tr} = p_k^N$ if p_k^N is inside the trust-region; otherwise, we find an approximate solution p_{tr} to (3.6) on the boundary of the trust-region by the dogleg strategy, see Section 2.2.2. To generate a new feasible iterate, in Step 5 we use the projection map $P_\Omega(x) = \max\{l, \min\{x, u\}\}$. Specifically, we project $x_k + p_{tr}$ onto the box Ω and let $\bar{p}_{tr}$ be a possibly modified step such that $x_k + \bar{p}_{tr}$ is feasible.

Steps 6–7 attempt to find a feasible iterate $x_{k+1} = x_k + p_k$ which provides a sufficient decrease in the value of θ with respect to x_k. In Step 6 we impose a sufficient decrease of m_k in comparison to the generalized Cauchy step p_k^C; this is crucial to make the method globally convergent [1, 11]. We let $p_k = \bar{p}_{tr}$ if $\bar{p}_{tr}$ satisfies (3.16), otherwise, we look for a step of the form $p_k = tp_k^C + (1-t)\bar{p}_{tr}$, $t \in (0,1]$, satisfying the required condition. This task can be accomplished easily in two different ways. The first possibility is to simply set $t = 1$, i.e. $p_k = p_k^C$ [11]. The second one is to find the scalar $t \in (0, 1)$ such that $\rho_c(p_k) = \beta_1$; this is equivalent to find the smallest positive root of the scalar quadratic equation in t of the form $\rho_c(tp_k^C + (1-t)\bar{p}_{tr}) - \beta_1 = 0$, [2].

In Step 7, we measure the quality of the quadratic model m_k as an approximation to θ around x_k. If the sufficient improvement condition (3.17) is satisfied, the new iterate is $x_{k+1} = x_k + p_k$ and Δ_{k+1} is fixed so that $\Delta_{k+1} \geq \Delta_{min}$; otherwise, p_k is rejected and the trust-region size Δ_k is reduced. We point out that in each iteration of the method the initial radius is greater than or equal to Δ_{min} while on termination of the iteration the trust-region radius may be smaller than Δ_{min}.

We conclude this section making some further comments. In both the `TREBO-GN` and `TREBO-LM` methods the use of the step p_k^N is motivated by the fast convergence attainable ultimately. This property will be shown in the next Section 3.3.

The use of the minimum norm step (3.7) in the `TREBO-GN` method is common to the methods given in [16, 25, 70]. The reason for using the dogleg strategy in the solution of (3.6) instead of the Moré and Sorensen algorithm [57], depends on the occurrence of the so-called "hard case". If $J_k^T J_k$ is positive semidefinite, the Moré and Sorensen strategy steps to the boundary of the trust-region even when the unconstrained minimizer p_k^N of m_k is safely inside and loses the opportunity of taking such step, see Section 2.2.2.

In the unconstrained setting, many versions of the Levenberg-Marquardt method have been proposed using various strategies for choosing the regularization parameter μ_k. In particular, the implementation of the Levenberg-Marquardt method as a trust-region algorithm is due to Moré [56]. Therefore, the combination of the Levenberg-Marquardt model and the trust-region strategy proposed here may be viewed as a double regularization. We point out that our choice (3.12) of the Levenberg-Marquardt parameter is not implemented as in a trust-region strategy. Global convergence of the `TREBO-LM` algorithm depends on the trust-region strategy while the aim of the parameter μ_k is to replace the Gauss-Newton model by a nearby strictly convex model and to achieve strong local convergence properties. In particular, quadratic convergence to a zero-residual solution can be achieved also in the case where $J^T J$ is singular at such point.

To complete the discussion on the Levenberg-Marquardt algorithm, we note from (3.12) that μ_k may become very small near to a zero-residual solution to (BCLS). The danger of this occurrence is that the matrix in (3.8) may become numerically singular. To cope with this situation, safeguards are needed in practical implementations to prevent μ_k from becoming too small. Finally, Fan and Yuan [20] noted some possible defects of the choice (3.12) and suggested the use of $\mu_k = O(\|\Theta_k\|)$ while still preserving local quadratic convergence properties; however, the numerical behaviour of Levenberg-Marquardt methods for the different choices of $\{\mu_k\}$ has not been investigated thoroughly and may deserve further analysis.

3.3 Convergence analysis

In this section we establish global and local convergence properties of the methods presented. The analysis in Section 3.3.1 and Section 3.3.2 is carried out considering the general form of the `TREBO` method, whereas for the local convergence analysis in Sections 3.3.3, 3.3.4 and 3.3.5 it will be necessary to distinguish between the `TREBO-LM` method and the `TREBO-GN` method.

Throughout the section we let $\{x_k\}$ be the sequence generated by any implementation of the `TREBO` method and, without loss of generality, we assume that for all k, x_k is not a stationary point for the least-squares problem (BCLS), i.e.

$$\|D_k \nabla \theta_k\| \neq 0. \tag{3.18}$$

Moreover we make the following basic assumptions on the function Θ in (BCLS).

Assumption 1 *There exists an open, bounded and convex set L containing the whole sequence $\{x_k\}$ such that $L \supset \{x \in \mathbb{R}^n : \exists\ x_k \quad s.t. \ \|x - x_k\| \leq r\}$, for some $r > 0$, and the Jacobian matrix J is Lipschitz continuous in L with Lipschitz constant $2\gamma_D$, i.e. for all $x, z \in L$*

$$\|J(x) - J(z)\| \leq 2\gamma_D \|x - z\|. \tag{3.19}$$

Assumption 2 *$\|\Theta'\|$ is bounded above on L and $\chi_L = \sup_{x \in L} \|J(x)\|$.*

It is easy to see that if Θ has the form (3.1) then these assumptions are equivalent to suppose that $C'_E(x)$, $C'_I(x)$ are bounded in norm for $x \in L$ and that $C'_E(x)$ and $C'_I(x)$ are Lipschitz continuous at every point of the set L. If Θ has the form (3.2) it can be shown following the lines of [16, Lemma 4.1] that the Assumptions 1 and 2 are equivalent to suppose that $C_I(x)$, $C'_E(x)$, $C'_I(x)$ are bounded in norm for $x \in L$ and that $C'_E(x)$ and $C'_I(x)$ are Lipschitz continuous at every point of the set L.

Trivially, Assumption 1 implies that the sequence $\{x_k\}$ is bounded. Moreover the following lemma can be easily proved.

Lemma 3.1 *Let Assumptions 1 and 2 hold. Then for all $x, z \in L$*

$$\|\Theta(x) - \Theta(z)\| \leq \chi_L \|x - z\|, \tag{3.20}$$
$$\|\Theta(x) - \Theta(z) - J(z)(x - z)\| \leq \gamma_D \|x - z\|^2. \tag{3.21}$$

Proof. Since Θ is continuously differentiable, by the Mean Value Theorem A.2 and Assumption 2 we have

$$\|\Theta(x) - \Theta(z)\| \leq \int_0^1 \|J(x + t(x - z))\| \, \|x - z\| dt \ \leq \chi_L \, \|x - z\|.$$

To prove (3.21), we use the Mean Value Theorem A.2 and (3.19) as follows

$$\begin{aligned}
\|\Theta(x) - \Theta(z) - J(z)(x - z)\| &= \left\| \int_0^1 (J(z + t(x - z)) - J(z))\,(x - z)\, dt \right\| \\
&\leq \int_0^1 \|J(z + t(x - z)) - J(z)\| \, \|x - z\| \, dt \\
&\leq \int_0^1 2\gamma_D \|x - z\|^2 \, t \, dt = \gamma_D \|x - z\|^2.
\end{aligned}$$

□

The properties below follow from the contractivity of the projection map P_Ω.

Lemma 3.2 *Let $x \in \Omega$, let p be a vector of $\mathbb{R}^n$ and let $\bar{p} = P_\Omega(x+p) - x$. Then we have*

$$\begin{aligned} \|\bar{p}\| &\leq \|p\|, & (3.22)\\ \|x + \bar{p} - z\| &\leq \|x + p - z\|, \ z \in \Omega. & (3.23) \end{aligned}$$

Proof. The projection map $P_\Omega(x)$ satisfies the contractivity property [6]:

$$\|P_\Omega(x) - P_\Omega(z)\| \leq \|x - z\|, \quad \text{for all } x \in \mathbb{R}^n, \ z \in \Omega.$$

This implies (3.22) since $\|\bar{p}\| = \|P_\Omega(x+p) - x\|$. Moreover, for all $z \in \Omega$

$$\|x + \bar{p} - z\| = \|x + P_\Omega(x+p) - x - z\| \leq \|x + p - z\|.$$

□

3.3.1 Termination of the iteration

An important result in any trust-region analysis is that condition (3.17) holds if the trust-region radius is small enough. This implies that the k-th iteration of the `TREBO` method terminates and the method is well-defined.

In order to prove this result we turn our attention to the actual reduction $ared(p_k)$ in θ at $x_k + p_k$

$$ared(p_k) = \theta(x_k) - \theta(x_k + p_k),$$

and to the predicted reduction $pred(p_k)$, i.e. the decrease in the quadratic model m_k at p_k

$$pred(p_k) = m_k(0) - m_k(p_k).$$

Lemma 3.3 *Let Assumption 1 hold. If p_k satisfies $m_k(p_k) \leq m_k(0)$ then*

$$ared(p_k) \geq pred(p_k) - \left(\frac{1}{2}\mu_k + \gamma_D\|\Theta_k\| + \frac{1}{2}\gamma_D^2\|p_k\|^2\right)\|p_k\|^2. \tag{3.24}$$

Proof. From the Mean Value Theorem A.2 it follows that

$$\|\Theta(x_k + p_k)\|^2 = \|\Theta_k + J_k p_k + w(x_k, p_k)\|^2,$$

where

$$w(x_k, p_k) = \int_0^1 (J(x_k + tp_k) - J(x_k))\, p_k\, dt.$$

Thus, we have

$$\begin{aligned} |m_k(p_k) - \theta(x_k + p_k)| &= \frac{1}{2}\left|\, \|\Theta_k + J_k p_k\|^2 + \mu_k\|p_k\|^2 - \|\Theta(x_k + p_k)\|^2 \,\right| \\ &\leq \frac{1}{2}\mu_k\|p_k\|^2 + \|\Theta_k + J_k p_k\|\|w(x_k, p_k)\| \\ &\quad + \frac{1}{2}\|w(x_k, p_k)\|^2. \end{aligned} \tag{3.25}$$

From the Lipschitz continuity of J_k, we obtain

$$\|w(x_k, p_k)\| \leq \gamma_D \|p_k\|^2$$

and by using $m_k(p_k) \leq m_k(0)$ we get $\|\Theta_k + J_k p_k\| \leq \|\Theta_k\|$. Then the inequality (3.25) becomes

$$|m_k(p_k) - \theta(x_k + p_k)| \leq \left(\frac{1}{2}\mu_k + \gamma_D\|\Theta_k\| + \frac{1}{2}\gamma_D^2\|p_k\|^2\right)\|p_k\|^2. \tag{3.26}$$

Since $m_k(0) = \theta(x_k)$, we have

$$\begin{array}{rcl} ared(p_k) & = & \theta(x_k) - m_k(p_k) + m_k(p_k) - \theta(x_k + p_k) \\ & \geq & pred(p_k) - |m_k(p_k) - \theta(x_k + p_k)|, \end{array}$$

and this along with (3.26) gives the thesis. □

Next lemma shows that the progress $pred(p_k)$ that can be made at iteration k on the model m_k, is at least proportional to $\|D_k^{1/2}\nabla\theta_k\|$. This quantity is an indication of the amount by which x_k violates (2.26).

Lemma 3.4 *If p_k satisfies (3.16) then*

$$pred(p_k) \geq \frac{\beta_1}{2}\|D_k^{1/2}\nabla\theta_k\| \min\left\{\frac{\Delta_k}{\|D_k^{1/2}\|}, \frac{\|D_k^{1/2}\nabla\theta_k\|}{\|D_k^{1/2}(J_k{}^T J_k + \mu_k I_n)D_k^{1/2}\|}, \frac{\|D_k^{1/2}\nabla\theta_k\|}{\|\nabla\theta_k\|_\infty}\right\}. \tag{3.27}$$

Proof. First, we note that (3.18) implies that $\|D_k^{1/2}\nabla\theta_k\| \neq 0$ and $\|\nabla\theta_k\|_\infty \neq 0$. By (3.16) we have $pred(p_k) \geq \beta_1\, pred(p_k^C)$. Hence, to prove (3.27) we determine a lower bound for $pred(p_k^C)$.

Note that from (3.13), p_k^C is given by

$$p_k^C = \left\{\begin{array}{ll} \hat{c}_k d_k & \text{if} \quad x_k + \hat{c}_k d_k \in \Omega, \\ \lambda_k d_k & \text{otherwise,} \end{array}\right.$$

where d_k, $\hat{c}_k$ and λ_k are given in (2.27), (3.14) and (3.15) respectively. If $p_k^C = \hat{c}_k d_k$ and $\hat{c}_k = \dfrac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|J_k d_k\|^2 + \mu_k\|d_k\|^2}$, then

$$\begin{array}{rcl} pred(p_k^C) & = & \hat{c}_k\|D_k^{1/2}\nabla\theta_k\|^2 - \dfrac{1}{2}\hat{c}_k^2(\|J_k d_k\|^2 + \mu_k\|d_k\|^2) = \dfrac{1}{2}\dfrac{\|D_k^{1/2}\nabla\theta_k\|^4}{\|J_k d_k\|^2 + \mu_k\|d_k\|^2} \\ & = & \dfrac{1}{2}\dfrac{\|D_k^{1/2}\nabla\theta_k\|^4}{(D_k^{1/2}\nabla\theta_k)^T(D_k^{1/2}(J_k^T J_k + \mu_k I_n)D_k^{1/2})(D_k^{1/2}\nabla\theta_k)} \\ & \geq & \dfrac{1}{2}\dfrac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|D_k^{1/2}(J_k^T J_k + \mu_k I_n)D_k^{1/2}\|}. \end{array} \tag{3.28}$$

On the other hand, if $p_k^C = \hat{c}_k d_k$ and $\hat{c}_k = \Delta_k/\|d_k\|$, then $\hat{c}_k \leq \dfrac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|J_k d_k\|^2 + \mu_k\|d_k\|^2}$ and we obtain

$$\begin{aligned} pred(p_k^C) &= \hat{c}_k(\|D_k^{1/2}\nabla\theta_k\|^2 - \frac{1}{2}\hat{c}_k(\|J_k d_k\|^2 + \mu_k\|d_k\|^2)) \\ &\geq \frac{1}{2}\hat{c}_k\|D_k^{1/2}\nabla\theta_k\|^2 \geq \frac{1}{2}\Delta_k\frac{\|D_k^{1/2}\nabla\theta_k\|}{\|D_k^{1/2}\|}. \qquad (3.29) \end{aligned}$$

Now, let us consider the case $p_k^C = \lambda_k d_k$, where λ_k is defined in (3.15). Since by construction $\lambda_k \leq \hat{c}_k \leq \dfrac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|J_k d_k\|^2 + \mu_k\|d_k\|^2}$ and that by [11, Lemma 3.1] $\lambda_k \geq 1/\|\nabla\theta_k\|_\infty$, we get

$$\begin{aligned} pred(p_k^C) &= \lambda_k\|D_k^{1/2}\nabla\theta_k\|^2 - \frac{1}{2}\lambda_k^2(\|J_k d_k\|^2 + \mu_k\|d_k\|^2) \\ &\geq \frac{1}{2}\lambda_k\|D_k^{1/2}\nabla\theta_k\|^2 \geq \frac{1}{2}\frac{\|D_k^{1/2}\nabla\theta_k\|^2}{\|\nabla\theta_k\|_\infty}. \qquad (3.30) \end{aligned}$$

Thus $pred(p_k) \geq \beta_1 pred(p_k^C)$, (3.28), (3.29) and (3.30) yield the thesis. □

We next show that each iteration k of the `TREBO` method is well-defined.

Lemma 3.5 *Let Assumption 1 hold. Then condition (3.17) is satisfied in a finite number of trials.*

Proof. First we need to bound $\|p_k\|$, where p_k is formed in Step 6 of Algorithm 3.1. By $\|p_{tr}\| \leq \Delta_k$, (3.22) yields $\|\bar{p}_{tr}\| \leq \Delta_k$ and from the definition of the Cauchy point we have $\|p_k^C\| \leq \Delta_k$. By construction p_k satisfies (3.16) and since either $p_k = \bar{p}_{tr}$ or p_k is a convex combination of p_k^C and $\bar{p}_{tr}$, we conclude

$$\|p_k\| \leq \Delta_k. \qquad (3.31)$$

From assumption (3.18) it follows $\|D_k^{1/2}\nabla\theta_k\| \neq 0$ and $\|\nabla\theta_k\|_\infty \neq 0$. Hence, letting

$$\Delta_k \leq \|D_k^{1/2}\| \min\left\{\frac{\|D_k^{1/2}\nabla\theta_k\|}{\|D_k^{1/2}(J_k^T J_k + \mu_k I_n)D_k^{1/2}\|}, \frac{\|D_k^{1/2}\nabla\theta_k\|}{\|\nabla\theta_k\|_\infty}\right\},$$

(3.27) implies $\Delta_k \leq \tilde{C}_k\, pred(p_k)$, with $\tilde{C}_k = 2\|D_k^{1/2}\|/(\beta_1\|D_k^{1/2}\nabla\theta_k\|)$ and from (3.31) we get

$$\|p_k\| \leq \tilde{C}_k\, pred(p_k). \qquad (3.32)$$

Then, (3.24), (3.31) and (3.32) yield

$$\begin{aligned} ared(p_k) &\geq pred(p_k) - \left(\frac{1}{2}\mu_k\Delta_k + \gamma_D\,\|\Theta_k\|\,\Delta_k + \frac{1}{2}\gamma_D^2\,\Delta_k^3\right)\|p_k\|, \\ &\geq g_k(\Delta_k)\, pred(p_k), \qquad (3.33) \end{aligned}$$

where

$$g_k(\Delta_k) = 1 - \tilde{C}_k \left(\frac{1}{2}\mu_k \Delta_k + \gamma_D \|\Theta_k\| \Delta_k + \frac{1}{2}\gamma_D^2 \Delta_k^3 \right). \tag{3.34}$$

By $g_k(0) = 1$, $\beta_2 \in (0,1)$, and the continuity of the function g_k, there exists a $\tilde{\Delta}_k > 0$ such that $g_k(\Delta_k) \geq \beta_2$ when $\Delta_k \leq \tilde{\Delta}_k$. Then, if

$$\Delta_k \leq \min \left\{ \tilde{\Delta}_k, \frac{\|D_k^{1/2}\| \, \|D_k^{1/2} J_k^T \Theta_k\|}{\|D_k^{1/2}(J_k^T J_k + \mu_k I_n) D_k^{1/2}\|}, \frac{\|D_k^{1/2}\| \, \|D_k^{1/2} J_k^T \Theta_k\|}{\|\nabla\theta_k\|_\infty} \right\},$$

condition (3.17) is met. □

3.3.2 Global convergence

Under Assumptions 1 and 2, the `TREBO-GN` and `TREBO-LM` methods are globally convergent. The convergence analysis is provided in the following theorem where we prove that independently of the choice of the initial guess x_0, the limit points of the sequence $\{x_k\}$ generated by the `TREBO` method are stationary points for the problem (BCLS).

Theorem 3.1 *Let Assumptions 1 and 2 hold and $\{x_k\}$ be the sequence generated by the* `TREBO` *method. Then every limit point of the sequence $\{x_k\}$ is a first-order stationary point for the problem (BCLS), i.e.*

$$\lim_{k\to\infty} \|D_k \nabla\theta_k\| = 0.$$

Proof. Since $\{x_k\}$ is bounded, there exists a constant $\chi_D > 0$ such that $\|D_k^{1/2}\| < \chi_D$ for all k. Hence it suffices to prove that $\lim_{k\to\infty} \|D_k^{1/2}\nabla\theta_k\| = 0$ to obtain the thesis. From Assumptions 1, the gradient $\nabla\theta(x)$ is Lipschitz continuous in L [59, p. 295]. Moreover by Assumption 2 there exists a positive scalar χ_g such that $\|\nabla\theta_k\|_\infty < \chi_g$. Further by construction $\mu_k \leq \mu_0$ for all k. First, we will prove that

$$\liminf_{k\to\infty} \|D_k^{1/2}\nabla\theta_k\| = 0. \tag{3.35}$$

We will proceed by contradiction. Assume that there exists $\epsilon > 0$ such that

$$\liminf_{k\to\infty} \|D_k^{1/2}\nabla\theta_k\| > \epsilon.$$

This implies that there exists $\bar{k}$ such that $\|D_k^{1/2}\nabla\theta_k\| > \epsilon$ whenever $k > \bar{k}$. Assume $k > \bar{k}$. The sequence $\{\theta_k\}$ is monotone decreasing and bounded from below. Then, it is convergent and $\lim_{k\to\infty}(\theta_k - \theta_{k+1}) = 0$. By construction, at each iteration (3.17) is satisfied, i.e.

$$\theta_k - \theta_{k+1} \geq \beta_2 pred(p_k).$$

Then, by (3.27)

$$\theta_k - \theta_{k+1} \geq \frac{\beta_1 \beta_2}{2} \epsilon \min \left\{ \frac{\Delta_k}{\chi_D}, \frac{\epsilon}{\chi_D^2(\chi_L^2 + \mu_0)}, \frac{\epsilon}{\chi_g} \right\}, \tag{3.36}$$

with χ_L given in Assumption 2 and $\lim_{k\to\infty}(\theta_k - \theta_{k+1}) = 0$ implies $\lim_{k\to\infty}\Delta_k = 0$. Then, there exists $\hat{k} > \bar{k}$ such that $\Delta_k \le \chi_D \min\{\epsilon/(\chi_D^2(\chi_L^2 + \mu_0)), \epsilon/\chi_g\}$ when $k \ge \hat{k}$. Assume $k > \hat{k}$.

Using (3.33) and (3.34) and taking into account that $\tilde{C}_k \le 2\chi_D/(\beta_1\epsilon)$, we get the following inequality

$$ared(p_k) \ge g(\Delta_k)pred(p_k),$$

where

$$g(\Delta_k) = 1 - \frac{2\chi_D}{\beta_1\epsilon}\left(\frac{1}{2}\mu_0\Delta_k + \gamma_D\,\|\Theta_0\|\,\Delta_k + \frac{1}{2}\gamma_D^2\,\Delta_k^3\right).$$

Since $g(0) = 1$, g is continuous and $\beta_2 \in (0,1)$, there exists a $\tilde{\Delta} > 0$, independent from k, such that $g(\Delta_k) \ge \beta_2$ when $\Delta_k \le \tilde{\Delta}$. Consequently, $ared(p_k) \ge \beta_2 pred(p_k)$ holds for

$$\Delta_k \le \min\left\{\tilde{\Delta}, \frac{\epsilon}{\chi_D(\chi_L^2 + \mu_0)}, \frac{\epsilon\chi_D}{\chi_g}\right\}.$$

Thus, letting $\bar{\Delta}_k$ be the initial value of Δ_k in Algorithm 3.1, at termination of the k-th iteration, we have either $\Delta_k = \bar{\Delta}_k \ge \Delta_{min}$ or

$$\Delta_k \ge \min\left\{\delta\tilde{\Delta}, \frac{\epsilon}{\chi_D(\chi_L^2 + \mu_0)}, \frac{\epsilon\chi_D}{\chi_g}\right\}.$$

Thus, Δ_k is bounded away from zero. This is a contradiction and we must have $\text{liminf}_{k\to\infty}\|D_k^{1/2}\nabla\theta_k\| = 0$.

Finally we prove that $\lim_{k\to\infty}\|D_k^{1/2}\nabla\theta_k\| = 0$. Now we assume that there exists a sequence $\{m_i\}$ such that $\|D_{m_i}^{1/2}\nabla\theta_{m_i}\| \ge \epsilon_1$ for some $\epsilon_1 \in (0,1)$. By using (3.35), we can state that for any $\epsilon_2 \in (0,\epsilon_1)$ there exists a subsequence of $\{m_i\}$, without loss of generality we assume it is the full sequence, and a sequence $\{l_i\}$ such that

$$\|D_k^{1/2}\nabla\theta_k\| \ge \epsilon_2, \qquad m_i \le k < l_i \qquad \|D_{l_i}^{1/2}\nabla\theta_{l_i}\| < \epsilon_2. \tag{3.37}$$

Then (3.36) yields

$$\theta_k - \theta_{k+1} \ge \frac{1}{2}\beta_1\,\beta_2\,\epsilon_2 \min\left\{\frac{\Delta_k}{\chi_D}, \frac{\epsilon_2}{\chi_D^2(\chi_L^2 + \mu_0)}, \frac{\epsilon_2}{\chi_g}\right\}, \quad m_i \le k < l_i,$$

and since from (3.31) we have $\|x_{k+1} - x_k\| \le \Delta_k$, we can conclude that

$$\theta_k - \theta_{k+1} \ge \frac{1}{2}\beta_1\,\beta_2\,\epsilon_2 \min\left\{\frac{\|x_{k+1} - x_k\|}{\chi_D}, \frac{\epsilon_2}{\chi_D^2(\chi_L^2 + \mu_0)}, \frac{\epsilon_2}{\chi_g}\right\}, \quad m_i \le k < l_i. \tag{3.38}$$

Since $\lim_{k\to\infty}(\theta_k - \theta_{k+1}) = 0$, from (3.38) we have

$$\theta_k - \theta_{k+1} \ge \epsilon_3\|x_{k+1} - x_k\|, \quad m_i \le k < l_i, \tag{3.39}$$

for i sufficiently large and $\epsilon_3 = \frac{1}{2}\beta_1\,\beta_2\,\epsilon_2/\chi_D$. Thus, using the triangle inequality we get

$$\theta_{m_i} - \theta_{k_i} \ge \epsilon_3\|x_{m_i} - x_{k_i}\|, \quad m_i \le k_i \le l_i, \tag{3.40}$$

and we can conclude that $\|x_{m_i} - x_{k_i}\|$ tends to zero. Moreover, from the Lipschitz continuity of $\nabla\theta$ and the fact that $\|x_{m_i} - x_{k_i}\|$ tends to zero, it follows

$$\|\nabla\theta_{m_i} - \nabla\theta_{k_i}\| \leq \epsilon_2, \tag{3.41}$$

for i sufficiently large.

Without loss of generality, assume that the full sequence $\{x_{l_i}\}$ converges to a point, say x^*. From (3.40) we have that $\{x_{m_i}\}$ converges to x^*.

If $(\nabla\theta(x^*))_j \neq 0$ for some $j \in \{1, \dots, n\}$, then (2.25) implies $|(v_{m_i})_j - (v_{l_i})_j| \leq |(x_{m_i})_j - (x_{l_i})_j|$ for i sufficiently large. Consequently $\lim_{i\to\infty} \|(D_{m_i}^{1/2} - D_{l_i}^{1/2}) = 0$ and therefore

$$\|(D_{m_i}^{1/2} - D_{l_i}^{1/2})\nabla\theta_{l_i}\| \leq \epsilon_2, \tag{3.42}$$

for i sufficiently large. Finally, from $\|D_{m_i}^{1/2}\nabla\theta_{m_i}\| \geq \epsilon_1$, (3.37), (3.41), (3.42) and

$$\begin{aligned} \|D_{m_i}^{1/2}\nabla\theta_{m_i}\| \quad \leq \quad & \|D_{m_i}^{1/2}\| \, \|\nabla\theta_{m_i} - \nabla\theta_{l_i}\| + \\ & \|(D_{m_i}^{1/2} - D_{l_i}^{1/2})\nabla\theta_{l_i}\| + \|D_{l_i}^{1/2}\nabla\theta_{l_i}\|, \end{aligned}$$

we get

$$\epsilon_1 \leq (\chi_D + 2)\,\epsilon_2,$$

i.e. a contradiction since $\epsilon_2 \in (0, \epsilon_1)$ can be arbitrarily small. □

Since a limit point x^* may be such that $\|\Theta(x^*)\| > 0$, we list the cases where the limit points are zero-residual solutions to problem (BCLS).

Theorem 3.2 *Let Assumptions 1 and 2 hold and $\{x_k\}$ be the sequence generated by the* `TREBO` *method.*

i. If x^* is a limit point of $\{x_k\}$ and $\|\Theta(x^*)\| = 0$, then all the limit points of $\{x_k\}$ are zero-residual solutions to problem (BCLS).

ii. If the problem is either square or underdetermined and x^* is a limit point of $\{x_k\}$ such that $x^* \in int(\Omega)$ and $J(x^*)$ has full rank, then x^* is such that $\|\Theta(x^*)\| = 0$.

Proof. i. The sequence $\{\theta_k\}$ is monotone decreasing and bounded from below; hence it converges. Since $\|\Theta(x^*)\| = 0$, then $\lim_{k\to\infty} \theta_k = 0$.

ii. From Theorem 3.1, x^* satisfies $\|D(x^*)\nabla\theta(x^*)\| = 0$. Since $x^* \in int(\Omega)$ it follows $J(x^*)^T\Theta(x^*) = 0$. Then, $rank(J(x^*)) = m$ yields $\|\Theta(x^*)\| = 0$. □

3.3.3 Assumptions for local convergence

Under further assumptions, we are able to carry out the local convergence analysis. We let S be the set of zero-residual solutions to problem (BCLS), $d(x, \mathcal{S})$ denote the distance from the point x to the set $\mathcal{S}$ and $[x]_{\mathcal{S}} \in \mathcal{S}$ be such that $\|x - [x]_{\mathcal{S}}\| = d(x, \mathcal{S})$, i.e.

$$\mathcal{S} = \{y \in \Omega : \|\Theta(y)\| = 0\}, \tag{3.43}$$

$$d(x, \mathcal{S}) = \inf\{\|x - y\|, y \in \mathcal{S}\}, \qquad [x]_{\mathcal{S}} = \underset{y\in\mathcal{S}}{\text{argmin}} \;\|x - y\|. \tag{3.44}$$

We make the following assumptions.

Assumption 3 *The zero-residual solution set $\mathcal{S}$ of problem (BCLS) is nonempty. The sequence $\{x_k\}$ generated by the* `TREBO` *method has a limit point $x^* \in \mathcal{S}$.*

It is important to remark that due to Assumption 3 and Theorem 3.2 we know that $\lim_{k\to\infty} \|\Theta_k\| = 0$.

The next lemma provides two useful properties that are a direct consequence of Assumptions 1, 2 and 3.

Lemma 3.6 *Let Assumptions 1, 2 and 3 hold. Then there exists a constant $\epsilon > 0$ such that for $x \in B_\epsilon(x^*)$*

$$x \in L \textit{ and } [x]_{\mathcal{S}} \in L, \tag{3.45}$$

$$\|\Theta(x)\| \leq \chi_L \; d(x, \mathcal{S}). \tag{3.46}$$

Proof. Let r be the scalar given in Assumption 1 and $\epsilon < r/2$. Since from Assumption 3 x^* is a limit point of the sequence $\{x_k\}$, there exists k such that $\|x_k - x^*\| \leq r - 2\epsilon$. Then, if $x \in B_\epsilon(x^*)$ we have $\|x - x_k\| \leq \|x - x^*\| + \|x^* - x_k\| \leq r - \epsilon$, i.e. $x \in L$. Further, let $[x]_{\mathcal{S}}$ as in (3.44). Then, $\|\,[x]_{\mathcal{S}} - x_k\| \leq \|\,[x]_{\mathcal{S}} - x\| + \|x - x_k\| \leq \|x^* - x\| + \|x - x_k\| \leq r$ i.e. $[x]_{\mathcal{S}} \in L$.

The second part of the thesis follows from $\Theta(x) = \Theta(x) - \Theta([x]_{\mathcal{S}})$, (3.20) and (3.44). □

To prove that the sequence $\{x_k\}$ is q-quadratically convergent, we distinguish between the `TREBO-GN` and `TREBO-LM` method. In particular, for the `TREBO-LM` method we assume an error bound condition in a neighbourhood of a limit point $x^* \in \mathcal{S}$. For the `TREBO-GN` method we require that the Jacobian matrix J is full rank at $x^* \in \mathcal{S}$. These assumptions are stated below.

Assumption 4 (Error bound condition) *Let $\{x_k\}$ be the sequence generated by the* `TREBO-LM` *method. For a limit point $x^* \in \mathcal{S}$ of $\{x_k\}$, there exists positive constants ρ and α_0 such that*

$$\frac{1}{\alpha_0} \; d(x, S) \leq \|\Theta(x)\| \quad \textit{for all } x \in B_\rho(x^*). \tag{3.47}$$

Assumption 5 (Full rank condition) *Let $\{x_k\}$ be the sequence generated by the* `TREBO-GN` *method. For a limit point $x^* \in \mathcal{S}$ of $\{x_k\}$, the Jacobian $J(x^*)$ is full rank.*

Assumptions 4 and 5 and their relationship deserve some considerations. When (3.47) is satisfied the function $\|\Theta\|$ provides a local error bound for the problem (BCLS) near $x^* \in S$. It is important to remark that (3.47) depends on the point in $\mathcal{S}$ in the sense that it may fail in a neighbourhood of a point in $\mathcal{S}$ different from x^*, see [51, 62].

The local error bound condition is known to be weaker than the standard nonsingularity assumption of $J(x^*)$ in case J is a square matrix. For nonsquare Jacobian matrices, (3.47) is weaker than the nonsingularity assumption of $J(x^*)^T J(x^*)$. It is interesting to note that, if problem (BCLS) is overdetermined and $J(x^*)$ is full rank, then (3.47) is guaranteed on some neighbourhood of x^*, see the forthcoming Lemma 3.7; in this case, x^* is an isolated solution. The converse is not true and Assumption 4 allows the solution set $\mathcal{S}$ to be locally nonunique.

Typically, condition (3.47) is assumed to hold in a region of the form $B_\rho(x^*) \cap \Omega$ [62]. However, in our convergence analysis we need to apply the error bound condition on points that may lie outside Ω. For this reason, we drop the restriction on Ω and consider the assumption (3.47). Although the condition (3.47) is more restrictive when $x \in B_\rho(x^*)$ than in the case where $x \in B_\rho(x^*) \cap \Omega$, it has been shown that such condition is still significantly weaker than the nonsingularity of $J(x^*)^T J(x^*)$, see [48, §3].

In view of these properties, Assumption 4 is used in the analysis of the `TREBO-LM` method and allows for strong convergence properties, i.e. quadratic convergence in the case where $J(x^*)^T J(x^*)$ is singular.

The following lemma shows an important feature of overdetermined problems. If $J(x^*)$ is full rank, then $\|\Theta\|$ is guaranteed to provide a local error bound on some neighbourhood of x^* and x^* is an isolated zero-residual solution to (BCLS).

Lemma 3.7 *Let Assumptions 1, 2 and 3 hold. If $m \geq n$ and $J(x^*)$ is full rank, then there exist positive constants α_0 and ω such that if $x \in B_\omega(x^*)$ then*

$$\frac{1}{\alpha_0}\ d(x, S) = \frac{1}{\alpha_0}\ \|x - x^*\| \leq \|\Theta(x)\|. \tag{3.48}$$

Proof. Let $x \in B_\epsilon(x^*)$ where ϵ is the scalar given in Lemma 3.6. Since $J(x^*)$ is full column rank, then $J(x^*)^+ = (J(x^*)^T J(x^*))^{-1} J(x^*)^T$ and $J(x^*)^+ J(x^*) = I_n$, see Appendix A.4. Also, by Assumption 1 we get

$$\|I_n - J(x^*)^+ J(x)\| \leq \|J(x^*)^+\|\, \|J(x^*) - J(x)\| \leq 2\gamma_D \|J(x^*)^+\|\, \|x - x^*\|.$$

Choosing $\omega < \min\{\epsilon,\, 1/(4\gamma_D \|J(x^*)^+\|)\}$, we have $\|I_n - J(x^*)^+ J(x)\| \leq 1/2$ for $x \in B_\omega(x^*)$. Then, by using the Mean Value Theorem A.2 we obtain

$$\begin{aligned} \|J(x^*)^+ \Theta(x)\| &= \|(x - x^*) - \int_0^1 \left(I_n - J(x^*)^+ J(x^* + t(x - x^*))\right) (x - x^*)\, dt\| \\ &\geq \left(1 - \frac{1}{2}\right) \|x - x^*\|. \end{aligned}$$

Hence

$$\|\Theta(x)\| \geq \frac{\|J(x^*)^+ \Theta(x)\|}{\|J(x^*)^+\|} \geq \frac{1}{2\|J(x^*)^+\|} \|x - x^*\|.$$

This inequality implies that x^* is an isolated zero-residual solution to (BCLS) and reducing ω if necessary, we get $d(x, \mathcal{S}) = \|x - x^*\|$ for $x \in B_\omega(x^*)$. Thus (3.48) is obtained with $\alpha_0 = 2\|J(x^*)^+\|$. □

The following lemma states useful local properties of the Jacobian $J(x)$ and its pseudoinverse $J(x)^+$, if x is sufficiently close to x^* and $J(x^*)$ is full rank.

Lemma 3.8 *Let Assumptions 1, 2 and 3 hold. If $J(x^*)$ is full rank, then there exist positive constants τ and ν such that for $x \in B_\tau(x^*)$*

$$J(x) \ \ \textit{is full rank and} \ \ \|J(x)^+\| \leq \nu. \tag{3.49}$$

Proof. Let ϵ be given in Lemma 3.6 and let $\tau \leq \epsilon$. Fix $x \in B_\tau(x^*)$. Let $q = \min\{m, n\}$ and $\sigma_q(J(x))$ be the smallest singular value of $J(x)$. Since $rank(J(x^*)) = \min\{m, n\}$, we know that $\sigma_q(J(x^*)) > 0$. Using Theorem A.6 and Assumption 1 we get

$$|\sigma_q(J(x)) - \sigma_q(J(x^*))| \leq \|J(x) - J(x^*)\| \leq 2\gamma_D \|x - x^*\|,$$

and consequently

$$\sigma_q(J(x)) \geq \sigma_q(J(x^*)) - 2\gamma_D \|x - x^*\|. \tag{3.50}$$

Then, reducing τ so that $\tau < \min\{\epsilon, \sigma_q(J(x^*))/(2\gamma_D)\}$, (3.50) implies that $J(x)$ is full rank for all $x \in B_\tau(x^*)$. Setting $\nu = 1/(\sigma_q(J(x^*)) - 2\gamma_D\tau)$, by (A.4) we obtain (3.49).□

3.3.4 Analysis of the minimum norm step p_k^N

Let x_k be an iterate of the TREBO method, p_k^N be the step given in (3.7) and in (3.8) and $\bar{p}_k^N$ be defined as

$$\bar{p}_k^N = P_\Omega(x_k + p_k^N) - x_k. \tag{3.51}$$

The steps p_k^N and $\bar{p}_k^N$ play a central role in the asymptotic behaviour of the sequence $\{x_k\}$. Here we provide an analysis of such steps in the vicinity of a limit point $x^* \in \mathcal{S}$. The main provided result is that if x_k is sufficiently close to x^* then the trust-region constraint is inactive and the unconstrained minimizer of the model m_k is the solution of the trust-region problem (3.6).

A result on the distance of a point from the zero-residual solution set $\mathcal{S}$ follows from the contractivity of the projection map P_Ω. From definition (3.44) we have $d(x_k + \bar{p}_k^N, \mathcal{S}) \leq \|x_k + \bar{p}_k^N - [x_k + p_k^N]_\mathcal{S}\| \leq \|x_k + p_k^N - [x_k + p_k^N]_\mathcal{S}\|$, i.e.

$$d(x_k + \bar{p}_k^N, \mathcal{S}) \leq d(x_k + p_k^N, \mathcal{S}). \tag{3.52}$$

The next two lemmas concern with upper bounds on the quantity $d(x_k + \bar{p}_k^N, \mathcal{S})$ that will be crucial for proving the quadratic convergence rate. In particular, Lemma 3.9 refers to the TREBO-LM method and Lemma 3.10 is related to the TREBO-GN method.

Lemma 3.9 *Let Assumptions 1 – 3 and 4 hold. Then for the* TREBO-LM *method there exist positive constants ψ_1 and Γ such that if $x_k \in B_{\psi_1}(x^*)$ then*

$$d(x_k + \bar{p}_k^N, \mathcal{S}) \leq \Gamma\ d(x_k, \mathcal{S})^2. \tag{3.53}$$

Proof. By the definition (3.12) of μ_k, it follows

$$\mu_k = \mu \|\Theta_k\|^2, \tag{3.54}$$

for x_k sufficiently close to x^*. Let ϵ as in Lemma 3.6, ρ as in Assumption 4 and $\psi_1 \leq \min\{\epsilon, \rho\}$ small enough so that if $x_k \in B_{\psi_1}(x^*)$ then (3.54) holds. Fix $x_k \in B_{\psi_1}(x^*)$.

First, we provide an upper bound on the norm of the step p_k^N showing that

$$\|p_k^N\| \leq \alpha_1\, d(x_k, \mathcal{S}), \tag{3.55}$$

for some positive scalar α_1. We prove this fact in the same way as in [48, Lemma 2.3]. In particular, by (3.54), (3.47) and (3.46)

$$\frac{\mu}{\alpha_0^2}\, d(x_k,\mathcal{S})^2 \le \mu_k \le \mu\chi_L^2\, d(x_k,\mathcal{S})^2. \tag{3.56}$$

Also, since p_k^N is the global minimum of the model m_k we have

$$m_k(p_k^N) \le m_k(x_k - [x_k]_{\mathcal{S}}),$$

where $[x_k]_{\mathcal{S}}$ is the closest solution to x_k, see (3.44). Then, by (3.21)

$$\begin{aligned} 2\,m_k(p_k^N) &\le \|J_k(x_k-[x_k]_{\mathcal{S}})+\Theta_k\|^2+\mu_k\|x_k-[x_k]_{\mathcal{S}}\|^2 \\ &\le \gamma_D^2\, d(x_k,\mathcal{S})^4+\mu_k\, d(x_k,\mathcal{S})^2, \end{aligned} \tag{3.57}$$

and by (3.56)

$$\|p_k^N\|^2 \le \frac{2}{\mu_k}\, m_k(p_k^N) \le \left(1+\frac{\alpha_0^2\,\gamma_D^2}{\mu}\right) d(x_k,\mathcal{S})^2.$$

Thus, we obtain (3.55) setting $\alpha_1=\sqrt{1+\alpha_0^2\gamma_D^2/\mu}$. Moreover, note that from (3.55) and (3.56) we get the inequality

$$\mu_k\,\|p_k^N\|^2 \le \mu\,\alpha_1^2\,\chi_L^2\, d(x_k,\mathcal{S})^4. \tag{3.58}$$

Second, we show that if ψ_1 is sufficiently small then

$$x_k+p_k^N \in B_\rho(x^*), \quad x_k+p_k^N \in B_\epsilon(x^*), \quad x_k+p_k^N\in L, \quad [x_k+p_k^N]_{\mathcal{S}}\in L. \tag{3.59}$$

To this end, reduce ψ_1 if necessary so that $\psi_1 \le \min\{\epsilon,\rho\}/(1+\alpha_1)$ and note that (3.55) yields

$$\|x_k+p_k^N-x^*\| \le \|x_k-x^*\|+\|p_k^N\| \le (1+\alpha_1)\psi_1 \le \min\{\epsilon,\rho\}.$$

Then, the last two statements in (3.59) derive from Lemma 3.6.

Finally, to prove (3.53), note that

$$\begin{aligned} \left\|\begin{pmatrix}\Theta(x_k+p_k^N)\\ 0\end{pmatrix}\right\| &\le \left\|\begin{pmatrix}\Theta(x_k+p_k^N)-J_kp_k^N-\Theta_k\\ -\sqrt{\mu_k}\,p_k^N\end{pmatrix}\right\|+\left\|\begin{pmatrix}J_kp_k^N+\Theta_k\\ \sqrt{\mu_k}\,p_k^N\end{pmatrix}\right\| \\ &= \left(\|\Theta(x_k+p_k^N)-J_kp_k^N-\Theta_k\|^2+\mu_k\|p_k^N\|^2\right)^{\frac{1}{2}}+(2m_k(p_k^N))^{\frac{1}{2}}. \end{aligned}$$

Hence, using (3.21), (3.55), (3.58), (3.57) and (3.56) we obtain

$$\|\Theta(x_k+p_k^N)\| \le \eta\, d(x_k,\mathcal{S})^2, \tag{3.60}$$

with $\eta=\sqrt{\alpha_1^4\gamma_D^2+\mu\alpha_1^2\chi_L^2}+\sqrt{\gamma_D^2+\mu\chi_L^2}$. We complete the proof using (3.47) and (3.52) and setting $\Gamma=\alpha_0\eta$. □

An analogous lemma holds for the `TREBO-GN` method.

Lemma 3.10 *Let Assumptions 1 – 3 and 5 hold. Then for the* `TREBO-GN` *method there exist positive constants* ψ_1 *and* Γ *such that if* $x_k \in B_{\psi_1}(x^*)$ *then*

$$d(x_k + \bar{p}_k^N, \mathcal{S}) \le \Gamma\ d(x_k, \mathcal{S})^2. \tag{3.61}$$

Proof. Let τ given by Lemma 3.8, $\psi_1 < \tau$ and fix $x_k \in B_{\psi_1}(x^*)$. Note that by (3.7), (3.49) and (3.46) we get

$$\|p_k^N\| \le \nu\|\Theta_k\| \le \alpha_1 d(x_k, \mathcal{S}), \tag{3.62}$$

with $\alpha_1 = \nu\chi_L$.

Consider the case $m \ge n$. Reduce ψ_1 if necessary so that $\psi_1 < \min\{\tau, \omega\}/(1+\alpha_1)$ where ω is given in Lemma 3.7. Then (3.62) yields

$$\|x_k + p_k^N - x^*\| \le \|x_k - x^*\| + \|p_k^N\| \le (1+\alpha_1)\psi_1 \le \min\{\tau, \omega\}.$$

and then

$$x_k + p_k^N \in B_\omega(x^*), \quad x_k + p_k^N \in B_\tau(x^*), \quad x_k + p_k^N \in L, \quad [x_k + p_k^N]_{\mathcal{S}} \in L.$$

where the last two statements follows from Lemma 3.6.

Setting $\mu_k = 0$ in (3.57), we have $2\,m_k(p_k^N) \le \gamma_D^2 d(x_k, \mathcal{S})^4$. Hence by (3.21) and (3.62) we obtain

$$\begin{aligned} \|\Theta(x_k + p_k^N)\| &\le \|\Theta(x_k + p_k^N) - J_k p_k^N - \Theta_k\| + \|J_k p_k^N + \Theta_k\| \\ &\le \gamma_D \|p_k^N\|^2 + (2\,m_k(p_k^N))^{\frac{1}{2}} \\ &\le \gamma_D(\alpha_1^2 + 1) d(x_k, \mathcal{S})^2. \end{aligned}$$

Using (3.48) and (3.52) this inequality implies (3.61) with $\Gamma = \alpha_0\gamma_D(\alpha_1^2+1)$.

Now let us consider the case $m \le n$. Let $\alpha_2 = \nu\gamma_D$ and reduce ψ_1 so that

$$\psi_1 \le \min\left\{\frac{1}{2\alpha_1\alpha_2}, \frac{\tau}{1+2\alpha_1}\right\}. \tag{3.63}$$

To prove the thesis we need intermediate results. Consider the sequence $\{w_{k+l}\}_l$, $l \ge 0$, of the form

$$w_k = x_k, \quad w_{k+l+1} = w_{k+l} + s_{k+l}^N, \quad l \ge 0, \tag{3.64}$$

with

$$s_{k+l}^N = -J(w_{k+l})^+\,\Theta(w_{k+l}), \ l \ge 0. \tag{3.65}$$

Note that for $l = 0$, we get $s_k^N = p_k^N$ with p_k^N given in (3.7). First, we show that $\{w_{k+l}\} \subseteq B_\tau(x^*)$. Second, we prove that $\{w_{k+l}\}$ has limit point in $\mathcal{S}$. We begin proving that $\{w_{k+l}\} \subseteq B_\tau(x^*)$ by induction. The thesis trivially holds for $w_k = x_k$. Then, we suppose that $w_{k+j} \in B_\tau(x^*)$ for $j \le l$ and show that $w_{k+l+1} \in B_\tau(x^*)$. By (3.65) and (3.49) we get

$$\|s_{k+j}^N\| \le \nu\|\Theta(w_{k+j})\| \quad 1 \le j \le l, \tag{3.66}$$

$$\|J(w_{k+j-1})s_{k+j-1}^N + \Theta(w_{k+j-1})\| = 0, \quad 1 \le j \le l. \tag{3.67}$$

Moreover, from (3.62) it follows

$$\|p_k^N\| \le \alpha_1\psi_1, \tag{3.68}$$

while (3.66), (3.67), (3.64) and (3.21) provide

$$\begin{aligned} \|s_{k+j}^N\| &\le \nu\|\Theta(w_{k+j-1}+s_{k+j-1}^N)-\Theta(w_{k+j-1})-J(w_{k+j-1})s_{k+j-1}^N\| \\ &\le \alpha_2\|s_{k+j-1}^N\|^2, \end{aligned}$$

with $1 \le j \le l$. So by (3.68) and the definition (3.63) of ψ_1 we obtain

$$\begin{aligned} \|s_{k+j}^N\| &\le \alpha_2^{2^j-1}\|p_k^N\|^{2^j} \\ &\le (\alpha_1\alpha_2\psi_1)^{2^j-1}\|p_k^N\| \\ &\le \left(\frac{1}{2}\right)^{2^j-1}\|p_k^N\|. \end{aligned} \tag{3.69}$$

It then follows

$$\|w_{k+l+1}-x^*\| \le \sum_{j=0}^{l}\|w_{k+j+1}-w_{k+j}\|+\|x_k-x^*\| \le \|p_k^N\|\sum_{j=0}^{\infty}\left(\frac{1}{2}\right)^j+\psi_1,$$

and (3.68) yields to

$$\|w_{k+l+1}-x^*\| \le 2\|p_k^N\|+\psi_1 \le (2\alpha_1+1)\psi_1 \le \tau.$$

As a consequence, $\{w_{k+l}\} \subset B_\tau(x^*)$ and w_{k+l} satisfies Lemma 3.6 and Lemma 3.8 for all $l \ge 0$. Further, the conditions (3.66) and (3.67) hold for $j \ge 1$.

Second, we prove that $\{w_{k+l}\}$ is a Cauchy sequence with limit point $\bar{x} \in \mathcal{S}$. In fact, letting $p > q \ge 0$ and proceeding as above we obtain

$$\|w_{k+p}-w_{k+q}\| \le \sum_{j=q}^{p-1}\|s_{k+j}^N\| \le \sum_{j=0}^{\infty}\|s_{k+j}^N\| \le 2\alpha_1\psi_1.$$

Thus, $\{w_{k+l}\}$ is a Cauchy sequence and the limit is denoted as $\bar{x}$. To show that $\bar{x} \in \mathcal{S}$ note that (3.67) and (3.21) yield

$$\|\Theta(w_{k+l+1})\| = \|\Theta(w_{k+l}+s_{k+l}^N)-\Theta(w_{k+l})-J(w_{k+l})s_{k+l}^N\| \le \gamma_D\|s_{k+l}^N\|^2,$$

for $l \ge 0$. Since $s_{k+l}^N = w_{k+l+1}-w_{k+l}$, it follows $\|\Theta(\bar{x})\| = \lim_{l\to\infty}\|\Theta(w_{k+l+1})\| = 0$.

Now we can prove the thesis of the lemma. Note that $\|x_k+p_k^N-\bar{x}\| = \|w_{k+1}-\bar{x}\| \le \sum_{j=1}^{\infty}\|s_{k+j}^N\|$, since

$$w_{k+l} = w_{k+1}+\sum_{j=1}^{l-1}s_{k+j}^N, \quad \lim_{l\to\infty}w_{k+l} = \bar{x},$$

and from the continuity of the norm

$$\begin{aligned}\|w_{k+1} - \bar{x}\| &= \left\|w_{k+1} - \lim_{l\to\infty} w_{k+l}\right\| = \left\|\lim_{l\to\infty} \sum_{j=1}^{l-1} s^N_{k+j}\right\| \\ &\leq \lim_{l\to\infty} \sum_{j=1}^{l-1} \|s^N_{k+j}\| = \sum_{j=1}^{\infty} \|s^N_{k+j}\|,\end{aligned}$$

see [48]. From (3.69) and (3.68) we get

$$\|x_k + p^N_k - \bar{x}\| \leq \alpha_2 \sum_{j=1}^{\infty} (\alpha_2 \|p^N_k\|)^{2^j-2} \|p^N_k\|^2 \leq \alpha_2 \sum_{j=1}^{\infty} (\alpha_1\alpha_2\psi_1)^{2^j-2} \|p^N_k\|^2.$$

Then, in a way analogous to above and using (3.62) we get

$$\|x_k + p^N_k - \bar{x}\| \leq 2\alpha_2 \|p^N_k\|^2 \leq 2\alpha_2\nu^2\|\Theta_k\|^2.$$

Since $d(x_k + p^N_k, \mathcal{S}) \leq \|x_k + p^N_k - \bar{x}\|$, we get

$$d(x_k + p^N_k, \mathcal{S}) \leq \eta \|\Theta_k\|^2. \tag{3.70}$$

with $\eta = 2\alpha_2\nu^2$. Finally, applying (3.52) and (3.46) we easily obtain condition (3.61) with $\Gamma = \eta\chi_L^2$. □

The next lemma shows that if x_k is sufficiently close to x^*, then the trust-region constraint becomes inactive.

Lemma 3.11 *Let Assumptions 1 – 3 and 4 hold. Then for the* TREBO-LM *method there exists $\varsigma > 0$ such that if $x_k \in B_\varsigma(x^*)$ then the trust-region solution p_{tr} in Algorithm 3.1 is the step p^N_k given in (3.8).*

Proof. Let $\psi_1 > 0$ be given in Lemma 3.9 and let $x_k \in B_{\psi_1}(x^*)$. Since $x^* \in \mathcal{S}$ and (3.55) holds, there exists a scalar $\varsigma \leq \psi_1$ sufficiently small so that if $x_k \in B_\varsigma(x^*)$ then $\|p^N_k\| \leq \Delta_{min}$. Namely, the unconstrained minimizer of the quadratic model m_k lies in the trust-region. □

The above result holds for the TREBO-GN method as well.

Lemma 3.12 *Let Assumptions 1 – 3 and 5 hold. Then for the* TREBO-GN *method there exists $\varsigma > 0$ such that if $x_k \in B_\varsigma(x^*)$ then the trust-region solution p_{tr} in Algorithm 3.1 is the step p^N_k given in (3.7).*

Proof. Let $\psi_1 > 0$ be given in Lemma 3.10 and let $x_k \in B_{\psi_1}(x^*)$. Using (3.62) and choosing $\varsigma < \psi_1$ sufficiently small, the proof can be completed as in the proof of Lemma 3.11. □

3.3.5 Convergence of the sequence $\{x_k\}$ and rate of convergence

In this section we first show that if x_k is sufficiently close to x^* then the step $\bar{p}_k^N$ defined in (3.51) satisfies both conditions (3.16) and (3.17) and it is taken to form the new iterate. Second, we prove that the whole sequence $\{x_k\}$ generated by the methods converges to $x^* \in \mathcal{S}$ and the convergence rate is q-quadratic.

We start giving useful asymptotic bounds on quantities that will be used to analyze conditions (3.16) and (3.17).

Lemma 3.13 *Let Assumptions 1, 2 and 3 hold. If $x_k + p_k^N \in L$ and $[x_k + p_k^N]_{\mathcal{S}} \in L$, then*

$$\begin{aligned} \|J_k \bar{p}_k^N + \Theta_k\| &\leq \chi_L d(x_k + p_k^N, \mathcal{S}) + \gamma_D \|p_k^N\|^2, && (3.71)\\ \|\Theta(x_k + \bar{p}_k^N)\|^2 - \|J_k \bar{p}_k^N + \Theta_k\|^2 &\leq \gamma_D^2 \|p_k^N\|^4 + 2\gamma_D \|J_k \bar{p}_k^N + \Theta_k\| \, \|p_k^N\|^2. && (3.72)\end{aligned}$$

Proof. By (3.51) and Lemma 3.2, $x_k + p_k^N \in L$ implies $x_k + \bar{p}_k^N \in L$. Consider the equality

$$J_k \bar{p}_k^N + \Theta_k = \Theta(x_k + \bar{p}_k^N) - \Theta([x_k + p_k^N]_{\mathcal{S}}) + J_k \bar{p}_k^N - (\Theta(x_k + \bar{p}_k^N) - \Theta_k).$$

Then by (3.20), (3.22) and (3.23) we obtain

$$\begin{aligned} \|J_k \bar{p}_k^N + \Theta_k\| &\leq \chi_L \|x_k + \bar{p}_k^N - [x_k + p_k^N]_{\mathcal{S}}\| + \gamma_D \|\bar{p}_k^N\|^2 \\ &\leq \chi_L \|x_k + p_k^N - [x_k + p_k^N]_{\mathcal{S}}\| + \gamma_D \|p_k^N\|^2 \\ &\leq \chi_L d(x_k + p_k^N, \mathcal{S}) + \gamma_D \|p_k^N\|^2, \end{aligned}$$

and the (3.71) is proved.

To prove (3.72) we use the Mean Value Theorem A.2 to get the statement

$$\Theta(x_k + \bar{p}_k^N) = \Theta_k + \int_0^1 J(x_k + t\bar{p}_k^N)\, \bar{p}_k^N \, dt + J_k \bar{p}_k^N - J_k \bar{p}_k^N.$$

Hence,

$$\begin{aligned} \|\Theta(x_k + \bar{p}_k^N)\|^2 &= \|J_k \bar{p}_k^N + \Theta_k\|^2 + \| \int_0^1 (J(x_k + t\bar{p}_k^N) - J_k)\, \bar{p}_k^N \, dt\|^2 \\ &\quad +2 \left(\int_0^1 (J(x_k + t\bar{p}_k^N) - J_k)\, \bar{p}_k^N \, dt \right)^T \left(J_k \bar{p}_k^N + \Theta_k \right), \end{aligned}$$

and consequently by (3.19) and (3.22)

$$\|\Theta(x_k + \bar{p}_k^N)\|^2 - \|J_k \bar{p}_k^N + \Theta_k\|^2 \leq \gamma_D^2 \|p_k^N\|^4 + 2\gamma_D \|J_k \bar{p}_k^N + \Theta_k\| \, \|p_k^N\|^2.$$

□

Now we prove that if x_k is sufficiently close to x^* then $x_{k+1} = x_k + \bar{p}_k^N$.

Lemma 3.14 *Let Assumptions 1 – 3 and 4 hold. Then for the* `TREBO-LM` *method there exists $\psi_2 > 0$ such that if $x_k \in B_{\psi_2}(x^*)$ the iterate x_{k+1} has the form*

$$x_{k+1} = x_k + \bar{p}_k^N.$$

Proof. Let ς as in Lemma 3.11 and suppose $\psi \le \varsigma$. Fix $x_k \in B_\psi(x^*)$. Then, the step p_k^N is the solution to the trust-region problem (3.6) and (3.59) holds.

To show the thesis we will prove that $\bar{p}_k^N$ defined in (3.51) satisfies both conditions (3.16) and (3.17). First consider condition (3.16). If $\bar{p}_k^N = p_k^N$ then (3.16) trivially follows by $m_k(p_k^N) < m_k(p_k^C)$. If $\bar{p}_k^N \ne p_k^N$ note that by (3.47) we have

$$\rho_c(\bar{p}_k^N) \ge \frac{m_k(0) - m_k(\bar{p}_k^N)}{m_k(0)} \ge 1 - \alpha_0^2 \, \frac{\|J_k\bar{p}_k^N + \Theta_k\|^2 + \mu_k\|\bar{p}_k^N\|^2}{d(x_k,\mathcal{S})^2}. \tag{3.73}$$

Thus, to investigate condition (3.16) we need to estimate $\|J_k\bar{p}_k^N + \Theta_k\|$. From (3.71), (3.47), (3.60) and (3.55) we obtain

$$\|J_k\bar{p}_k^N + \Theta_k\| \le \alpha_0\chi_L\|\Theta(x_k + p_k^N)\| + \gamma_D\|p_k^N\|^2 \le \varphi \; d(x_k,\mathcal{S})^2, \tag{3.74}$$

where $\varphi = (\alpha_0\eta\chi_L + \gamma_D\alpha_1^2)$. Thus, combining (3.73), (3.74) and (3.58)

$$\rho_c(\bar{p}_k^N) \ge 1 - \alpha_0^2(\varphi^2 + \mu\,\alpha_1^2\,\chi_L^2)d(x_k,\mathcal{S})^2 \ge 1 - \alpha_0^2(\varphi^2 + \mu\,\alpha_1^2\,\chi_L^2)\|x_k - x^*\|^2,$$

i.e. $\bar{p}_k^N$ satisfies condition (3.16) if x_k is sufficiently close to x^*.

Second, we focus on condition (3.17). Let us assume that $\bar{p}_k^N$ satisfies condition (3.16). From (3.72) and (3.74) we have

$$\|\Theta(x_k + \bar{p}_k^N)\|^2 - \|J_k\bar{p}_k^N + \Theta_k\|^2 \quad \le \quad \left(\gamma_D^2\|p_k^N\|^2 + 2\varphi\gamma_D\, d(x_k,\mathcal{S})^2\right)\|p_k^N\|^2. \tag{3.75}$$

Furthermore, using (3.47), (3.74) and the fact that $x_k \in B_\psi(x^*)$ we have

$$\|\Theta_k\|^2 - \|J_k\bar{p}_k^N + \Theta_k\|^2 \ge \left(\frac{1}{\alpha_0^2} - \varphi^2 d(x_k,\mathcal{S})^2\right) d(x_k,\mathcal{S})^2 \ge \left(\frac{1}{\alpha_0^2} - \varphi^2\psi^2\right) d(x_k,\mathcal{S})^2.$$

Reduce ψ if needed so that $1/\alpha_0^2 - \varphi^2\psi^2 \ge 1/(2\alpha_0^2)$. This fact, (3.75) and (3.55) yield

$$\begin{aligned}
\rho_\theta(\bar{p}_k^N) &= 1 - \frac{\|\Theta(x_k + \bar{p}_k^N)\|^2 - \|J_k\bar{p}_k^N + \Theta_k\|^2}{\|\Theta_k\|^2 - \|J_k\bar{p}_k^N + \Theta_k\|^2} \\
&\ge 1 - \frac{2\alpha_0^2\left(\gamma_D^2\|p_k^N\|^2 + 2\varphi\gamma_D\, d(x_k,\mathcal{S})^2\right)\|p_k^N\|^2}{d(x_k,\mathcal{S})^2} \\
&\ge 1 - 2\alpha_0^2\alpha_1^2\gamma_D\left(\alpha_1^2\gamma_D + 2\varphi\right) d(x_k,\mathcal{S})^2.
\end{aligned}$$

Finally, by (3.44)

$$\rho_\theta(\bar{p}_k^N) \ge 1 - 2\alpha_0^2\alpha_1^2\gamma_D\left(\alpha_1^2\gamma_D + 2\varphi\right)\|x_k - x^*\|^2.$$

Hence, $\bar{p}_k^N$ satisfies both the conditions (3.16) and (3.17) if x_k is sufficiently close to x^*, i.e. if $x_k \in B_{\psi_2}(x^*)$ for some $\psi_2 \le \psi$. □

An analogous result holds for the `TREBO-GN` method.

Lemma 3.15 *Let Assumptions 1 – 3 and 5 hold. Then for the* `TREBO-GN` *method there exists $\psi_2 > 0$ such that if $x_k \in B_{\psi_2}(x^*)$ the iterate x_{k+1} has the form*

$$x_{k+1} = x_k + \bar{p}_k^N.$$

Proof. Let ς as in Lemma 3.12 and suppose $\psi \leq \varsigma$. Fix $x_k \in B_\psi(x^*)$. Then, the step p_k^N is the solution to the trust-region problem (3.6) and $x_k + p_k^N$, $[x_k + p_k^N]_{\mathcal{S}} \in L$.

Consider the case $m \geq n$. The thesis can be proved following the lines of the proof of Lemma 3.14, using (3.48) in place of (3.47) and setting $\mu = 0$.

Consider the case $m \leq n$. To show the thesis we will prove that $\bar{p}_k^N$ defined in (3.51) satisfies both condition (3.16) and (3.17). First consider condition (3.16). If $\bar{p}_k^N = p_k^N$, condition (3.16) trivially follows as $m_k(p_k^N) < m_k(p_k^C)$. If $\bar{p}_k^N \neq p_k^N$ note that $m_k(0) - m_k(p_k^C) \leq m_k(0)$ and

$$\rho_c(\bar{p}_k^N) \geq 1 - \frac{\|J_k\bar{p}_k^N + \Theta_k\|^2}{\|\Theta_k\|^2}. \tag{3.76}$$

Using (3.71), (3.70) and (3.62) yield

$$\|J_k\bar{p}_k^N + \Theta_k\| \leq (\chi_L\eta + \gamma_D\nu^2)\|\Theta_k\|^2. \tag{3.77}$$

Thus, (3.76), (3.77) and (3.46) give

$$\rho_c(\bar{p}_k^N) \geq 1 - (\chi_L\eta + \gamma_D\nu^2)^2\,\|\Theta_k\|^2 \geq 1 - \chi_L^2\,(\chi_L\eta + \gamma_D\nu^2)^2\,d(x_k,\mathcal{S})^2,$$

i.e. $\bar{p}_k^N$ satisfies condition (3.16) if x_k is sufficiently close to x^*.

Now, let us assume that $\bar{p}_k^N$ satisfies condition (3.16). To prove that $\bar{p}_k^N$ satisfies (3.17), use (3.77), (3.46) and since $x_k \in B_\psi(x^*)$ we have

$$\begin{aligned}\|\Theta_k\|^2 - \|J_k\bar{p}_k^N + \Theta_k\|^2 &\geq \left(1 - (\chi_L\eta + \gamma_D\nu^2)^2\|\Theta_k\|^2\right)\|\Theta_k\|^2\\ &\geq \left(1 - (\chi_L\eta + \gamma_D\nu^2)^2\chi_L^2 d(x_k,\mathcal{S})^2\right)\|\Theta_k\|^2\\ &\geq \left(1 - (\chi_L\eta + \gamma_D\nu^2)^2\chi_L^2\psi^2\right)\|\Theta_k\|^2. \end{aligned}\tag{3.78}$$

Reduce ψ if needed so that $1 - (\chi_L\eta + \gamma_D\nu^2)^2\chi_L^2\psi^2 > 0$. Using (3.72) and (3.78) we have

$$\rho_\theta(\bar{p}_k^N) \geq 1 - \frac{\left(\gamma_D^2\|p_k^N\|^2 + 2\gamma_D\|J_k\bar{p}_k^N + \Theta_k\|\right)\|p_k^N\|^2}{\left(1 - (\chi_L\eta + \gamma_D\nu^2)^2\chi_L^2\psi^2\right)\|\Theta_k\|^2},$$

and from (3.62), (3.77) it follows

$$\rho_\theta(\bar{p}_k^N) \;\geq\; 1 - \frac{\nu^4\gamma_D^2 + 2\nu^2\gamma_D(\chi_L\eta + \gamma_D\nu^2)}{1 - (\chi_L\eta + \gamma_D\nu^2)^2\chi_L^2\psi^2}\,\|\Theta_k\|^2,$$

i.e. $\bar{p}_k^N$ satisfies condition (3.17) if x_k is sufficiently close to x^*.

Hence there exists $\psi_2 < \psi$ such that if $x_k \in B_{\psi_2}(x^*)$ then $x_{k+1} = x_k + \bar{p}_k^N$. □

Now we provide the main result on the behavior of the sequence $\{x_k\}$.

Theorem 3.3 *Let Assumptions 1, 2 and 3 hold. Moreover let Assumption 4 and Assumption 5 hold for the* `TREBO-LM` *method and the* `TREBO-GN` *method respectively. Then, the sequence $\{x_k\}$ generated by the* `TREBO` *method converges to x^* q-quadratically.*

Proof. Consider the TREBO-LM method first. Let ψ_2 as in Lemma 3.14 and $\zeta \le \min\{\psi_2/(1+2\alpha_1), 1/(2\Gamma)\}$ and Γ is given in (3.53). Since x^* is a limit point of $\{x_k\}$, there exists x_k such that $x_k \in B_\zeta(x^*)$.

We begin showing that if $x_k \in B_\zeta(x^*)$ then $x_l \in B_{\psi_2}(x^*)$ for $l > k$. We proceed by induction. First, we show that $x_{k+1} \in B_{\psi_2}(x^*)$. In fact, by (3.22) we have $\|x_{k+1} - x^*\| = \|x_k + \bar{p}_k^N - x^*\| \le \zeta + \|p_k^N\|$. Thus by (3.55) and the definition of ζ, we get $\|x_{k+1} - x^*\| \le (1+\alpha_1)\zeta \le \psi_2$. Second, we assume $x_{k+1}, \ldots, x_{k+m-1} \in B_{\psi_2}(x^*)$, and show that $x_{k+m} \in B_{\psi_2}(x^*)$. From (3.53) it follows

$$d(x_{k+l}, \mathcal{S}) \le \Gamma\ d(x_{k+l-1}, \mathcal{S})^2 \le \cdots \le \Gamma^{2^l-1} d(x_k, \mathcal{S})^{2^l} \le \Gamma^{2^l-1}\zeta^{2^l} \le \zeta \left(\frac{1}{2}\right)^{2^l-1},$$

for $l = 1, \ldots, m$, where the last inequality is due to the choice of ζ. Thus,

$$\begin{aligned} \|x_{k+m} - x^*\| &\le \|x_{k+m} - x_{k+m-1}\| + \cdots + \|x_k - x^*\| \\ &\le \sum_{l=0}^{m-1} \|\bar{p}_{k+l}^N\| + \zeta \\ &\le \alpha_1 \sum_{l=0}^{m-1} d(x_{k+l}, \mathcal{S}) + \zeta, \end{aligned}$$

where the last inequality follows from (3.55), and

$$\|x_{k+m} - x^*\| \le (\alpha_1 \sum_{l=0}^{m-1} \left(\frac{1}{2}\right)^{2^l-1} + 1)\zeta \le (\alpha_1 \sum_{l=0}^{\infty} \left(\frac{1}{2}\right)^{l} + 1)\zeta = (2\,\alpha_1 + 1)\zeta \le \psi_2.$$

By Lemma 3.14 we have $x_{k+l} = x_{k+l-1} + \bar{p}_{k+l-1}^N$ for $l > 0$. Moreover, letting $p > q \ge k$ we have

$$\|x_p - x_q\| \le \sum_{l=q}^{p-1} \|\bar{p}_l^N\| \le \alpha_1 \sum_{l=0}^{\infty} \left(\frac{1}{2}\right)^l \zeta = 2\,\alpha_1\zeta.$$

This means that $\{x_k\}$ is a Cauchy sequence and hence it converges. Since x^* is a limit point we conclude $\lim_{k\to\infty} x_k = x^*$.

To establish the convergence rate of $\{x_k\}$, let k sufficiently large so that $x_{k+j+1} \in B_{\psi_2}(x^*)$ for $j \ge 0$. By (3.55) and (3.53) we obtain

$$\|\bar{p}_{k+j+1}^N\| \le \alpha_1\ d(x_{k+j+1}, \mathcal{S}) \le \alpha_1\Gamma\ d(x_{k+j}, \mathcal{S})^2.$$

Then, we proceed as above and using $\|x_{k+1} - x^*\| \le \sum_{j=0}^{\infty} \|\bar{p}_{k+j+1}^N\|$, and (3.53) we get

$$\|x_{k+1} - x^*\| \le \alpha_1\Gamma \left(\sum_{j=0}^{\infty} (\Gamma\ d(x_k, \mathcal{S}))^{2^{j+1}-2} \right) d(x_k, \mathcal{S})^2 \le 2\alpha_1\Gamma\|x_k - x^*\|^2.$$

This shows that $\{x_k\}$ converges q-quadratically to x^*.

The proof of the theorem for the TREBO-GN method can be easily drawn from the above considerations and using Lemmas 3.10, 3.12 and 3.15. □

3.4 Numerical experiments

The `TREBO-GN` and `TREBO-LM` methods were implemented in `Matlab` codes and tested on a number of problems arising in different areas. In this section we present two sets of experiments: the first set concerns the numerical performance analysis of the `TREBO-GN` and the `TREBO-LM` procedures, see Section 3.4.1; the second set concerns the numerical comparison of formulations (3.1) and (3.2) of problem (FP), see Section 3.4.2.

The results obtained from these experiments indicate a slight superiority of the `TREBO-GN` method with respect to the `TREBO-LM` method in terms of robustness and efficiency. Moreover, the numerical comparison of the two reformulations of problem (FP) is in favour of (3.2). These observations encouraged us in developing a new `Matlab` solver which implements the `TREBO-GN` method and the reformulation (3.2) for nonlinear feasibility problems; this will be the subject of Chapter 4.

3.4.1 Numerical comparison of the methods

The first set of experiments is devoted to a comparison of the performance of the `TREBO-GN` and `TREBO-LM` methods. The main implementation issues referring to Algorithm 3.1 are listed below.

The trust-region parameters are: $\Delta_0 = 1$, $\Delta_{min} = 10^{-12}$, $\beta_1 = 1/10$, $\beta_2 = 1/4$. In Step 1 the value of μ_k is assigned. The `TREBO-GN` method is free of such parameter; in fact, according to the rule (3.12), the method is obtained setting $\mu = 0$ and consequently $\mu_k = 0$ for all k. On the contrary, for the `TREBO-LM` method we followed the paper [48] and fixed the scalars $\mu = 1$, $\hat{\mu} = 10^{-8}$. Moreover a safeguard was introduced to prevent μ_k from being too small. Thus we set $\mu_0 = 10^{-8}\|\Theta_0\|^2$, $\mu_k = \max\{10^{-10}, \min\{\mu_{k-1}, \|\Theta_k\|^2\}\}$, $k > 0$.

In Step 2, the Jacobian matrix J is formed by using finite differences. Then the computation of the vector p_k^N in (3.7) is performed using the `Matlab` backslash operator if $m = n$ and J_k is nonsingular; otherwise it is done using the singular value decomposition of J_k. The vector p_k^N defined in (3.8) is computed by the QR decomposition applied to the least-squares problem (3.9).

Regarding Steps 6–7, if $\bar{p}_{tr}$ does not satisfy condition (3.16) we find the scalar $t \in (0, 1)$ such that $\rho_c(t\,p_k^C + (1-t)\bar{p}_{tr}) = \beta_1$ and set $p_k = t\,p_k^C + (1-t)\bar{p}_{tr}$. Then, the trust-region update is performed as follows: if the step p_k fails to satisfy (3.17) the trust-region radius is reduced setting $\Delta_k = \min\{\Delta_k/4,\ \|p_k\|/2\}$; if the step p_k satisfies (3.17) and $\rho_\theta(p_k) \geq 3/4$ we set $\Delta_{k+1} = \max\{\Delta_k,\ 2\|p_k\|,\ \Delta_{min}\}$, otherwise we let $\Delta_{k+1} = \max\{\Delta_k,\ \Delta_{min}\}$.

Successful termination of the algorithms means that they return an approximation to a zero-residual solution to the problem (BCLS). In practice we stop the algorithms when

$$\|\Theta_k\| \leq 10^{-6}.$$

A failure is declared when a stationary nonzero-residual point for the problem (BCLS) is found, i.e. $\|\Theta_k\| > 10^{-6}$ whereas

$$\|D_k \nabla \theta_k\| \leq 100\,\epsilon_m \qquad \text{or} \qquad \|\Theta_{k+1} - \Theta_k\| \leq 100\,\epsilon_m \|\Theta_k\|,$$

where $\epsilon_m \approx 2 \cdot 10^{-16}$ is the machine precision. Moreover, the algorithms fail when the trust-region radius is less than Δ_{min} or the number of iterations is greater than 300.

The two methods were compared on 30 test problems which are the constraint sets of nonlinear programming problems from the handbook of tests [24] by Floudas et al. and the Hock-Schittkowski test collection [41]. The problems have the form (FP) and are listed in Table 3.1 where we report their names, their sources and their dimensions. All the test problems contain simple bounds on the variables, except for problems marked with the symbol * where we added simple bounds on the variables letting $\Omega = \{x \in \mathbb{R}^n : x \geq 0\}$. All the problems were written as the bound-constrained least-squares problem (BCLS) where the residual function Θ has the form (3.2) and each problem was solved starting from three different starting guesses.

name, source	p	m_E	m_I	name, source	p	m_E	m_I
Test 3.4, [24]	6	0	6	Test 7.2.7, [24]	4	0	2
Test 3.5, [24]	3	0	3	Test 7.2.8, [24]	8	0	4
Test 4.10, [24]	2	0	2	Test 7.2.9, [24]	10	0	7
Test 14.1.3, [24]	2	2	0	Test 7.2.10, [24]	11	0	9
Test 14.1.5, [24]	5	5	0	HS32, [41]	3	1	1
Test 14.1.6, [24]	8	8	0	Test 3.3, [24]	5	0	6
HS8*, [41]	2	2	0	Test 7.2.1, [24]	7	0	14
HS14*, [41]	2	1	1	Test 7.2.5, [24]	5	0	6
HS15, [41]	2	0	2	HS20, [41]	2	0	3
HS55, [41]	2	6	0	HS23, [41]	2	0	5
Test 5.2.4, [24]	7	1	5	HS24, [41]	2	0	3
Test 6.3.2, [24]	8	6	0	HS44, [41]	4	0	6
Test 7.2.2, [24]	6	4	1	HS59, [41]	2	0	3
Test 7.2.3, [24]	8	0	6	HS74, [41]	4	3	2
Test 7.2.4, [24]	8	0	4	HS83, [41]	5	0	6

Table 3.1: First set of experiments, problem data.

Both methods showed to be reliable and quite insensitive to the choice of the starting point. In fact, on a total of 90 runs the methods TREBO-GN and TREBO-LM solved 80 and 77 tests respectively. The ability of the methods to handle bounds is supported by the fact that an active solution to (BCLS) was computed in 20 runs for the TREBO-GN method and in 21 runs for the TREBO-LM method. Concerning the TREBO-LM implementation, the coefficient matrix in (3.8) resulted safely numerically nonsingular for all the runs.

Most problems were solved with a low number of function evaluations and this number is, on average, favorable for the TREBO-GN method. To compare the overall computational effort of our methods we plot the performance profile proposed by Dolan and Moré [18]. We considered the 90 tests performed by each algorithm. For each test T and algorithm A, we let $\texttt{fe}_{T,A}$ denote the number of Θ-evaluations required to solve test T by the algorithm A and $\texttt{fe}_T$ be the lowest number of Θ-evaluations required by the two algorithms to solve test T. Then, for the algorithm A the performance profile is defined as

$$\pi(\tau) = \frac{\text{number of tests s.t. } \texttt{fe}_{T,A}/\texttt{fe}_T \leq \tau}{\text{number of tests}}, \quad \tau \geq 1.$$

Figure 3.1 shows the function evaluation performance profile. The difference between the performance profiles of the two methods is modest but on the whole the TREBO-GN

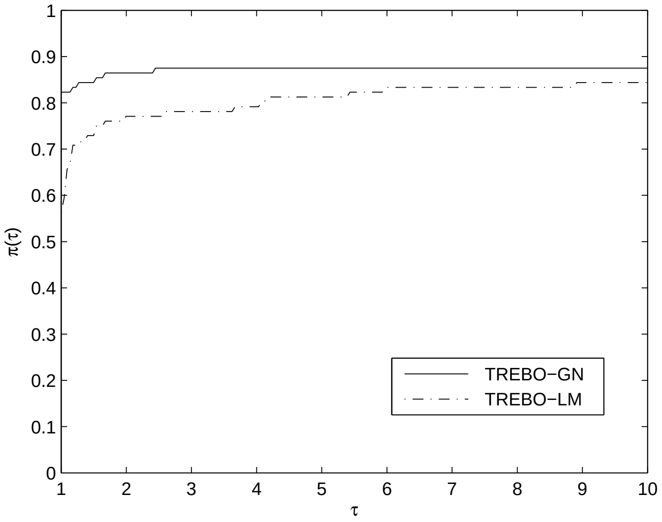

Figure 3.1: Function Θ-evaluation performance profile.

method can be considered more efficient than the TREBO-LM method. In particular, the TREBO-GN method is the most efficient algorithm for about 84% of the runs and the TREBO-LM method is within a factor 2 of the TREBO-GN method for about 80% of the runs.

Concerning the failures, it is important to note that there are occurrences where the methods converged to a first-order stationary point for (BCLS) that is not a zero-residual solution. These failures occurred 3 and 5 times for TREBO-GN and TREBO-LM method respectively and considering the overall ability of the methods to compute a first-order stationary point for (BCLS), the TREBO-GN and TREBO-LM methods solved 92% and 91% of the tests respectively.

From the above results we drew some conclusions. The methods are reliable in solving the problem (BCLS) and in most cases a zero-residual solution is computed. However our experience showed that the reliability and efficiency of the TREBO-LM method strongly depend on the choice of the sequence of scalars (3.12) and proper safeguarding. Therefore, we concluded that the most promising and reliable method is the TREBO-GN method.

3.4.2 Further experiments

The second set of experiments was conducted with the TREBO-GN method and its aim was twofold. First, we intended to study the TREBO-GN method on "hard" problems, i.e. problems with nonisolated solutions. Second, we wanted to investigate which reformulation of problem (FP) between (3.1) and (3.2) gives better numerical results.

We tested the TREBO-GN method on 67 problems: 63 problems are the feasible regions of nonlinear programming problems from the Hock-Schittkowski test collection [41]; the remaining four problems are underdetermined nonlinear systems from [14]. In Table

3.2 we report the problem names together with their dimensions and their references. The symbol * indicates the problems where the unknown was not subject to simple bounds and a nonnegative constraint was added on all the components; all the resulting problems admit a solution.

name, source	p	m_E	m_I	name, source	p	m_E	m_I
HS6*, [41]	2	1	0	Problem 4*, [14]	300	150	0
HS7*, [41]	2	1	0	Problem 10*, [41]	2	0	1
HS8*, [41]	2	2	0	Problem 11*, [41]	2	0	1
HS26*, [41]	3	1	0	HS12*, [41]	2	0	1
HS27*, [41]	2	1	0	HS13, [41]	2	0	1
HS28*, [41]	3	1	0	HS14*, [41]	2	1	1
HS39*, [41]	4	2	0	HS15, [41]	2	0	2
HS40*, [41]	4	3	0	HS16, [41]	2	0	2
HS41, [41]	4	1	0	HS17, [41]	2	0	2
HS42*, [41]	3	2	0	HS18, [41]	2	0	2
HS46*, [41]	5	2	0	HS19, [41]	2	0	2
HS47*, [41]	5	3	0	HS22*, [41]	2	0	2
HS48*, [41]	5	2	0	HS29*, [41]	3	0	1
HS49*, [41]	5	2	0	HS31, [41]	2	0	1
HS50*, [41]	5	3	0	HS32, [41]	3	1	1
HS53, [41]	5	3	0	HS33, [41]	3	0	2
HS55, [41]	6	6	0	HS34, [41]	3	0	2
HS56*, [41]	7	4	0	HS35, [41]	3	0	1
HS60, [41]	3	1	0	HS36, [41]	3	0	1
HS61*, [41]	3	2	0	HS37, [41]	3	0	2
HS62, [41]	3	1	0	HS43*, [41]	4	0	3
HS63, [41]	3	2	0	HS57, [41]	2	0	1
HS77*, [41]	5	2	0	HS64, [41]	3	0	1
HS78*, [41]	5	3	0	HS65, [41]	3	0	1
HS79*, [41]	5	3	0	HS71, [41]	4	1	1
HS80, [41]	5	3	0	HS73, [41]	4	1	2
HS87, [41]	6	4	0	HS76, [41]	4	0	3
HS99, [41]	7	2	0	HS93, [41]	6	0	2
HS107, [41]	9	6	0	HS100*, [41]	7	0	4
HS111, [41]	10	3	0	HS101, [41]	7	0	6
HS112, [41]	10	3	0	HS104, [41]	8	0	6
Problem 3*, [14]	100	100	0	HS106, [41]	10	0	8
Problem 3*, [14]	300	300	0	HS113*, [41]	10	0	8
Problem 4*, [14]	100	50	0				

Table 3.2: Second set of experiments: problem data.

The set of underdetermined systems of equalities with simple bounds consists of 35 problems. The remaining problems are made up of systems of mixed equalities and inequalities and have been solved using both reformulations (3.1) and (3.2); for sake of clarity, let R_S and R_M denote these reformulations respectively.

Three initial guesses were used for each problem. All the algorithmic options are the same of those described in Section 3.4.1 except for the setting $\Delta_{min} = \sqrt{\epsilon_m}$ and for the computation of the step p_k^N in (3.7). In fact, if $m \neq n$ or $m = n$ but J_k is singular, p_k^N is computed using the complete orthogonal factorization of J_k.

The experiments carried out on underdetermined systems of equalities showed that the TREBO-GN method is able to solve most of the problems with a low computational cost. In particular, on a total of 105 runs our method solved 87 tests and 12 failures out of 18 occurred as the sequence $\{x_k\}$ approaches a nonzero-residual stationary point for (BCLS).

Concerning the solution of problems of the form (FP), the TREBO-GN method resulted very robust on problems transformed by R_M. In fact, on a total of 96 tests, using the reformulations R_S and R_M the TREBO-GN method solved 78 and 90 tests respectively. In particular we noted that (FP) resulted quite difficult to solve for some initial guesses when it consists only of inequalities and is reformulated as R_S. In such cases we noted that the step generated by the Gauss-Newton method pointed to a stationary point for the measure of infeasibility $\|s - C_I(v)\|^2$ such that $s > 0$, i.e. to a point which is not feasible for (BCLS). This fact yielded to a failure or slow convergence; in practice if some components of the iterates become active prematurely, the length of the step taken is quite small and a stationary point for (BCLS) is slowly approached.

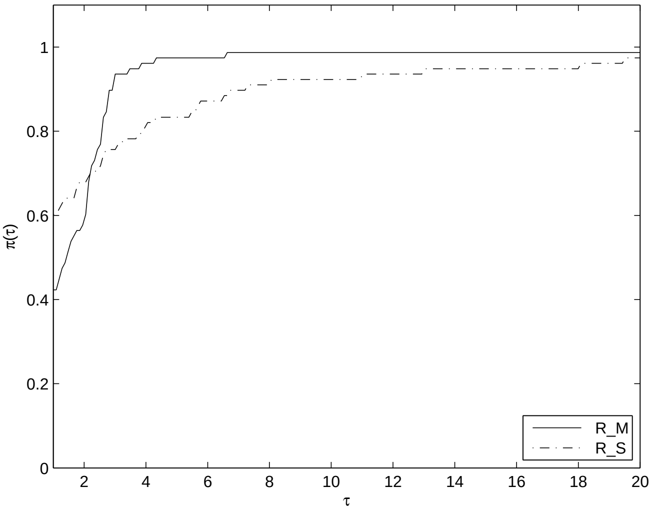

Figure 3.2: Problems (FP). Function Θ-evaluation performance profile for runs solved with both reformulations.

Finally, Figure 3.2 shows the Θ-evaluation performance profile for runs solved successfully with both reformulations. When reformulation R_S is used, the TREBO-GN method is the most efficient for 60% of the runs and that it is within a factor two and four with respect to R_M reformulation for 70% and 80% of the tests respectively. A reason why the method is more efficient when reformulation R_S is used, is that fast convergence is achieved in several runs. At this regard, it is important to make some comments on the Jacobian matrix J of Θ. Letting x^* be a solution to (FP), in case of reformulation R_M, we have $\max\{C_I(x^*), \mathbf{0}\} = \mathbf{0}$ where $\mathbf{0}$ is the null vector of dimension m_I. Hence, by (3.4) matrix $J(x^*)$ is not full rank and we cannot expect quadratic convergence. On the other hand, the full rank condition of $J(x^*)$ is not precluded if the reformulation R_S is used.

Chapter 4

A new `Matlab` solver: `TRESNEI`

We introduce a `Matlab` implementation of the `TREBO-GN` method presented in Chapter 3. The solver is adequate for solving zero and small-residual bound-constrained nonlinear least-squares problems and handles the solution of nonlinear feasibility problems. For this reason it is called `TRESNEI`, Trust-REgion Solver for Nonlinear Equalities and Inequalities.

Our solver addresses the solution of nonlinear feasibility problems offering their internal reformulation (3.2) and solving the resulting problem (BCLS) by the `TREBO-GN` algorithm. For sake of generality, the solver is designed using a widespread modelling of the problems. Moreover, `TRESNEI` can be applied in the solution of nonsquare bound-constrained nonlinear systems and it turns out to be a nontrivial extension of the solver `STRSCNE` [3] for square bound-constrained systems.

The functions for solving bound-constrained nonlinear least-squares problems provided by the `Matlab` Optimization Toolbox [54] cannot solve underdetermined problems. On the other hand, `TRESNEI` overcomes this limitation and can be applied in the solution of problem (BCLS) irrespective of its dimensions.

`TRESNEI` has been intensively tested and the goals of our experiments were twofold. First, we were interested in assessing if the formulation (BCLS) may offer an advantage as compared with an unconstrained least-squares formulation. Second, we were interested in comparing the computational cost and robustness of our algorithm with competing solvers. The function `lsqnonlin` from the `Matlab` Optimization Toolbox served our purposes as will be shown in Sections 4.3 and 4.5.

The overall performance of `TRESNEI` show that it is cost effective and robust. In this chapter we present the structure and the usage of our solver and the results of a benchmarking process for `TRESNEI` and the `lsqnonlin` function from the `Matlab` Optimization Toolbox.

4.1 Problem statement

`TRESNEI` implements the `TREBO-GN` method for solving the bound-constrained least-squares problem

$$\min_{x \in \Omega} \theta(x) = \frac{1}{2}\|\Theta(x)\|^2, \tag{BCLS}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ and Ω is the n-dimensional box $\Omega = \{x \in \mathbb{R}^n \mid l \leq x \leq u\}$, $l \in (\mathbb{R} \cup -\infty)^n$, $u \in (\mathbb{R} \cup \infty)^n$, $l < u$.

The problems handled by the solver have a general formulation. They are the bound-constrained least-squares problem stated as

$$\min_{L \leq x \leq U} \frac{1}{2}\|C_E(x)\|^2, \tag{4.1}$$

where $C_E : \mathbb{R}^n \to \mathbb{R}^{m_E}$, and the nonlinear feasibility problem given by

$$\begin{array}{l} C_E(x) = 0, \\ C_I(x) \leq 0, \\ L \leq x \leq U, \end{array} \tag{4.2}$$

where $C_E : \mathbb{R}^n \to \mathbb{R}^{m_E}$, $C_I : \mathbb{R}^n \to \mathbb{R}^{m_I}$. Following a widespread modelling of the above problems, in both (4.1) and (4.2), the components of the box constraints $L, U \in \mathbb{R}^n$ satisfy $-\infty \leq L_i \leq U_i \leq \infty$, $i = 1, \cdots, n$. It follows that the components x_i of x can be either free or bounded on one side or bounded from above and from below, or "fixed" i.e. variables with equal upper and lower bounds. In fact, it is common to provide the problem parameters as "fixed" variables, see e.g. the standard adopted in CUTEr [33]. In what follows, we let I_{fx}, I_{lb} and I_{ub} be the sets containing the indices of fixed, lower and upper bounded variables respectively:

$$I_{fx} = \{i \in \{1, \ldots, n\} : L_i = U_i\}, \tag{4.3}$$

$$I_{lb} = \{i \in \{1, \ldots, n\} : i \notin I_{fx} \text{ and } L_i \neq -\infty\}, \tag{4.4}$$

$$I_{ub} = \{i \in \{1, \ldots, n\} : i \notin I_{fx} \text{ and } U_i \neq +\infty\}. \tag{4.5}$$

Obviously I_{lb} and I_{ub} may not be disjoint.

The first step of TRESNEI is to express the problems considered as a bound-constrained least-squares problem where Θ is continuously differentiable and $l < u$, as required by the TREBO-GN method. Suppose that the problem (4.1) contains no fixed variables. Then, TRESNEI attempts to solve (BCLS) where

$$\begin{array}{l} \Theta(x) = C_E(x), \\ m = m_E, \quad l = L, \quad u = U. \end{array} \tag{4.6}$$

This is the case also if (4.2) contains no fixed variables neither nonlinear inequalities. If problem (4.1) contains fixed variables, the bounds on such variables are dropped introducing equalities of the form

$$[x]_{I_{fx}} - [U]_{I_{fx}} = 0. \tag{4.7}$$

Thus, problem (4.1) takes the form (BCLS) where

$$\Theta(x) = \begin{pmatrix} C_E(x) \\ [x]_{I_{fx}} - [U]_{I_{fx}} \end{pmatrix}, \tag{4.8}$$

$\Theta : \mathbb{R}^n \to \mathbb{R}^{m_E + n_{fx}}$ and n_{fx} is the cardinality of the set I_{fx}. The bounds l and u are given by

$$l_i = \left\{ \begin{array}{ll} -\infty & \text{if } i \in I_{fx} \\ L_i & \text{otherwise} \end{array} \right., \qquad u_i = \left\{ \begin{array}{ll} +\infty & \text{if } i \in I_{fx} \\ U_i & \text{otherwise} \end{array} \right., \tag{4.9}$$

$i = 1, \dots, n$. Analogously, the problem (4.2) is posed as a bound-constrained nonlinear least-squares problem including fixed variables and the general inequalities $C_I(x) \leq 0$ into the function Θ. In particular, the general inequalities are converted into equalities using the continuously differentiable function $[t]_+ = \max\{t,\, 0\}^2/2$ and the function Θ in (BCLS) takes the form

$$\Theta(x) = \begin{pmatrix} C_E(x) \\ [x]_{I_{fx}} - [U]_{I_{fx}} \\ [C_I(x)]_+ \end{pmatrix}. \tag{4.10}$$

The number of components of Θ is $m = m_E + m_I + n_{fx}$, where n_{fx} is the cardinality of the set I_{fx}. The remaining simple bounds are kept separate from the objective function and the bounds l and u are as in (4.9).

4.2 The algorithm

The procedure implemented in `TRESNEI` consists of two phases: in the first phase, the problem (BCLS) is formed; in the second phase such problem is solved. The implementation is fully described in Algorithm 4.1 which is an adaptation to our framework of Algorithm 3.1 for the `TREBO-GN` method. The notations used are in accordance with those used in Section 4.1. The subsections that follow provide a detailed description of the steps of Algorithm 4.1.

4.2.1 Problem description

The problem to be solved by the `TREBO-GN` algorithm is formed in Step 1. The statement of Algorithm 4.1 refers to the general form (4.10) of Θ which includes all the problems considered.

`TRESNEI` covers the solution of (4.1) providing the residual function C_E and the bounds L and U. Then either the reformulation (4.6) or (4.8) is internally carried out. Clearly, these functions Θ can be viewed as a special case of (4.10) where the vector function C_I is empty.

Regarding (4.2), `TRESNEI` offers its user the facility of requiring a minimal description and building an internal reformulation of the problem. In particular, the functions C_E, C_I and the bounds L and U are expected. Then, the problem (BCLS) is internally formed as described in the previous paragraph. It is important to note that the user may prefer an alternative transformation of the nonlinear inequalities to the one employed in (4.10). For example, the use of slack variables casting nonlinear inequalities into nonlinear equalities in (3.1), can be accomplished providing the resulting system of nonlinear equations to `TRESNEI`.

If the Jacobian matrices C_E', C_I' have been provided along with C_E and C_I, the Jacobian matrix J is formed. Otherwise, a finite difference approximation of matrix J is evaluated.

Finally, the user must supply an initial guess x_0 belonging to the box Ω.

Algorithm 4.1 TRESNEI

```
Input: $C_E$, $C_I$, $L$, $U$, $m_E$, $m_I$, $n$,
       $x_0$, $\Delta_0$, $k_M$, $\epsilon_1$, $\epsilon_2$.
1. Problem description
    Let $l$, $u$ as in (4.9), $\Omega = \{x \in \mathbb{R}^n \mid l \le x \le u\}$; if $x_0 \notin \Omega$, exit.
    Let $m$ be the length of $\Theta$ in (4.10).
    Compute $\Theta(x_0)$ by (4.10) and $J(x_0)$. Set $k = 1$.
2. Internal parameters
    Set $\beta_1 = 1/10$, $\beta_2 = 1/4$, $\beta_3 = 3/4$.
    Let $\epsilon_m$ be the machine precision, set $\Delta_{m1} = \sqrt{\epsilon_m}$, $\Delta_{m2} = \epsilon_m$.

    While $k \le k_M$ do
    3. Solve the trust-region problem
        3.1 If $m \ne n$, compute $p_k^N$ given in (3.7) by the complete
                orthogonal decomposition.
            If $m = n$, solve the linear system $J_k\, p_k^N = -\Theta_k$;
                If $J_k$ is singular, compute $p_k^N$ by the complete
                orthogonal factorization.
        3.2 If $\|p_k^N\| \le \Delta_k$, set $p_{tr} = p_k^N$;
            Else form $p_k^c$ given in (4.11),
                compute the dogleg step $p_{tr}$ between $p_k^N$ and $p_k^c$.
    4. Compute the trial step $p_k$
        4.1 Let $\bar{p}_{tr} = P_\Omega(x_k + p_{tr}) - x_k$.
        4.2 Compute $p_k^C$ in (3.13).
        4.3 If $\bar{p}_{tr}$ satisfies (3.16), set $p_k = \bar{p}_{tr}$;
             Else compute the positive root $t^*$ of $\rho_c(t\, p_k^C + (1-t)\bar{p}_{tr}) - \beta_1 = 0$,
                set $p_k = t^*\, p_k^C + (1-t^*)\bar{p}_{tr}$.
    5. Test on $p_k$ and trust-region radius update
        5.1 Compute $\Theta(x_k + p_k)$ by (4.10).
             If $p_k$ satisfies (3.17), set $x_{k+1} = x_k + p_k$;
            Else set $\Delta_k = \min\{\Delta_k/4,\ \|p_k\|/2\}$;
                If $\Delta_k > \Delta_{m2}$, go to Step 3.2;
                Else exit.
        5.2 If $\rho_\theta(p_k) \ge \beta_3$, set $\Delta_{k+1} = \max\{\Delta_k,\ \Delta_{m1},\ 2\|p_k\|\}$;
             Else set $\Delta_{k+1} = \max\{\Delta_k,\ \Delta_{m1}\}$.
    6. Termination test
        6.1 If (4.13) or (4.14) is satisfied, exit.
             Else compute $J(x_{k+1})$ and increment $k$.
```

4.2.2 Solution of the trust-region problem

The solution of the trust-region problem (3.6) is addressed in Step 3 of the algorithm. If the Jacobian matrix J is square, the computation of the step p_k^N given in (3.7) is attempted by the `Matlab` backslash operator. If J is square and results close to singular or J is nonsquare, the complete orthogonal decomposition of J is applied using procedures which are slight modifications of those given by Higham in [40]. Clearly, the use of matrix factorization sets limits on the size of problems that can be solved efficiently by `TRESNEI`.

If p_k^N does not solve the trust-region problem, then we use the classical dogleg path to approximate the trust-region solution, see Section 2.2.2. The Cauchy step p_k^c for the problem (3.6) has the form

$$p_k^c = -\min\left\{\frac{\|\nabla\theta_k\|^2}{\|J_k\nabla\theta_k\|^2}, \frac{\Delta_k}{\|\nabla\theta_k\|}\right\}\nabla\theta_k. \tag{4.11}$$

The procedure described above was implemented achieving economies in the calculations. Since a zero component of $[C_I(x_k)]_+$ gives rise to a zero component in Θ and to a null row in J, instead of (3.5) we use the reduced model

$$\hat{m}_k(p) = \frac{1}{2}\|\hat{J}_k\, p + \hat{\Theta}_k\|^2, \tag{4.12}$$

where $\hat{\Theta}_k$ is the vector formed by $C_E(x_k)$ and the nonzero components of $[C_I(x_k)]_+$ and $\hat{J}_k$ is the Jacobian of $\hat{\Theta}$ at x_k, see [34].

4.2.3 Computation of the trial step

In Step 4 of the algorithm, the trial step p_k is formed testing the condition (3.16); the scalar β_1 used in (3.16) is an internal parameter fixed in Step 2. The Generalized Cauchy step p_k^C is evaluated in Step 4.2 using (3.13), (3.14) and (3.15). The step $\bar{p}_{tr}$ is accepted as the trial step p_k if it satisfies (3.16). Alternatively, we find the step p_k of the form $t\,p_k^C + (1-t)\bar{p}_{tr}, t \in [0,1]$ such that $\rho_c(p_k) = \beta_1$; this is equivalent to solve a quadratic scalar equation admitting a unique positive root t^*.

4.2.4 Test on the trial step and trust-region radius update

The trial step p_k is accepted in Step 5 of the algorithm if condition (3.16) is satisfied. In this case the trust-region radius for the next iterate is updated following a standard strategy and imposing $\Delta_{k+1} \geq \Delta_{m1}$. Clearly, on termination of each iteration, the trust-region radius may be smaller that Δ_{m1}. On the other hand, if p_k fails to satisfy (3.17), then it is rejected and the trust-region radius Δ_k is reduced. Note that if Δ_k becomes smaller than the fixed parameter Δ_{m2}, we terminate the procedure and declare a failure. The parameters β_2, β_3, Δ_{m1}, Δ_{m2} are set internally in Step 2.

4.2.5 Termination criteria and accuracy

Successful termination of `TRESNEI` means that one of the following conditions is met

$$\|\Theta_k\|_\infty \leq \epsilon_1, \tag{4.13}$$

$$\min\{\|D_k\nabla\theta_k\|, \|P(x_k - \nabla\theta_k) - x_k\|\} \leq \epsilon_2\sqrt{n}, \tag{4.14}$$

where ϵ_1 and ϵ_2 are prescribed tolerances. The condition (4.14) involves two optimality measures: the scaled gradient $D\nabla f$ which is a key ingredient of our method, and the projected gradient of function f. The use of both measures is due to the fact that the value $\|D\nabla f\|$ may oscillate and exhibit a large growth at some iterations. Thus, the use of the norm of the projected gradient provides a more reliable stopping condition.

4.3 The function `lsqnonlin`

The use of the commercial `Matlab` Optimization Toolbox software for solving least-squares problems gives rise to an approach alternative to the one used in `TRESNEI`. In particular, it yields to testing an unconstrained least-squares formulation of (4.2).

The `MATLAB` Optimization Toolbox includes the function `lsqnonlin` which consists of two implementations: the large-scale algorithm and the medium-scale algorithm. The large-scale algorithm is a subspace trust-region method while the medium-scale algorithm uses either the Levenberg-Marquardt method or the Gauss-Newton method globalized by a line search strategy.

The applicability of `lsqnonlin` has some limitations. Bounds on the variables can be handled only by the large-scale algorithm. On the other hand, such algorithm cannot solve problems where the number of elements of Θ is lower than the number of variables. Therefore, bound-constrained underdetermined least-squares problems cannot be solved by `lsqnonlin`.

Because of the above limitations, the only way to solve a variety of systems of equalities and inequalities without restrictions on their dimensions consists in expressing the problems as unconstrained least-squares problems. In fact, given (4.2) we apply the medium-scale algorithm to the problem

$$\min_{x\in\mathbb{R}^n} g(x) = \|G(x)\|^2, \tag{4.15}$$

where

$$G(x) = \begin{pmatrix} C_E(x) \\ [x]_{I_{fx}} - [U]_{I_{fx}} \\ [C_I(x)]_+ \\ \max\left\{ \Big[[L-x]_{I_{lb}} \Big]_+, \Big[[x-U]_{I_{ub}} \Big]_+ \right\} \end{pmatrix}. \tag{4.16}$$

Note that the function G differs from (4.10) as it incorporates the simple bounds in the sets I_{lb}, I_{ub} and that G is continuously differentiable.

We will consider the solution of (4.15) by the medium-scale algorithm of `lsqnonlin`. It terminates successfully either if the directional derivative along the search direction s_k and the ∞-norm of the gradient of g_k are less than prescribed tolerances, i.e.

$$\nabla g_k^T s_k \le \zeta_1 \quad \text{and} \quad \|\nabla g_k\|_\infty \le 10(\zeta_1+\zeta_2), \tag{4.17}$$

or if the magnitude of search direction is sufficiently small, i.e.

$$\|s_k\|_\infty \le \zeta_2. \tag{4.18}$$

On the other side, a failure is declared if the line search strategy can not sufficiently decrease the residual along the current search direction.

4.4 Benchmarking

The solvers `TRESNEI` and `lsqnonlin` do not test a uniform stopping criterium. Hence, it is essential benchmarking the two solvers in order to guarantee that the returned approximate solutions satisfy the same accuracy requirement, see e.g. [19, 36].

We adopt the benchmarking process proposed in [19] for general constrained optimization problems and we fit it to problem (BCLS). It consists in computing and checking a specific test for the solvers a posteriori. Specifically, each solver is run using the default tolerances. If the approximate solution returned by the solver does not satisfy the a posteriori convergence test, then the native solver tolerances are reduced and the problem is solved again. Further tolerance reductions are made until the a posteriori convergence test is satisfied or a failure is declared.

The definition of the a posteriori convergence test is given in terms of measures for feasibility and stationarity. Such measures are defined using an error measure function $\delta[\cdot,\cdot]$ which involves a mixture of absolute and relative error. In particular, given real numbers ξ_1 and ξ_2, $\delta[\xi_1,\xi_2]$ is defined as

$$\delta[\xi_1,\xi_2] = \min\left\{|\xi_1-\xi_2|, \frac{|\xi_1-\xi_2|}{|\xi_1|+|\xi_2|}\right\},$$

with $\delta[0,0]=0$ and $\delta[\xi_1,\xi_2]=1$ if either ξ_1 or ξ_2 is infinite. The function $\delta[\cdot,\cdot]$ is continuous.

The feasibility measure is given by

$$\nu_f(x) = \|v(x)\|_\infty, \tag{4.19}$$

where $v(x)\in \mathbb{R}^n$ and

$$(v(x))_i = \begin{cases} 0 & \text{if } l_i \le x_i \le u_i, \\ \min\{\delta[x_i,l_i],\delta[x_i,u_i]\} & \text{otherwise.}\end{cases}$$

Clearly, ν_f is null at feasible points while it measures the constraint violations at infeasible points. Given a small positive scalar τ, a vector x is defined to be *τ-feasible* if $0<\nu_f(x)\le\tau$, [19].

The stationarity measure ν_s can be defined as

$$\nu_s(x,\tau) = \|r(x,\tau)\|_\infty, \tag{4.20}$$

where τ is a small positive scalar, $r(x,\tau)\in\mathbb{R}^n$ and

$$(r(x,\tau))_i = \begin{cases} \min\{0,(\nabla\theta(x))_i\} & \text{if } \delta[x_i,l_i]\le\tau, \delta[x_i,u_i]>\tau, \\ \max\{0,(\nabla\theta(x))_i\} & \text{if } \delta[x_i,l_i]>\tau, \delta[x_i,u_i]\le\tau, \\ (\nabla\theta(x))_i & \text{if } \delta[x_i,l_i]>\tau, \delta[x_i,u_i]>\tau, \\ 0 & \text{if } \delta[x_i,l_i]\le\tau, \delta[x_i,u_i]\le\tau \text{ or} \\ & \text{if } \delta[x_i,U_i]\le\tau, i\in I_{fx}. \end{cases} \tag{4.21}$$

Note that the last assignment in (4.21) is related to the equation (4.7) and that U_i is the value of the fixed variable $x_i, i\in I_{fx}$, in problem (4.2).

The relationship between the optimality measures (4.19) and (4.20) and the first-order optimality conditions for the problem (BCLS) is clarified by the following theorem.

Theorem 4.1 *Let $\tau > 0$ be given and let x^* be a first-order stationary point for the problem (BCLS). If $\{x_k\}$ is a sequence that converges to x^*, then $\{\nu_f(x_k)\}$ and $\{\nu_s(x_k, \tau)\}$ converge to zero.*

Proof. The sequence $\{\nu_f(x_k)\}$ trivially converges to zero since the sequence $\{x_k\}$ converges to a feasible point x^* of problem (BCLS).

Now we prove that $\{\nu_s(x_k, \tau)\}$ converges to zero. Consider the i-th component of x_k for k sufficiently large and without lack of generality suppose that if x_i^* is active then $x_i^* = l_i$. Since $\{x_k\}$ converges to x^*, if $x_i^* = l_i$ then $\delta[(x_k)_i, l_i] \le \tau$ for all k sufficiently large, otherwise either $\delta[(x_k)_i, l_i] \le \tau$ or $\delta[(x_k)_i, l_i] > \tau$ may hold.

If $x_i^* = l_i$ and $(\nabla\theta(x^*))_i > 0$, then by (4.21) and the continuity of the gradient, we get

$$(r(x_k, \tau))_i = \min\{0, (\nabla\theta(x_k))_i\} = 0,$$

for k sufficiently large.

If $x_i^* = l_i$ and $(\nabla\theta(x^*))_i = 0$ or $x_i^* > l_i$ and $\delta[(x_k)_i, l_i] \le \tau$, then

$$(r(x_k, \tau))_i = \min\{0, (\nabla\theta(x_k))_i\} \le |(\nabla\theta(x_k))_i|. \tag{4.22}$$

Since $\lim_{k\to\infty}(\nabla\theta(x_k))_i = 0$, then from (4.22) we obtain $\lim_{k\to\infty}(r(x_k, \tau))_i = 0$.

Finally if $x_i^* > l_i$ and $\delta[(x_k)_i, l_i] > \tau$, $\lim_{k\to\infty}(r(x_k, \tau))_i = 0$ easily follows from (4.21) and (2.23).

The case $x_i^* = u_i$ can be studied as above and therefore we can conclude that $\lim_{k\to\infty} \nu_s(x_k, \tau) = 0$. □

The first requirement on the solutions returned by the solvers TRESNEI and lsqnonlin is their τ-feasibility. Moreover, we assess the accuracy of the solutions by using the stationarity measure ν_s. In practice, benchmarking requires that the solutions delivered by TRESNEI and lsqnonlin satisfy

$$\nu_f(x) \le \tau_f, \quad \nu_s(x, \tau_f) \le \tau_s, \tag{4.23}$$

for specified tolerances τ_f and τ_s. The τ-feasibility feature is nontrivially fulfilled by lsqnonlin as it may return a nonzero-residual stationary point for (4.15). On the other hand, since TRESNEI generates a sequence of feasible iterations, enforcing (4.23) means controlling only the stationarity measure ν_s.

4.5 Numerical experience

In this section we discuss the numerical experiments with TRESNEI and lsqnonlin, with particular emphasis on the effects of the enforcement of the convergence test (4.23). All the tests were performed on an Intel Xeon (TM) 3.4 Ghz, 1GB RAM using Matlab 7.6 and machine precision $\epsilon_m \approx 2 \cdot 10^{-16}$.

4.5.1 The problem set

The test examples are from the CUTEr test collection [33]. In view of the suitability of `TRESNEI` for medium-size problems, we selected 135 problems of the form (4.2) and we adjusted their dimensions to obtain variants where $n \leq 500$. Among the problems considered, there are 14 systems of nonlinear equations; the rest of the problems are constraint sets of programming problems.

In Tables 4.2-4.4 we report the names along with the main characteristics of the problems under consideration. In particular, n_{fr} and n_{fx} indicate the number of free and fixed variables respectively, n_b the number of variables that are bounded at least on one side and n_r the number of variables bounded on both sides ("range" variables). Moreover, the number m_E of equalities and the number m_I of general inequalities are reported.

The starting point x_0 and the analytical Jacobian matrices C_E' and C_I' are provided by CUTEr as part of each problem specification.

4.5.2 Algorithmic options

`TRESNEI` was run with the initial trust-region radius $\Delta_0 = 1$. The initial guess x_0 was the one provided by CUTEr if it is feasible. Otherwise such vector was projected onto the box Ω; this case occurred for 17 problems.

In `lsqnonlin`, for the line search algorithm we selected a safeguarded cubic polynomial method instead of the default strategy. This choice is recommended in [54] if gradients are supplied and can be calculated quite inexpensively. The initial point used is the one employed in `TRESNEI`.

For both solvers, all attempts to solve the test problems were limited to a maximum of 1000 iterations or 1000 function evaluations. The default tolerances ϵ_1, ϵ_2 in (4.13) and (4.14) and ζ_1, ζ_2 in (4.17) and (4.18) are

$$\epsilon_1 = \epsilon_2 = 10^{-6}, \qquad \zeta_1 = \zeta_2 = 10^{-6}.$$

Benchmarking of the solvers has been performed as follows. If one solver reports a failure with the default tolerances then the benchmarking process is not activated. Otherwise, the a posteriori test (4.23) is checked as suggested in [19], i.e. setting

$$\tau_f = \tau_s = 10^{-6}.$$

In case (4.23) is not satisfied, the tolerances provided to the solvers are reduced by a factor 10 and the problem is solved again. The progressive reduction of the tolerances is stopped, and a failure is declared, when they reach the value 10^{-16}. It is important to remark that if the solvers fail during the repeated runs but the test (4.23) is satisfied at the returned approximation, then we declare a successful run.

4.5.3 Results

Firstly, we tested `TRESNEI` and `lsqnonlin` on the problem set using the default tolerances. Secondly, we compared `TRESNEI` and `lsqnonlin` checking the a posteriori test (4.23).

Let consider the experiments conducted with the default tolerances. Both the Gauss-Newton method and the Levenberg-Marquardt method implemented in the medium-scale algorithm of `lsqnonlin` were run, see §4.3. `TRESNEI` solved 121 of the 135 problems, the Levenberg-Marquardt and the Gauss-Newton implementations of `lsqnonlin` solved 131 and 79 problems respectively. In fact, the Gauss-Newton implementation of `lsqnonlin` fails to handle overdetermined problems with rank-deficient Jacobian matrices. Due to this pitfall, we will refer to the Levenberg-Marquardt implementation of `lsqnonlin` in the remaining of the section.

For the successful runs, we analyzed the value of residual functions Θ in (4.10) and G in (4.16) returned by `TRESNEI` and `lsqnonlin` respectively; Figure 4.1 shows the values

$$\rho_T = -\log_{10} \|\Theta(x)\|, \qquad \rho_l = -\log_{10} \|G(x)\|. \tag{4.24}$$

We note that the final residual is less than 10^{-6} in 70 problems for `TRESNEI` and in 26 problems for `lsqnonlin`. For problem `HS99EXP`, the residual $\|G\|$ returned is around 1.

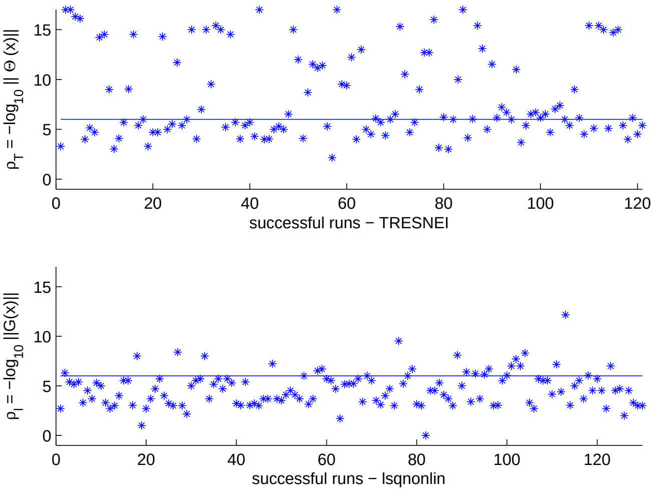

Figure 4.1: Plot of final residuals (4.24) for the successful runs.

A further information of interest is the value of the final constraint violations $\|[C_I]_+\|_\infty$ for the 53 problems containing inequalities. For all the successful runs this norm is lower than 1. In Table 4.1 we list five ranges of values for $\|[C_I]_+\|_\infty$ at the computed solution and report the number of problems which attained a value in such intervals. We remark that the inequalities $C_I(x) \leq 0$ are satisfied for 9 problems solved by `TRESNEI` and for 7 problems solved by `lsqnonlin`. In the remaining problems, the constraint violations appear to be smaller at the solutions computed by `TRESNEI`.

Since the convergence tests of the solvers are not consistent, conclusions are difficult to be drawn from the results obtained. It is quite evident that `TRESNEI` returns small values of $\|\Theta\|$; hence we can safely conclude that the solutions returned by the solver are accurate approximations to the solutions of problem (4.2). On the other hand, assessing the accuracy of the solutions delivered by `lsqnonlin` is more difficult. The residual

	`TRESNEI`	`lsqnonlin`
$\|[C_I]_+\|_\infty$	`no.of problems`	`no.of problems`
0	9	7
$(0\ 10^{-5})$	4	0
$[10^{-5}\ 10^{-2})$	18	10
$[10^{-2}\ 10^{-1})$	14	29
$[10^{-1}\ 1)$	0	3

Table 4.1: Constraint violations at computed solutions.

function G in (4.15) includes the simple bounds and the values of $\|G\|$ shown in Figure 4.1 may indicate the computation of an infeasible solution of (4.2) with respect to the simple bounds.

Performing the benchmark, `TRESNEI` and `lsqnonlin` computed a solution satisfying (4.23) in 119 (88%) of the 135 problems and 117 (87%) of the tests, respectively. Figure 4.2 displays the function-evaluation count performance profile [18] for these runs. The plot shows that both solvers are very reliable and makes clear that `TRESNEI` is the most efficient for about 75% of the runs and `lsqnonlin` is within a factor 5 of `TRESNEI` for about 80% of the runs.

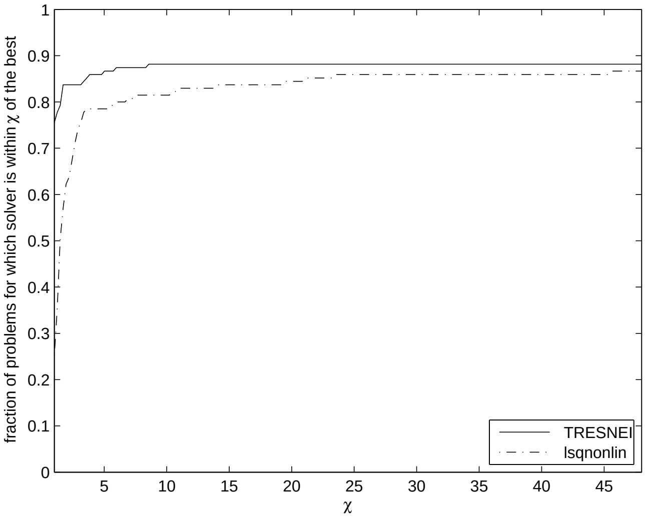

Figure 4.2: Performance profile, $\psi(\chi)$: function-evaluation counts for the 135 CUTEr problems under consideration.

Some comments on the failures occurring in the benchmarking process are needed. `TRESNEI` fails to satisfy the stationarity requirement ν_s given in (4.23) for 2 problems while `lsqnonlin` fails 13 times as the τ-feasibility required in (4.23) is not met. The reason why `lsqnonlin` fails to satisfy this requirement is that its iterates may not be feasible and typically the problem (4.2) has nonisolated solutions. Thus, it may happen that the sequence generated by `lsqnonlin` converges to a solution to problem (4.15) that is not feasible but close to the boundary of the box Ω. This situation can be verified numerically as the value of ν_f settles down for decreasing values of the tolerances ζ_1 and ζ_2 in (4.17) and (4.18).

The purpose of what follows is to investigate the effect of the convergence criteria (4.23) on solvers performance. In Figures 4.3-4.4, for each problem we report the bar showing the values

$$\Pi = -\log_{10}(\tau_B), \tag{4.25}$$

where τ_B is the tolerance needed by either TRESNEI or lsqnonlin to satisfy the condition (4.23). If one solver failed either with the default tolerances or in the benchmarking, no bar is plotted. Concerning runs made with TRESNEI, Figure 4.3 shows that the height of 65 bars is equal to 6 and that for only 7 problems the bars reach values greater than or equal to 9. On the other hand, for lsqnonlin 26 bars have height equal to 6 and 24 bars are higher than 9; this indicates that lower tolerances than the default ones are often necessary to obtain accurate solutions in the sense of (4.23), see Figure 4.4.

These observations are confirmed in Figures 4.5-4.6 where for each problem and solver we plot the performance metric

$$p(x) = -\log_{10}(\max\{\nu_f(x), \nu_s(x, \tau_f)\}), \tag{4.26}$$

for the computed solution x. Clearly, the heights of the bars give the levels of accuracy reached, and returned solutions x such that $p(x) < 6$ do not satisfy the test (4.23). The white bars indicate the values $p(x)$ obtained using the default tolerances. The black bars indicate the values $p(x)$ resulting from the benchmarking process; if a black bar is not present, then (4.23) is satisfied using the default tolerances. If no bar is present, then the solver fails with the default tolerances.

Figure 4.5 shows that TRESNEI is able to compute highly accurate solutions and, in accordance to Figure 4.3, the convergence test (4.23) is satisfied with the default tolerances for most of the problems. Therefore, we can conclude that criteria (4.13)-(4.14) tend to agree with (4.23) in most cases. On the other hand, comparing Figure 4.5 and Figure 4.6 it is evident that the level of accuracy of the solutions computed by lsqnonlin is remarkably lower than in TRESNEI.

4.6 Final remarks

The solver TRESNEI is accessible through the web site: http://TRESNEI.de.unifi.it. Since TRESNEI does not require any special toolbox, it can easily serve as a template for translations in another language.

TRESNEI has been used in several contexts. We are aware of its use in [67] for solving nonlinear feasibility problems arising in the restoration phase of a filter method for nonlinear optimization with expensive function. Moreover in [5], TRESNEI has been applied to bound-constrained systems resulting from the performance analysis of multi-radio wireless networks.

The overall performance of TRESNEI against lsqnonlin encourages us to study possible improvements and extensions of the algorithm implemented. A chance of extending the applicability of TRESNEI comes from the use of iterative linear solvers in the trust-region solution. Next chapter is devoted to the design and analysis of the TREBO-GN method for large-scale problems.

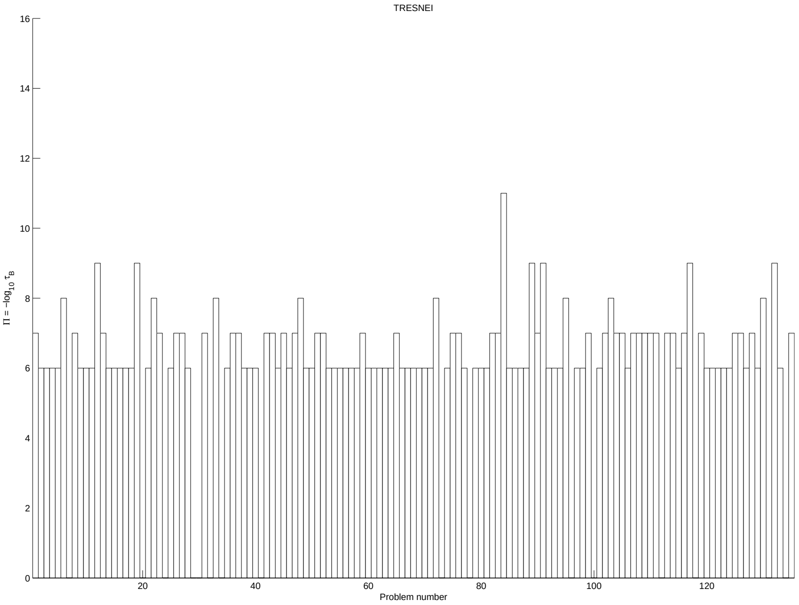

Figure 4.3: Graph of the performance measure (4.25) for TRESNEI.

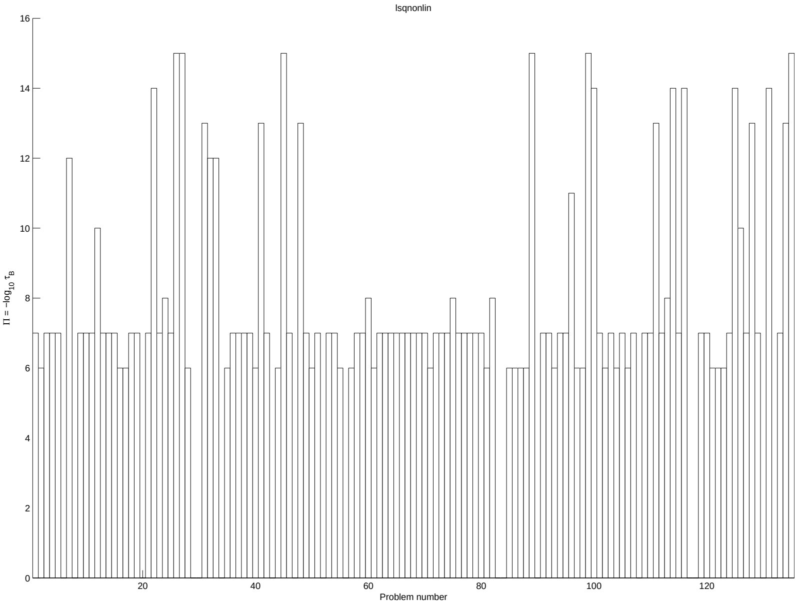

Figure 4.4: Graph of the performance measure (4.25) for lsqnonlin.

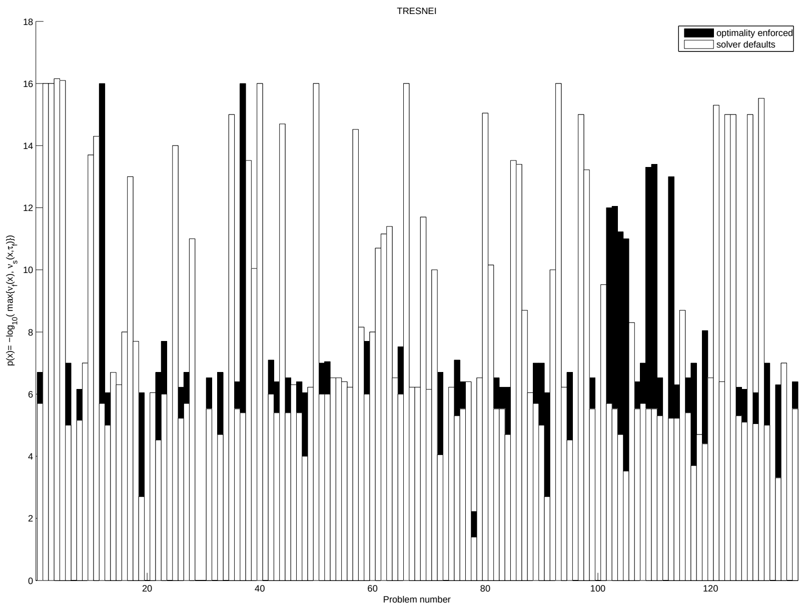

Figure 4.5: Graph of the performance metric (4.26) for `TRESNEI`.

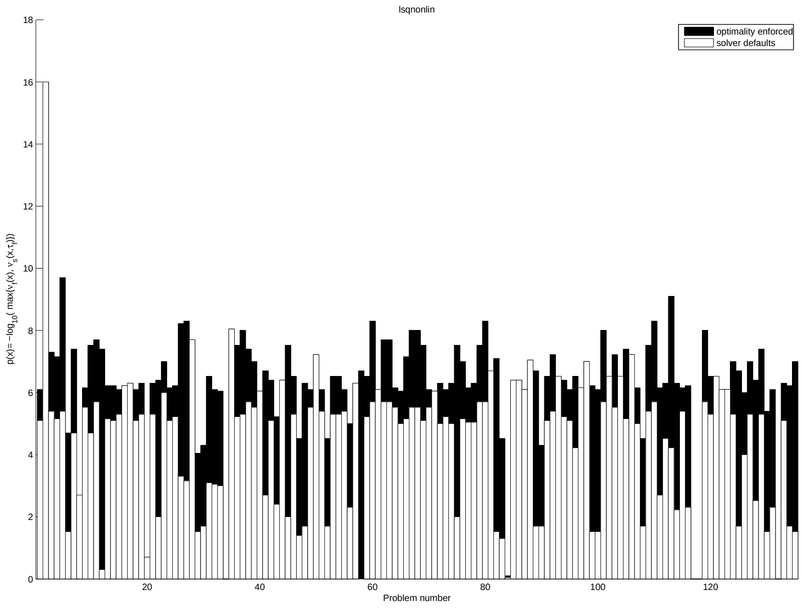

Figure 4.6: Graph of the performance metric (4.26) for `lsqnonlin`.

Problem	n_{fr}	n_b	n_r	n_{fx}	m_E	m_I
AIRPORT	0	0	84	0	0	42
ALLINITC	1	1	1	1	1	0
ALJAZZAF	0	100	0	400	1	0
ALSOTAME	0	0	2	0	1	0
ANTWERP	0	3	24	0	8	2
AVGASA	0	0	8	0	0	10
AVION2	0	0	49	0	15	0
BATCH	0	2	46	0	12	61
BIGGSC4	0	0	4	0	0	13
BLOCKQP1	0	0	30	0	10	1
BLOWEYA	11	0	11	0	12	0
BT13	4	1	0	0	1	0
CAMSHAPE	0	0	100	0	0	304
CANTILVR	0	5	0	0	0	1
CHANDHEQ	0	10	0	0	10	0
CHEMRCTA	0	10	0	0	10	0
CHEMRCTB	0	10	0	0	10	0
CLNLBEAM	51	0	98	4	100	0
CONCON	10	5	0	0	11	0
CORE2	0	41	116	0	108	26
CORKSCRW	47	0	40	9	60	10
C-RELOAD	0	84	258	0	200	84
CSFI1	0	4	1	0	2	3
CSFI2	0	5	0	0	2	3
CVXQP1	0	0	100	0	50	0
DALLASM	0	0	196	0	151	0
DALLASS	0	0	46	0	31	0
DECONVC	0	51	10	0	1	0
DEGENLPA	0	0	20	0	15	0
DEMBO7	0	0	16	0	0	21
DISC2	22	0	7	0	17	6
DISCS	21	12	0	3	18	48
DNIEPER	1	0	56	4	24	0
DRUGDISE	10	30	19	4	50	0
DUAL1	0	0	85	0	1	0
EG3	1	0	100	0	1	299
EIGENA	0	0	110	0	110	0
EIGMAXA	0	0	101	0	101	0
EIGMAXB	0	0	101	0	101	0
FCCU	0	19	0	0	8	0
FEEDLOC	0	0	87	3	19	288
FLETCHER	3	1	0	0	1	3
GRIDNETA	26	16	4	14	36	0
GRIDNETC	40	20	0	0	36	0
HAGER4	10	10	0	1	10	0

Table 4.2: Test problem characteristics.

Problem	n_{fr}	n_b	n_r	n_{fx}	m_E	m_I
HIMMELBI	0	100	0	0	0	12
HIMMELBJ	0	43	0	2	14	0
HIMMELBK	0	24	0	0	14	0
HIMMELP5	0	0	2	0	0	3
HONG	0	0	4	0	1	0
HS15	1	1	0	0	0	2
HS17	0	1	1	0	0	2
HS18	0	0	2	0	0	2
HS19	0	0	2	0	0	2
HS23	0	0	2	0	0	5
HS41	0	0	4	0	1	0
HS53	0	0	5	0	3	0
HS54	0	0	6	0	1	0
HS59	0	0	2	0	0	3
HS60	0	0	3	0	1	0
HS63	0	3	0	0	2	0
HS68	0	0	4	0	2	0
HS69	0	0	4	0	2	0
HS71	0	0	4	0	1	1
HS72	0	0	4	0	0	2
HS73	0	0	4	0	1	2
HS74	0	0	4	0	3	2
HS75	0	0	4	0	3	2
HS80	0	0	5	0	3	0
HS83	0	0	5	0	0	6
HS87	0	0	6	0	4	0
HS95	0	0	6	0	0	4
HS101	0	0	7	0	0	5
HS104	0	0	8	0	0	5
HS106	0	0	8	0	0	6
HS107	4	2	3	0	6	0
HS108	8	1	0	0	0	13
HS109	0	2	7	0	6	4
HS111	0	0	10	0	3	0
HS112	0	10	0	0	3	0
HS114	0	0	10	0	3	8
HS116	0	0	13	0	0	15
HS119	0	0	16	0	8	0
HS99EXP	21	0	7	3	21	0
HUES-MOD	0	10	0	0	2	0
HUESTIS	0	10	0	0	2	0
LEAKNET	80	70	6	0	153	0
LEWISPOL	0	0	6	0	9	0
LOTSCHD	0	12	0	0	7	0
MANNE	0	199	100	1	0	200

Table 4.3: Test problem characteristics.

Problem	n_{fr}	n_b	n_r	n_{fx}	m_E	m_I
MCONCON	10	5	0	0	11	0
MINC44	0	11	12	4	18	0
MINPERM	0	1	4	0	5	0
MISTAKE	8	1	0	0	0	13
MRIBASIS	0	0	24	12	9	46
NET1	20	0	23	5	38	19
NCVXQP1	0	0	10	0	5	0
ODFITS	0	10	0	0	6	0
OPTCDEG3	40	39	40	3	80	0
OPTCNTRL	9	10	10	3	20	0
ORTHREGE	35	1	0	0	20	0
ORTHREGF	78	2	0	0	25	0
PFIT1	2	1	0	0	3	0
PFIT2	2	1	0	0	3	0
PFIT3	2	1	0	0	3	0
PFIT4	2	1	0	0	3	0
POLYGON	0	0	48	2	0	324
PRODPL0	0	60	0	0	20	9
QR3D	145	10	0	0	155	0
QR3DBD	117	10	0	0	155	0
READING1	0	0	5	1	2	0
READING4	0	0	50	1	0	100
READING5	0	0	100	1	100	0
READING6	50	0	51	1	50	0
READING9	100	0	101	1	100	0
RK23	11	6	0	0	11	0
ROCKET	102	101	100	4	252	0
SEMICON1	0	0	10	2	10	0
SEMICON2	0	0	10	2	10	0
SINROSNB	9	0	1	0	0	18
SOSQP1	0	0	20	0	11	0
SSNLBEAM	11	0	20	2	20	0
STCQP1	0	0	17	0	8	0
STCQP2	0	0	65	0	30	0
STEENBRA	0	432	0	0	108	0
STEERING	197	1	51	7	200	0
STNQP2	0	0	65	0	30	0
SWOPF	73	0	10	0	88	14
SYNTHES2	0	2	9	0	1	14
TRAINF	200	0	200	8	202	0
TRAINH	20	0	20	8	22	0
TRUSPYR1	3	8	0	0	3	1
TWOBARS	0	0	2	0	0	2
UBH1	54	0	33	12	60	0
WATER	0	0	31	0	10	0

Table 4.4: Test problem characteristics.

Chapter 5

An inexact Gauss-Newton method

The methods proposed in Chapter 3 and their implementations rely on direct methods for linear systems and linear least-squares problems. These direct methods may be preferable to iterative methods when the cost of a matrix factorization is not excessive, e.g. if the dimension of the problem is sufficiently small or the Jacobian matrix is structured. Otherwise, it becomes necessary to use iterative methods for the solution of the linear systems and linear least-squares problems arising at each iteration.

In this chapter we present a modification of the TREBO-GN method based on iterative methods for the linear algebra phase which can be especially suited for the *large-scale* setting. In Section 5.1 we introduce a version of such procedure where a solution of the trust-region problem is approximated by the Conjugate Gradient (CG) method. Provided a suitable control on the accuracy to which we attempt to solve the trust-region problem, in Sections 5.2 and 5.3 we prove that the properties of global and local convergence of the TREBO-GN method are retained.

5.1 Description of the method

In this section we describe an inexact Gauss-Newton trust-region method, called ITREBO-GN, for solving large bound-constrained least-squares problems of the form

$$\min_{x\in\Omega} \theta(x) = \frac{1}{2}\|\Theta(x)\|^2, \tag{BCLS}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ is a given continuously differentiable mapping and Ω is the n-dimensional box $\Omega = \{x \in \mathbb{R}^n \mid l \le x \le u\}$, $l \in (\mathbb{R} \cup -\infty)^n$, $u \in (\mathbb{R} \cup \infty)^n$, $l < u$.

The basic idea of the TREBO-GN method is to take the possibly projected minimum norm step (3.7) as frequently as possible, taking advantage of its good properties in a neighbourhood of a zero-residual solution. The aim of the new method is retaining this property and at the same time avoiding the high computational cost in forming the minimum norm step by direct factorization methods.

Algorithm 5.1 Solving the trust-region problem by CG method

```
Input: $x_k$, $0 \le \eta_k < 1$, $\Delta_k > 0$.
1. Set $p_k^{(0)} = 0$ and $j = 1$;
2. Compute the $j$-th CG iterate $p_k^{(j)}$ given in (5.3);
3. If $\|p_k^{(j)}\| \le \Delta_k$ and $p_k^{(j)}$ satisfies (5.5),
       then set $p_k^I = p_k^{(j)}$, return $p_{tr} = p_k^I$.
4. If $\|p_k^{(j)}\| > \Delta_k$,
       find $\tau$ such that $p_k^{ST} = p_k^{(j-1)} + \tau(p_k^{(j)} - p_k^{(j-1)})$ satisfies $\|p_k^{ST}\| = \Delta_k$,
       return $p_{tr} = p_k^{ST}$;
5. Set $j = j + 1$ and go to Step 2.
```

The ITREBO-GN method differs from the TREBO-GN method in the solution of the trust-region problems which are solved by the CG method. Now we describe the algorithm for computing an inexact trust-region step p_{tr} and use the properties of the CG method summarized in Section 2.2.2.

Given $x_k \in \Omega$, we consider the following trust-region problem

$$\min \{m_k(p) = \frac{1}{2}\|J_k\, p + \Theta_k\|^2 \ : \ \ \|p\| \le \Delta_k\}. \tag{5.1}$$

Let $p_k^{(0)} = 0$ and $\{p_k^{(j)}\}$ be the sequence of iterates generated by the CG method applied to the normal equations

$$J_k^T J_k p = -J_k^T \Theta_k. \tag{5.2}$$

We know that for $j \ge 1$

$$p_k^{(j)} = \text{argmin}\,\{m_k(p) \ : \ p \in \mathcal{K}_j\}, \tag{5.3}$$

where $\mathcal{K}_j$ is the Krylov subspace defined in (2.19).

Let p_k^I be the first CG iterate producing a prescribed reduction of the value of ∇m_k, i.e.

$$p_k^I = \text{argmin}\,\{m_k(p) \ : \ p \in K_j\}, \qquad \|\nabla m_k(p_k^I)\| \le \eta_k \|\nabla m_k(0)\|, \tag{5.4}$$

where $\eta_k \in [0, 1)$ is the so-called forcing term. We note that by $\nabla m_k(p) = J_k^T(J_k\, p + \Theta_k)$ and (5.4) it follows

$$\begin{array}{rcl} J_k^T J_k p_k^I & = & -J_k^T \Theta_k + r_k, \\ \|r_k\| & \le & \eta_k \|J_k^T \Theta_k\|, \end{array} \tag{5.5}$$

i.e. p_k^I is an inexact Gauss-Newton step for the problem $\nabla\theta(x) = 0$ and the forcing term η_k is used to control the accuracy in the solution of the system (5.2) [15].

Since we initialize $p_k^{(0)}$ to zero, each iterate $p_k^{(j)}$ is larger in norm than its predecessor and CG terminates in a finite number of iterations computing the minimum norm step

$$p_k^N = -J_k{}^+\, \Theta_k. \tag{5.6}$$

This implies that

$$\|p_k^I\| \leq \|p_k^N\|. \tag{5.7}$$

Therefore, we stop the CG iterations as soon as either the specified accuracy (5.5) is achieved or the trust-region boundary is reached. In the former case the approximate trust-region solution is the low-dimensional unconstrained minimizer p_k^I of m_k. In the latter case, no further iterates giving lower value of m_k will be inside the trust-region. If $p_k^{(j)}$ is such that $\|p_k^{(j-1)}\| < \Delta_k \leq \|p_k^{(j)}\|$ then we take the Steihaug-Toint point p_k^{ST} given in (2.21). This process is described in Algorithm 5.1.

Except for the solution of the trust-region problem, the `ITREBO-GN` method coincides with the `TREBO-GN` method given in Algorithm 3.1. For sake of completeness, in Algorithm 5.2 we describe the k-th iteration of the `ITREBO-GN` procedure. At each iteration we set an upper bound $\eta_{max} < 1$ on the forcing term η_k so that the sequence $\{\eta_k\}$ is uniformly less than 1. This is a basic requirement for the inexact Newton framework [15].

Algorithm 5.2 `ITREBO-GN`: k-TH ITERATION

Input: $x_k \in \Omega$, $0 < \Delta_{min} \leq \Delta_k$, $0 \leq \eta_k \leq \eta_{max} < 1$, β_1, $\beta_2, \delta \in (0,1)$.

1. Compute the inexact trust-region step p_{tr} using Algorithm 5.1.
2. Let $\bar{p}_{tr} = P_\Omega(x_k + p_{tr}) - x_k$.
3. Compute the generalized Cauchy step p_k^C based on (3.11).
4. If

   $$\rho_c(\bar{p}_{tr}) = \frac{m_k(0) - m_k(\bar{p}_{tr})}{m_k(0) - m_k(p_k^C)} \geq \beta_1, \tag{5.8}$$

   Set $p_k = \bar{p}_{tr}$;

   Else find

   $$p_k = t\, p_k^C + (1-t)\bar{p}_{tr}, \tag{5.9}$$

   $t \in (0,1]$, such that (5.8) holds.
5. If

   $$\rho_\theta(p_k) = \frac{\theta(x_k) - \theta(x_k + p_k)}{m_k(0) - m_k(p_k)} \geq \beta_2, \tag{5.10}$$

   Set $x_{k+1} = x_k + p_k$, choose $\Delta_{k+1} \geq \Delta_{min}$ and $\eta_{k+1} \in [0, \eta_{max}]$;

   Else reduce Δ_k, $\Delta_k = \delta\Delta_k$ and go to Step 1.

We conclude this section making some comments on Algorithm 5.2 and pointing out the differences between the `TREBO-GN` and the `ITREBO-GN` methods.

The CG approach in the `ITREBO-GN` method substitutes the computation of the step p_k^N and possibly of the dogleg step performed in the `TREBO-GN` method, see Step 4 of Algorithm 3.1.

It is interesting to note that the `TREBO-GN` method works directly with the matrix J_k and orthogonal transformations showing very satisfactory stability properties, while the `ITREBO-GN` method works with the matrix $J_k^T J_k$. More insight the CG approach, it requires the action of J_k and J_k^T on vectors but the explicit formation of the matrix $J_k^T J_k$ can be avoided using the factorized form $J_k^T(J_k p + \Theta_k) = 0$ of the Newton equation. Working with J_k and J_k^T separately has two important advantages. First, a small perturbation in $J_k^T J_k$, e.g. by roundoff, may change the solution much more than perturbations of similar size in J_k itself. Second, we avoid the fill which can occur in the formation of $J_k^T J_k$. On the other hand, the accuracy of the computed solution may depend on the square of the conditioning of J_k. In view of these considerations, we think that as long as the computational and storage cost is not prohibitive, the `TREBO-GN` method can be competitive with the inexact approach.

5.2 Global convergence analysis

The analysis is carried out under the Assumptions 1 and 2 stated in Section 3.3. Following the lines of the proof of Lemma 3.5 it is easy to show that the `ITREBO-GN` method is well-defined i.e. the k-th iteration of the method terminates in a finite number of trials.

The global convergence properties of the method are guaranteed by imposing the condition (5.8) in Step 4 of Algorithm 5.2 and they are summarized in the following theorem.

Theorem 5.1 *Let Assumptions 1 and 2 hold and $\{x_k\}$ be the sequence generated by the* `ITREBO-GN` *method.*

i) Every limit point of the sequence $\{x_k\}$ is a first-order stationary point for the problem (BCLS).

ii) If x^* is a limit point of $\{x_k\}$ and $\|\Theta(x^*)\| = 0$, then all the limit points of $\{x_k\}$ are zero-residual solutions to problem (BCLS).

iii) If x^* is a limit point of $\{x_k\}$ such that $x^* \in int(\Omega)$ and $J(x^*)$ has full row rank, then x^* is such that $\|\Theta(x^*)\| = 0$.

Proof. See Theorems 3.1 and 3.2. □

5.3 Local convergence analysis

The local convergence analysis of the `ITREBO-GN` method parallels that of the `TREBO-GN` method. Let assume that the sequence $\{x_k\}$ generated by the `ITREBO-GN` method has a limit point x^* such that $\|\Theta(x^*)\| = 0$ and that the Jacobian $J(x^*)$ is full rank. Then, under suitable choices of the sequence $\{\eta_k\}$ of the forcing terms, the use of the inexact step p_k^I as an approximation to the minimum norm step p_k^N preserves the convergence properties of the `TREBO-GN` method.

In this section we will show that eventually there is a simple transition from the global method with a direction p_k of the form (5.9) to the projected step

$$\bar{p}_k^I = P_\Omega(x_k + p_k^I) - x_k. \tag{5.11}$$

In fact, eventually the trust-region constraint becomes inactive and $\bar{p}_k^I$ satisfies (5.8) and (5.10). As a consequence the sequence converges to x^* and, for appropriate choices of $\{\eta_k\}$, the rate of convergence is q-quadratic, see Theorems 5.2 and 5.3.

As in Section 3.3.3, we let $\mathcal{S}$ be the nonempty set of zero-residual solutions to problem (BCLS). We denote the distance from the point x to the set $\mathcal{S}$ by $d(x, \mathcal{S})$ and let $[x]_{\mathcal{S}} \in \mathcal{S}$ be such that $\|x - [x]_{\mathcal{S}}\| = d(x, \mathcal{S})$, see (3.43) and (3.44). Note that Lemma 3.2 applied with the steps p_k^I, $\bar{p}_k^I$ given in (5.4) and (5.11) provides a condition analogous to (3.52), i.e.

$$d(x_k + \bar{p}_k^I, \mathcal{S}) \le d(x_k + p_k^I, \mathcal{S}). \tag{5.12}$$

The assumptions made throughout the section are the Assumptions 3 and 5 described in Section 3.3. For sake of clarity, we restate these two assumptions in one as follows.

Assumption 6 *The sequence $\{x_k\}$ generated by the* `ITREBO-GN` *method has a limit point $x^* \in \mathcal{S}$ and $J(x^*)$ is full rank.*

In the following study we will use the results proved in Lemmas 3.1, 3.2, 3.6, 3.7 and 3.8. In particular, for subsequent references, we let α_1 and α_2 be constants defined as

$$\alpha_1 = \nu \chi_L, \quad \alpha_2 = \nu \gamma_D, \tag{5.13}$$

where χ_L and γ_D are in Assumptions 1 and 2 and reduce the constant τ of Lemma 3.8 if necessary so that

$$\alpha_1 \alpha_2 \tau \le 1/4. \tag{5.14}$$

Under Assumptions 1, 2 and 6, (5.6), (3.49) and (3.46) imply that if $x_k \in B_\tau(x^*)$ then

$$\|p_k^N\| \le \nu \, \|\Theta_k\| \le \alpha_1 \, d(x_k, \mathcal{S}). \tag{5.15}$$

5.3.1 Properties of the inexact trust-region step

The next lemma establishes that if x_k is sufficiently close to x^*, then the trust-region is inactive and the inexact step p_k^I is taken as the approximate trust-region step.

Lemma 5.1 *Let Assumptions 1, 2 and 6 hold. Then there exists $\varsigma > 0$ such that if $x_k \in B_\varsigma(x^*)$ then the trust-region solution p_{tr} computed by Algorithm 5.1 is the step p_k^I given in (5.4).*

Proof. Let $\tau > 0$ be given in Lemma 3.8 and let $x_k \in B_\tau(x^*)$. Since $x^* \in \mathcal{S}$ and (5.15) holds, there exists a scalar $\varsigma \le \tau$ sufficiently small so that if $x_k \in B_\varsigma(x^*)$ then $\|p_k^N\| \le \Delta_{min}$. Namely, the unconstrained minimum norm minimizer of the quadratic model m_k lies in the trust-region. Therefore, by the property (5.7), Algorithm 5.1 returns the step p_k^I. □

Moreover, the following technical lemma holds.

Lemma 5.2 *Let Assumptions 1, 2 and 6 hold. Let $\tau_1 \le \tau$ where τ is given in Lemma 3.8 and $\tau_2 \le \tau_1/(1+\alpha_1)$. Then if $x_k \in B_{\tau_2}(x^*)$ then*

$$x_k + p_k^I \in B_{\tau_1}(x^*), \quad x_k + p_k^I \in L, \quad [x_k + p_k^I]_{\mathcal{S}} \in L, \tag{5.16}$$

and

$$x_k + \bar{p}_k^I \in B_{\tau_1}(x^*), \quad x_k + \bar{p}_k^I \in L. \tag{5.17}$$

Proof. Note that (5.7) yields

$$\|x_k + p_k^I - x^*\| \le \|x_k - x^*\| + \|p_k^I\| \le \|x_k - x^*\| + \|p_k^N\|.$$

Then, by (5.15) we have $\|x_k + p_k^I - x^*\| \le (1+\alpha_1)\tau_2 \le \tau_1$. Hence, by Lemma 3.6 the statements in (5.16) are proved. Similarly, using the contractivity (3.22), (5.17) holds. □

The analysis of the steps p_k^I and $\bar{p}_k^I$ is the subject of the next lemmas. Lemma 5.3 concerns the overdetermined case, while Lemma 5.4 refers to the underdetermined case.

Lemma 5.3 *Let $m \ge n$ and let Assumption 1, 2 and 6 hold. Let α_1 and α_2 be the constants defined in (5.13). Then, there exists a positive constant ρ^o, such that if $x_k \in B_{\rho^o}(x^*)$*

$$\|x_k + \bar{p}_k^I - x^*\| \le \|x_k + p_k^I - x^*\| \le \phi_k^o \; \|x_k - x^*\|, \tag{5.18}$$

where

$$\phi_k^o = \alpha_0(\gamma_D(\alpha_1^2+1)\|x_k - x^*\| + \alpha_1\chi_L\eta_k). \tag{5.19}$$

Proof. Let ω and τ as in Lemma 3.7 and Lemma 3.8 respectively. Fix $x_k \in B_{\rho^o}(x^*)$, where

$$\rho^o < \min\{\omega, \tau\}/(1+\alpha_1).$$

Then by Lemma 5.2 and using (5.16) with $\tau_1 = \min\{\omega,\tau\}$ we obtain

$$x_k + p_k^I \in B_\tau(x^*), \quad x_k + p_k^I \in B_\omega(x^*), \quad x_k + p_k^I \in L, \quad [x_k + p_k^I]_{\mathcal{S}} \in L. \tag{5.20}$$

By condition (3.48) we get

$$\|x_k + p_k^I - x^*\| \le \alpha_0\|\Theta(x_k + p_k^I)\|, \tag{5.21}$$

so we need to estimate $\|\Theta(x_k + p_k^I)\|$ to prove (5.18). By (3.21) we get

$$\begin{aligned} \|\Theta(x_k + p_k^I)\| &\le \|\Theta(x_k + p_k^I) - \Theta_k - J_k p_k^I\| + \|\Theta_k + J_k p_k^I\| \\ &\le \gamma_D\|p_k^I\|^2 + \|\Theta_k + J_k p_k^I\|. \end{aligned} \tag{5.22}$$

Consider the SVD decomposition of J_k, see Appendix A.4. By Lemma 3.8 J_k is full rank. Hence, let $J_k = U_k\Sigma_k V_k^T = (U_{k,1}, U_{k,2})\Sigma_k V_k^T$ where $U_{k,1} \in \Re^{m\times n}$, $U_{k,2} \in \Re^{m\times(m-n)}$, $V_k \in \mathbb{R}^{n\times n}$ $\Sigma_k \in \mathbb{R}^{m\times n}$, $\Sigma_k = \mathrm{diag}(\varsigma_1, \ldots, \varsigma_n)$, $\varsigma_i > 0$ for all $i = 1, \ldots, n$. Then we have that

$$U_{k,1}^T = U_{k,1}^T(J_k^T)^+ J_k^T,$$

because $(J_k^T)^+J_k^T$ is the orthogonal projection onto the range of J_k, [20]. As a consequence we may write that

$$\|U_{k,1}^T(\Theta_k + J_k p_k^I)\| = \|U_{k,1}^T(J_k^T)^+J_k^T(\Theta_k + J_k p_k^I)\|.$$

If we use (3.49), (5.5) and Assumption 2 we obtain

$$\begin{aligned}\|U_{k,1}^T(\Theta_k + J_k p_k^I)\| &\le \|J_k^+\|\|J_k^T(\Theta_k + J_k p_k^I)\| \\ &\le \alpha_1\ \eta_k\|\Theta_k\|. \end{aligned} \tag{5.23}$$

Moreover we verify easily that $U_{k,2}^T J_k = 0$ and so

$$\|U_{k,2}^T(\Theta_k + J_k p_k^I)\| = \|U_{k,2}^T\Theta_k\| = \|U_{k,2}U_{k,2}^T\Theta_k\|,$$

where the last equality follows from $U_{k,2}^T U_{k,2} = I_{m-n}$. Moreover the equality $I_m = U_k U_k^T$ yields

$$U_{k,2}U_{k,2}^T\Theta_k = (I_m - U_{k,1}U_{k,1}^T)\Theta_k$$

and by $J_k p_k^N = -J_k J_k^+\Theta_k = -U_{k,1}U_{k,1}^T\Theta_k$ we get

$$U_{k,2}U_{k,2}^T\Theta_k = \Theta_k + J_k p_k^N. \tag{5.24}$$

Since p_k^N is the global minimizer of $\|\Theta_k + J_k p\|$ we obtain from (5.24) that

$$\|U_{k,2}^T(\Theta_k + J_k p_k^I)\| = \|\Theta_k + J_k p_k^N\| \le \|\Theta_k + J_k(x_k - x^*)\|.$$

From (3.21) we get

$$\|U_{k,2}^T(\Theta_k + J_k p_k^I)\| \le \gamma_D\|x_k - x^*\|^2. \tag{5.25}$$

Combining together $\|\Theta_k + J_k p_k^I\| = \|U_k^T(\Theta_k + J_k p_k^I)\|$, (5.23) and (5.25) we find that

$$\|\Theta_k+J_k p_k^I\| \le \|U_{k,1}^T(\Theta_k+J_k p_k^I)\|+\|U_{k,2}^T(\Theta_k+J_k p_k^I)\| \le \alpha_1\ \eta_k\|\Theta_k\|+\gamma_D\|x_k-x^*\|^2. \tag{5.26}$$

By (5.22), (5.7), (5.15), (5.26), (3.46) and (3.48) we obtain

$$\|\Theta(x_k + p_k^I)\| \le (\gamma_D(\alpha_1^2 + 1)\|x_k - x^*\| + \alpha_1\chi_L\eta_k)\ \|x_k - x^*\|.$$

Hence (5.21) and (3.23) give (5.18). □

The same result holds if $m \le n$ and the proof is essentially the one of Lemma 3.10.

Lemma 5.4 *Let $m \le n$ and let Assumption 1, 2 and 6 hold. Let α_1, α_2 and ν be the constants defined in (5.13) and (3.49) respectively. Then there exists a positive constant ρ^u, such that if $x_k \in B_{\rho^u}(x^*)$ and if $\eta_k \le \min\{\eta_{max}, 1/(4\alpha_1)\}$, then*

$$d(x_k + p_k^I, S) \le 2\nu(\nu\alpha_2\|\Theta_k\| + \alpha_1\eta_k)\ \|\Theta_k\|, \tag{5.27}$$

and

$$d(x_k + \bar{p}_k^I, S) \le \phi_k^u\ d(x_k, \mathcal{S}), \tag{5.28}$$

where

$$\phi_k^u = 2\alpha_1^2(\alpha_2 d(x_k, \mathcal{S}) + \eta_k). \tag{5.29}$$

Proof. Let τ as in Lemma 3.8 and such that (5.14) holds. Fix $x_k \in B_{\rho^u}(x^*)$, where

$$\rho^u < \tau/(1+2\alpha_1). \tag{5.30}$$

Then by Lemma 5.2, we get

$$x_k + p_k^I \in B_\tau(x^*), \quad x_k + p_k^I \in L, \quad [x_k + p_k^I]_{\mathcal{S}} \in L. \tag{5.31}$$

To prove the thesis we need intermediate results. Consider the sequence $\{w_{k+l}\}_l$, $l \geq 0$, of the form

$$w_k = x_k, \quad w_{k+l+1} = w_{k+l} + s_{k+l}^I, \ \ l \geq 0, \tag{5.32}$$

where s_{k+l}^I is computed by applying CG method to the linear system

$$J(w_{k+l})^T J(w_{k+l}) s = -J(w_{k+l})^T \,\Theta(w_{k+l}).$$

Specifically, starting from the null initial guess, the step s_{k+l}^I is the first CG iterate such that

$$\|\tilde{r}_{k+l}\| \leq \tilde{\eta}_{k+l} \|J(w_{k+l})^T \Theta(w_{k+l})\|, \ l \geq 0,$$

where $\tilde{r}_{k+l}$ is given by

$$\tilde{r}_{k+l} = J(w_{k+l})^T J(w_{k+l}) s_{k+l}^I + J(w_{k+l})^T \,\Theta(w_{k+l}), \ l \geq 0, \tag{5.33}$$

and $\{\tilde{\eta}_{k+l}\}_{l\geq 0}$ is a sequence of positive scalars such that $\tilde{\eta}_k = \eta_k$ and $\sup_{j\geq 0} \tilde{\eta}_{k+j} \leq 1/(4\alpha_1)$. Note that for $l = 0$, we get $s_k^I = p_k^I$. Letting

$$s_{k+l}^N = -J(w_{k+l})^+ \,\Theta(w_{k+l}), \ l \geq 0. \tag{5.34}$$

we have

$$\|s_{k+l}^I\| \leq \|s_{k+l}^N\|, \ l \geq 0. \tag{5.35}$$

First, we show that $\{w_{k+l}\}_{l\geq 0} \subseteq B_\tau(x^*)$. Second, we prove that $\{w_{k+l}\}_{l\geq 0}$ has limit point in $\mathcal{S}$. We begin proving that $\{w_{k+l}\} \subseteq B_\tau(x^*)$ by induction. The thesis trivially holds for $w_k = x_k$. Then, we suppose that $w_{k+j} \in B_\tau(x^*)$ for $j \leq l$ and show that $w_{k+l+1} \in B_\tau(x^*)$. By (5.34), (5.35), (3.49), (3.21) and Lemma 3.8 we get

$$\begin{aligned}
\|s_{k+j}^I\| \leq \|s_{k+j}^N\| &\leq \nu \|\Theta(w_{k+j})\|, && (5.36)\\
\|\Theta(w_{k+j}) - \Theta(w_{k+j-1}) - J(w_{k+j-1}) s_{k+j-1}^I\| &\leq \gamma_D \|s_{k+j-1}^I\|^2, && (5.37)\\
\|\tilde{r}_{k+j}\| &\leq \chi_L \,\tilde{\eta}_{k+j} \|\Theta(w_{k+j})\|, && (5.38)
\end{aligned}$$

for $j = 1, \ldots, l$. Moreover, from (5.33), (5.34), (3.49) and $(J_k^T)^+ J_k^T = J_k J_k{}^+ = I_m$ it follows

$$\|J(w_{k+j-1})(s_{k+j-1}^N - s_{k+j-1}^I)\| \leq \nu \|\tilde{r}_{k+j-1}\|, \quad 1 \leq j \leq l.$$

Hence from (5.38) and (3.46)

$$\begin{aligned}
\|\Theta(w_{k+j})\| &= \|\Theta(w_{k+j}) - \Theta(w_{k+j-1}) - J(w_{k+j-1}) s_{k+j-1}^N \pm J(w_{k+j-1}) s_{k+j-1}^I\| \\
&\leq \gamma_D \|s_{k+j-1}^I\|^2 + \nu \|\tilde{r}_{k+j-1}\| \\
&\leq (\nu \alpha_2 \|\Theta(w_{k+j-1})\| + \alpha_1 \tilde{\eta}_{k+j-1}) \ \|\Theta(w_{k+j-1})\| \qquad (5.39)\\
&\leq (\alpha_1 \alpha_2 \, d(w_{k+j-1}, \mathcal{S}) + \alpha_1 \tilde{\eta}_{k+j-1}) \ \|\Theta(w_{k+j-1})\| \\
&\leq (\alpha_1 \alpha_2 \tau + \alpha_1 \tilde{\eta}_{k+j-1}) \ \|\Theta(w_{k+j-1})\| \\
&\leq \frac{1}{2} \ \|\Theta(w_{k+j-1})\|,
\end{aligned}$$

for $j = 1, \ldots, l$, since $\alpha_1\alpha_2\tau \le \frac{1}{4}$ and $\sup_{j\ge 0}\tilde{\eta}_{k+j} \le 1/(4\alpha_1)$. Then it follows that

$$\|\Theta(w_{k+j})\| \le \left(\frac{1}{2}\right)^j \|\Theta_k\|, \quad 1 \le j \le l,$$

and by (5.36)

$$\|s^I_{k+j}\| \le \nu \left(\frac{1}{2}\right)^j \|\Theta_k\|, \quad 1 \le j \le l. \tag{5.40}$$

It then follows from (5.40) that

$$\begin{aligned} \|w_{k+l+1} - x^*\| &\le \sum_{j=0}^{l} \|w_{k+j+1} - w_{k+j}\| + \|x_k - x^*\| \\ &\le \sum_{j=0}^{l} \|s^I_{k+j}\| + \rho^u \\ &\le \nu\|\Theta_k\| \sum_{j=0}^{\infty} \left(\frac{1}{2}\right)^j + \rho^u, \end{aligned}$$

and (3.46) and (5.30) yield to

$$\|w_{k+l+1} - x^*\| \le 2\nu\|\Theta_k\| + \rho^u \le 2\alpha_1\, d(x_k, \mathcal{S}) + \rho^u \le (2\alpha_1 + 1)\rho^u \le \tau.$$

As a consequence, $\{w_{k+l}\} \subset B_\tau(x^*)$ and w_{k+l} satisfies Lemma 3.6 and Lemma 3.8 for all $l \ge 0$. Further, the conditions (5.36), (5.37) and (5.38) hold for $j \ge 1$.

Second, we prove that $\{w_{k+l}\}$ is a Cauchy sequence with limit point $\bar{x} \in \mathcal{S}$. In fact, letting $p > q \ge 0$ and proceeding as above we obtain

$$\|w_{k+p} - w_{k+q}\| \le \sum_{j=q}^{p-1} \|s^I_{k+j}\| \le \sum_{j=q}^{p-1} \|s^N_{k+j}\| \le \sum_{j=0}^{\infty} \|s^N_{k+j}\| \le 2\alpha_1\rho^u.$$

Thus, $\{w_{k+l}\}$ is a Cauchy sequence and the limit is denoted as $\bar{x}$. To show that $\bar{x} \in \mathcal{S}$ note that (5.33), (3.21), (3.20), (5.32), (5.38) and the property $(J(w_{k+l})^T)^+ J(w_{k+l})^T = I_m$ yield

$$\begin{aligned} \|\Theta(w_{k+l+1})\| &= \|(J(w_{k+l})^T)^+ J(w_{k+l})^T \Theta(w_{k+l+1})\| \\ &\le \|J(w_{k+l})^+\| \|J(w_{k+l})^T(\Theta(w_{k+l+1}) - J(w_{k+l})s^I_{k+l} - \Theta(w_{k+l})) + \tilde{r}_{k+l}\| \\ &\le \nu\left(\chi_L\gamma_D\|s^I_{k+l}\|^2 + \|\tilde{r}_{k+l}\|\right) \\ &\le \nu\left(\chi_L\gamma_D\|s^I_{k+l}\|^2 + \chi_L\tilde{\eta}_{k+l}\|\Theta(w_{k+l})\|\right) \\ &\le \alpha_1\gamma_D\|w_{k+l+1} - w_{k+l}\|^2 + \alpha_1\tilde{\eta}_{k+l}(\|\Theta(w_{k+l+1}) - \Theta(w_{k+l})\| + \|\Theta(w_{k+l+1})\|) \\ &\le \alpha_1\gamma_D\|w_{k+l+1} - w_{k+l}\|^2 + 1/4(\chi_L\|w_{k+l+1} - w_{k+l}\| + \|\Theta(w_{k+l+1})\|), \end{aligned}$$

for $l \ge 0$. Hence

$$\|\Theta(w_{k+l+1})\| \le \frac{4}{3}\left(\alpha_1\gamma_D\|w_{k+l+1} - w_{k+l}\| + \chi_L/4\right)\|w_{k+l+1} - w_{k+l}\|,$$

for $l \geq 0$. Since $\lim_{l\to\infty} \|w_{k+l+1} - w_{k+l}\| = 0$, it follows $\|\Theta(\bar{x})\| = \lim_{l\to\infty} \|\Theta(w_{k+l+1})\| = 0$.

Now we can prove the thesis of the lemma. Note that $\|x_k + p_k^I - \bar{x}\| = \|w_{k+1} - \bar{x}\| \leq \sum_{j=1}^{\infty} \|s_{k+j}^I\|$, see [48]. From (5.36) and (5.40) we get

$$\|x_k + p_k^I - \bar{x}\| \quad \leq \quad \sum_{j=1}^{\infty} \|s_{k+j}^I\| \leq \sum_{j=1}^{\infty} \nu \left(\frac{1}{2}\right)^{j-1} \|\Theta(w_{k+1})\| = 2\nu \|\Theta(w_{k+1})\|.$$

Then, using equation (5.39) with $j = 1$ we obtain

$$\|x_k + p_k^I - \bar{x}\| \leq 2\nu(\nu\alpha_2\|\Theta_k\| + \alpha_1\eta_k)\|\Theta_k\|. \tag{5.41}$$

Since $d(x_k + p_k^I, \mathcal{S}) \leq \|x_k + p_k^I - \bar{x}\|$, (5.27) holds. Finally, applying (5.12), (5.27) and (3.46) we easily obtain condition (5.28). □

The next lemma gives useful asymptotic bounds on quantities that will be used in the proofs of Lemma 5.6 and Lemma 5.7.

Lemma 5.5 *Let Assumptions 1, 2 and 6 hold. Let α_1, α_2 and ν be the constants defined in (5.13) and (3.49) respectively. Then there exist a constant $\hat{\tau} > 0$ such that if $x_k \in B_{\hat{\tau}}(x^*)$ then*

$$\begin{aligned} \|J_k\bar{p}_k^I + \Theta_k\| \quad &\leq \quad \chi_L d(x_k + p_k^I, \mathcal{S}) + \nu\alpha_2\|\Theta_k\|^2, && (5.42)\\ \|\Theta(x_k + \bar{p}_k^I)\|^2 - \|J_k\bar{p}_k^I + \Theta_k\|^2 \quad &\leq \quad \left(\nu^2\alpha_2^2\|\Theta_k\|^2 + 2\nu\alpha_2\|J_k\bar{p}_k^I + \Theta_k\|\right)\|\Theta_k\|^2. && (5.43)\end{aligned}$$

Proof. Let τ as in Lemma 3.8. Fix $x_k \in B_{\hat{\tau}}(x^*)$, where $\hat{\tau} < \tau/(1+\alpha_1)$. Then by Lemma 5.2, (5.16) and (5.17) hold with $\tau_1 = \tau$.

Consider the equality

$$J_k\bar{p}_k^I + \Theta_k \quad = \quad \Theta(x_k + \bar{p}_k^I) - \Theta([x_k + p_k^I]_{\mathcal{S}}) + J_k\bar{p}_k^I - (\Theta(x_k + \bar{p}_k^I) - \Theta_k).$$

Then by (3.20), (3.21), (3.23), (3.22), (5.7) and (5.15) we obtain

$$\begin{aligned} \|J_k\bar{p}_k^I + \Theta_k\| \quad &\leq \quad \chi_L\|x_k + \bar{p}_k^I - [x_k + p_k^I]_{\mathcal{S}}\| + \gamma_D\|\bar{p}_k^I\|^2\\ &\leq \quad \chi_L\|x_k + p_k^I - [x_k + p_k^I]_{\mathcal{S}}\| + \gamma_D\|p_k^N\|^2\\ &\leq \quad \chi_L d(x_k + p_k^I, \mathcal{S}) + \nu\alpha_2\|\Theta_k\|^2, \end{aligned}$$

and (5.42) is proved.

To prove (5.43) we use Theorem A.2 to get the statement

$$\Theta(x_k + \bar{p}_k^I) = \Theta_k + \int_0^1 J(x_k + t\bar{p}_k^I)\,\bar{p}_k^I\,dt + J_k\bar{p}_k^I - J_k\bar{p}_k^I.$$

Hence,

$$\begin{aligned} \|\Theta(x_k + \bar{p}_k^I)\|^2 \quad = \quad & \|J_k\bar{p}_k^I + \Theta_k\|^2 + \|\int_0^1 (J(x_k + t\bar{p}_k^I) - J_k)\,\bar{p}_k^I\,dt\|^2\\ & +2\left(\int_0^1 (J(x_k + t\bar{p}_k^I) - J_k)\,\bar{p}_k^I\,dt\right)^T \left(J_k\bar{p}_k^I + \Theta_k\right), \end{aligned}$$

and consequently by (5.15)

$$\begin{aligned}\|\Theta(x_k+\bar{p}_k^I)\|^2-\|J_k\bar{p}_k^I+\Theta_k\|^2 &\leq \gamma_D^2\|\bar{p}_k^I\|^4+2\gamma_D\|J_k\bar{p}_k^I+\Theta_k\|\,\|\bar{p}_k^I\|^2\\ &\leq \nu^2\alpha_2^2\|\Theta_k\|^4+2\nu\alpha_2\|J_k\bar{p}_k^I+\Theta_k\|\,\|\Theta_k\|^2.\end{aligned}$$

□

5.3.2 Local analysis: the overdetermined case

The next results concern the case $m \geq n$. We begin proving that under proper assumptions on the forcing sequence $\{\eta_k\}$, the step p_k^I satisfies condition (5.8) and (5.10) whenever x_k is sufficiently close to x^* and k is sufficiently large. The proof follows closely the lines of Lemma 3.15.

Lemma 5.6 *Let $m \geq n$ and let Assumptions 1, 2 and 6 hold. Assume that $\lim_{k\to\infty}\eta_k = 0$. Then $\bar{p}_k^I$ satisfies conditions (5.8) and (5.10) whenever x_k is sufficiently close to x^* and k is sufficiently large.*

Proof. Let $\psi^o \leq \min\{\rho^o,\hat{\tau}\}$, where ρ^o and $\hat{\tau}$ are given in Lemma 5.3 and Lemma 5.5 respectively and fix $x_k \in B_{\psi^o}(x^*)$.

Note that $m_k(0)-m_k(p_k^C) \leq m_k(0)$ and

$$\rho_c(\bar{p}_k^I) \geq 1-\frac{\|J_k\bar{p}_k^I+\Theta_k\|^2}{\|\Theta_k\|^2}. \tag{5.44}$$

Since $d(x_k+p_k^I,\mathcal{S}) \leq \|x_k+p_k^I-x^*\|$, (5.42), (5.18), (3.46) and (3.48) yield

$$\begin{aligned}\|J_k\bar{p}_k^I+\Theta_k\| &\leq \chi_L\phi_k^o\|x_k-x^*\|+\nu\alpha_2\|\Theta_k\|^2\\ &\leq \sigma_k^o\|x_k-x^*\|,\end{aligned} \tag{5.45}$$

where ϕ_k^o is given in (5.19) and σ_k^o is defined as

$$\sigma_k^o=\chi_L\phi_k^o+\alpha_1\alpha_2\|x_k-x^*\|. \tag{5.46}$$

Thus, (5.44), (5.45) and (3.48) give

$$\rho_c(\bar{p}_k^I) \geq 1-\left(\frac{\sigma_k^o}{\alpha_0}\right)^2.$$

Since $\lim_{k\to\infty}\eta_k = 0$, if x_k is sufficiently close to x^* and k is sufficiently large we obtain that $\bar{p}_k^I$ satisfies condition (5.8).

Let p_k^I satisfy (5.8). To prove that $\bar{p}_k^I$ satisfies (5.10), observe that $m_k(0)=\|\Theta_k\|^2/2$, $m_k(p_k^I)<m_k(0)$ and

$$\rho_\theta(\bar{p}_k^I)=1-\frac{\|\Theta(x_k+\bar{p}_k^I)\|^2-\|J_k\bar{p}_k^I+\Theta_k\|^2}{\|\Theta_k\|^2-\|J_k\bar{p}_k^I+\Theta_k\|^2}. \tag{5.47}$$

From (3.48) and (5.45) we have

$$\|\Theta_k\|^2 - \|J_k\bar{p}_k^I + \Theta_k\|^2 \quad \geq \quad \left(\frac{1}{\alpha_0^2} - (\sigma_k^o)^2\right)\|x_k - x^*\|^2. \qquad (5.48)$$

Using (5.43), (5.48), (5.45), (3.46), (3.48) we have

$$\begin{aligned} \rho_\theta(\bar{p}_k^I) \quad &\geq \quad 1 - \frac{(\nu^2\alpha_2^2\|\Theta_k\|^2 + 2\nu\alpha_2\|J_k\bar{p}_k^I + \Theta_k\|)\,\|\Theta_k\|^2}{\left(\frac{1}{\alpha_0^2} - (\sigma_k^o)^2\right)\|x_k - x^*\|^2}, \\ &\geq \quad 1 - \chi_L^2\frac{\alpha_1^2\alpha_2^2\|x_k - x^*\|^2 + 2\nu\alpha_2\sigma_k^o\|x_k - x^*\|}{\frac{1}{\alpha_0^2} - (\sigma_k^o)^2}. \qquad (5.49) \end{aligned}$$

Then, if x_k is sufficiently close to the solution x^* and k sufficiently large, the second term in (5.49) can be made less than $(1 - \beta_2)$ and $\bar{p}_k^I$ satisfies condition (5.10). □

Next theorem provides the main result on the convergence rate of the sequence generated by the ITREBO-GN method for the overdetermined case.

Theorem 5.2 *Let $m \geq n$ and let Assumptions 1, 2 and 6 hold. Then the sequence $\{x_k\}$ generated by the* ITREBO-GN *method converges to x^* q-superlinearly if $\lim_{k\to\infty}\eta_k = 0$. Moreover the convergence rate is q-quadratic if $\eta_k = O(\|\Theta_k\|)$.*

Proof. Let $\{x_{k_j}\}$ a subsequence of $\{x_k\}$ converging to x^*. By Lemmas 5.1 and 5.6 if x_{k_j} is sufficiently close to x^* and for k_j sufficiently large, then the step taken is equal to $\bar{p}_{k_j}^I$ and by (3.22), (5.7) and (5.15) $\lim_{j\to\infty}\bar{p}_{k_j}^I = 0$. Then, since x^* is an isolated limit point of $\{x_k\}$, using Lemma A.1, we conclude that $\lim_{k\to\infty} x_k = x^*$.

To establish the convergence rate of $\{x_k\}$, let x_k sufficiently near to x^* and k sufficiently large so that $x_{k+1} = x_k + \bar{p}_k^I$ and Lemma 5.3 holds. Then from (5.18)

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \leq \phi_k^o,$$

where ϕ_k^o is defined in (5.19). Since $\lim_{k+\infty}\phi_k^o = 0$, x_k converges to x^* q-superlinearly. Moreover, if $\eta_k = O(\|\Theta_k\|)$, then $\eta_k = O(\|x_k - x^*\|)$ and it follows $\|x_{k+1} - x^*\| = O(\|x_k - x^*\|^2)$, i.e. the q-quadratic rate is guaranteed. □

5.3.3 Local analysis: the underdetermined case

The main local convergence properties of the ITREBO-GN method for the case $m \leq n$ follow. First we prove a lemma whose proof follows the lines of Lemma 5.6 for the overdetermined case.

Lemma 5.7 *Let $m \leq n$ and let Assumptions 1, 2 and 6 hold. Assume that $\lim_{k\to\infty}\eta_k = 0$. Then $\bar{p}_k^I$ satisfies conditions (5.8) and (5.10) whenever x_k is sufficiently close to x^* and k is sufficiently large.*

Proof. Let $\psi^u \leq \min\{\rho^u, \hat{\tau}\}$, where ρ^u and $\hat{\tau}$ are given in Lemma 5.4 and Lemma 5.5 respectively and let k sufficiently large that $\eta_k \leq \min\{\eta_{max}, 1/(4\alpha_1)\}$. Fix $x_k \in B_{\psi^u}(x^*)$. Note that since x^* is a limit point of $\{x_k\}$ and $\lim_{k\to\infty} \eta_k = 0$, there exists an iterate x_k and a forcing term η_k satisfying the above conditions.

First we prove that $\bar{p}_k^I$ satisfies (5.8). By (5.42), (5.27) and (3.46) we obtain

$$\|J_k \bar{p}_k^I + \Theta_k\| \leq \sigma_k^u \|\Theta_k\|, \tag{5.50}$$

where

$$\sigma_k^u = \alpha_1\alpha_2(2\alpha_1 + 1)d(x_k, \mathcal{S}) + 2\alpha_1^2\eta_k. \tag{5.51}$$

Thus, (5.44), (5.50) and (3.46) give

$$\rho_c(\bar{p}_k^I) \geq 1 - (\sigma_k^u)^2,$$

Then, $\bar{p}_k^I$ satisfies condition (5.8) if x_k is sufficiently close to x^* and k is sufficiently large so that $1 - (\sigma_k^u)^2 > \beta_1$.

Second we prove that $\bar{p}_k^I$ satisfies (5.10). Let $\bar{p}_k^I$ satisfy (5.8). From (5.50), (3.46) we have

$$\|\Theta_k\|^2 - \|J_k \bar{p}_k^I + \Theta_k\|^2 \quad \geq \quad \left(1 - (\sigma_k^u)^2\right) \|\Theta_k\|^2. \tag{5.52}$$

Using (5.47), (5.43) and (5.52) we obtain

$$\begin{aligned}
\rho_\theta(\bar{p}_k^I) \quad &\geq \quad 1 - \frac{\nu^2\alpha_2^2\|\Theta_k\|^2 + 2\nu\alpha_2\|J_k\bar{p}_k^I + \Theta_k\|}{1 - (\sigma_k^u)^2}, \\
&\geq \quad 1 - \frac{\nu^2\alpha_2^2\|\Theta_k\|^2 + 2\nu\alpha_2\sigma_k^u\|\Theta_k\|}{1 - (\sigma_k^u)^2}, \\
&\geq \quad 1 - \frac{\alpha_1\alpha_2^2 d(x_k, \mathcal{S})^2 + 2\alpha_1\alpha_2\sigma_k^u d(x_k, \mathcal{S})}{1 - (\sigma_k^u)^2}.
\end{aligned} \tag{5.53}$$

Then, if x_k is sufficiently close to the solution x^* and k is sufficiently large, the second term in (5.53) can be made less than $(1 - \beta_2)$, hence $\bar{p}_k^I$ satisfies condition (5.10). □

Now we provide the main result on the behavior of the sequence $\{x_k\}$ generated by the `ITREBO-GN` method for the underdetermined case. The proof of the theorem parallels that of Theorem 3.3.

Theorem 5.3 *Let $m \leq n$ and let Assumptions 1, 2 and 6 hold. Then the sequence $\{x_k\}$ generated by the* `ITREBO-GN` *method converges to x^* q-superlinearly if $\lim_{k\to\infty} \eta_k = 0$. Moreover the convergence rate is q-quadratic if $\eta_k = O(\|\Theta_k\|)$.*

Proof. Let $\hat{k}$ be sufficiently large so that ϕ^u in (5.29) satisfies $\phi_k^u \leq \frac{1}{2}$ and $\eta_k \leq \min\{\eta_{max}, 1/(4\alpha_1)\}$ for $k \geq \hat{k}$. Let $\psi_2 \leq \min\{\varsigma, \rho^u\}$, where ς and ρ^u are given in Lemma 5.1 and Lemma 5.4 respectively, and $\bar{k} \geq \hat{k}$ be such that if $x_k \in B_{\psi_2}(x^*)$ and $k \geq \bar{k}$ then $\bar{p}_k^I$ satisfies (5.8) and (5.10). Finally, let $\zeta < \frac{\psi_2}{1+2\alpha_1}$. Fix $k \geq \bar{k}$ and $x_k \in B_\zeta(x^*)$.

We begin showing that if $x_k \in B_\zeta(x^*)$ then $x_l \in B_{\psi_2}(x^*)$ for $l > k$. We proceed by induction. First, we show that $x_{k+1} \in B_{\psi_2}(x^*)$. In fact, by (3.22) we have $\|x_{k+1} - x^*\| =$

$\|x_k + \bar{p}_k^I - x^*\| \le \zeta + \|p_k^I\|$. Thus by (5.7) and (5.15) and the definition of ζ, we get $\|x_{k+1} - x^*\| \le (1+\alpha_1)\zeta \le \psi_2$. Second, we assume $x_{k+1}, \ldots, x_{k+m-1} \in B_{\psi_2}(x^*)$, and show that $x_{k+m} \in B_{\psi_2}(x^*)$. From (5.28) it follows

$$d(x_{k+l}, \mathcal{S}) \le \phi_{k+l-1}^u \, d(x_{k+l-1}, \mathcal{S}) \le \frac{1}{2}\, d(x_{k+l-1}, \mathcal{S}) \le \cdots \le \left(\frac{1}{2}\right)^{l-1} \phi_k^u \, d(x_k, \mathcal{S}) \le \, \zeta\left(\frac{1}{2}\right)^l,$$

for $l = 1, \ldots, m$. Thus,

$$\begin{aligned} \|x_{k+m} - x^*\| &\le \|x_{k+m} - x_{k+m-1}\| + \cdots + \|x_k - x^*\| \\ &\le \sum_{l=0}^{m-1} \|\bar{p}_k^I\| + \zeta \\ &\le \alpha_1 \sum_{l=0}^{m-1} d(x_{k+l}, \mathcal{S}) + \zeta, \end{aligned}$$

where the last inequality follows from (5.7) and (5.15), and

$$\|x_{k+m} - x^*\| \le (\alpha_1 \sum_{l=0}^{m-1} \left(\frac{1}{2}\right)^l + 1)\zeta \le (\alpha_1 \sum_{l=0}^{\infty} \left(\frac{1}{2}\right)^l + 1)\zeta = (2\,\alpha_1 + 1)\zeta \le \psi_2,$$

where the last inequality is due to the choice of ζ. Note that we have $x_{k+l} = x_{k+l-1} + \bar{p}_{k+l-1}^I$ for $l > 0$. Moreover, letting $p > q \ge k$ we have

$$\|x_p - x_q\| \le \sum_{l=q}^{p-1} \|\bar{p}_l^I\| \le \alpha_1 \sum_{l=0}^{\infty} \left(\frac{1}{2}\right)^l \zeta = 2\,\alpha_1 \zeta.$$

This means that $\{x_k\}$ is a Cauchy sequence and hence it converges. Since x^* is a limit point we conclude $\lim_{k\to\infty} x_k = x^*$. To establish the convergence rate of $\{x_k\}$, let $k \ge \bar{k}$ sufficiently large so that $x_{k+j+1} \in B_{\psi_2}(x^*)$ for $j \ge 0$. By (5.15) and (5.28) we obtain

$$\|\bar{p}_{k+j+1}^I\| \le \|\bar{p}_{k+j+1}^N\| \le \alpha_1 \, d(x_{k+j+1}, \mathcal{S}) \le \alpha_1 \phi_{k+j}^u \, d(x_{k+j}, \mathcal{S}) \le \frac{1}{2}\alpha_1 \, d(x_{k+j}, \mathcal{S}).$$

Then, we proceed as above and using $\|x_{k+1} - x^*\| \le \sum_{j=0}^{\infty} \|\bar{p}_{k+j+1}^I\|$, and (5.28) we get

$$\begin{aligned} \|x_{k+1} - x^*\| &\le \alpha_1 \sum_{j=0}^{\infty} d(x_{k+j+1}, \mathcal{S}) \\ &\le \alpha_1 \sum_{j=0}^{\infty} \left(\frac{1}{2}\right)^j d(x_{k+1}, \mathcal{S}) \\ &\le 2\alpha_1 \phi_k^u \, d(x_k, \mathcal{S}). \end{aligned}$$

Hence

$$\frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} \le \frac{2\alpha_1 \phi_k^u \, d(x_k, \mathcal{S})}{d(x_k, \mathcal{S})} = 2\alpha_1 \phi_k^u.$$

Since $\lim_{k+\infty} \phi_k^u = 0$, x_k converges to x^* q-superlinearly. If moreover $\eta_k = O(\|\Theta_k\|)$, then $\eta_k = O(d(x_k, \mathcal{S}))$ by (3.46) and hence for k sufficiently large there exists $\bar{\phi} > 0$ such that $\phi_k^u \leq \bar{\phi}\, d(x_k, \mathcal{S})$, i.e.

$$\|x_{k+1} - x^*\| \leq 2\alpha_1 \phi_k^u\, d(x_k, \mathcal{S}) \leq 2\alpha_1 \bar{\phi}\, d(x_k, \mathcal{S})^2 \leq 2\alpha_1 \bar{\phi}\, \|x_k - x^*\|^2,$$

and then the q-quadratic rate is guaranteed. □

Research perspectives

In this thesis, we have presented trust-region quadratic methods for solving bound-constrained least-squares problems and nonlinear feasibility problems. The presented TREBO-LM and TREBO-GN methods, have been studied from both a theoretical and practical point of view and the TREBO-GN method has been coded into the Matlab implementation TRESNEI. An inexact version of the TREBO-GN method, named ITREBO-GN, was also proposed and its local convergence properties were analyzed.

The next step in our research will consist in implementing the ITREBO-GN method and studying its numerical behaviour. The core of the implementation will be the use of iterative solvers for the solution of the trust-region problem and the choice of a suitable accuracy requirement in the computation of the inexact trust-region step. In particular, modules from the GALAHAD [35] package, as e.g. LSTR or GLTR, would serve our purposes. As a consequence, TRESNEI will be generalized to include the ITREBO-GN implementation and Fortran 95 will be the adequate programming language used.

Another issue we are interested in addressing, is the use of ∞-norm trust-regions. All the trust-region methods presented in this thesis attempt to solve a spherical trust-region problem at each iteration. Then the feasibility is enforced by projecting the trust-region step onto the box Ω defined by the simple bounds. Since the projected trust-region step may not produce a sufficient reduction in the quadratic model and in the objective function, the solution of the trust-region problem may result to be useless. An alternative approach takes into account the shape of the bound constraints and consider an ∞-norm trust-region problem. Taking advantage of the geometry of the problem, the current quadratic model is minimized over a box resulting from the intersection of Ω and the trust-region so that the feasibility is automatically preserved.

Finally, a further reasearch development concerns the introduction of filter techniques in our trust-region framework [23]. It is well known that the combination of filter techniques with the trust-region strategy enhances the efficiency and the robustness of the trust-region method and yields to strong global convergence properties. Specifically, the filter allows a nonmonotone behaviour of the values of the objective function at the iterates and, under reasonable assumptions, every limit point of the sequence generated is actually a zero-residual solution to the bound-constrained problem.

Appendix

Appendix A

This appendix gives prerequisites to our study. We give an account of the convergence of sequences of vectors, multivariable calculus and the generalization of the inverse for any nonzero matrix via its pseudoinverse. Also, we summarize local convergence results for iterative methods applied to nonlinear least-squares problems.

A.1 Convergence of sequences

Let $x^* \in \mathbb{R}^n$ and $\{x_k\}$ be a sequence of vectors in $\mathbb{R}^n$. The sequence is said to *converge* to x^*, i.e. $\lim_{k\to\infty} x_k = x^*$, if

$$\lim_{k\to\infty} \|x_k - x^*\| = 0,$$

where $\|\cdot\|$ is a norm on $\mathbb{R}^n$. Moreover, the sequence $\{x_k\}$ is said to be a *Cauchy sequence* if, given $\epsilon > 0$, there is an integer N such that $\|x_m - x_l\| < \epsilon$ for all $m, l > N$. Note that in $\mathbb{R}^n$, a sequence $\{x_k\}$ converges if and only if it is a Cauchy sequence.

We say that x^* is a *limit point* of the sequence $\{x_k\}$ if there is some infinite subsequence of indices $k_1, k_2, \dots,$ such that $\lim_{j\to\infty} x_{k_j} = x^*$. An *isolated limit point* x^* is such that there exists a neighbourhood of x^* in which x^* is the only limit point.

The following lemma gives a characterization of the sequences in $\mathbb{R}^n$ which have isolated limit points.

Lemma A.1 *[57, Lemma 4.10] Let x^* be an isolated limit point of a sequence $\{x_k\}$ in $\mathbb{R}^n$. If $\{x_k\}$ does not converge then there is a subsequence $\{x_{k_j}\}$ which converges to x^* and an $\epsilon > 0$ such that $\|x_{k_j+1} - x_{k_j}\| \geq \epsilon$.*

Let $\{x_k\} \subset \mathbb{R}^n$ be a sequence that converges to x^* and $\|\cdot\|$ a norm in $\mathbb{R}^n$. The *convergence rate* of the sequence $\{x_k\}$ can be classified as follows.

- If there exists a constant $c \in (0, 1)$ and an integer $K \geq 0$ such that for all $k \geq K$,

  $$\|x_{k+1} - x^*\| \leq c\,\|x_k - x^*\|,$$

  then $\{x_k\}$ is said to be *q-linearly convergent* to x^*.

- If

  $$\lim_{k\to\infty} \frac{\|x_{k+1} - x^*\|}{\|x_k - x^*\|} = 0,$$

  then $\{x_k\}$ is said to converge *q-superlinearly* to x^*.

– If there exist constants $p > 1$, $c > 0$ and $K \geq 0$ such that

$$\|x_{k+1} - x^*\| \leq c\,\|x_k - x^*\|^p,$$

for each $k \geq K$, then $\{x_k\}$ is said to converge to x^* with *q-order p*. If $p = 2$ then the convergence is said to be *q-quadratic*.

A.2 Multivariable calculus

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a multivariable function. Then, it is said to be *differentiable* if all its partial derivatives

$$\frac{\partial f(x)}{\partial x_i} = \lim_{h \to 0} \frac{f(x + he_i) - f(x)}{h}, \quad i = 1, \ldots, n,$$

exist, where e_i is the i-th coordinate vector in $\mathbb{R}^n$. If this is the case, then we define the gradient of f as the vector that groups all its partial derivatives, and we denote it by

$$\nabla f(x) = \begin{pmatrix} \frac{\partial f(x)}{\partial x_1} \\ \vdots \\ \frac{\partial f(x)}{\partial x_n} \end{pmatrix}.$$

If f is differentiable, and all derivatives of f are continuous with respect to x, then we say that f is *continuously differentiable*. The second partial derivatives of f are defined by

$$\frac{\partial^2 f(x)}{\partial x_i x_j} = \frac{\partial}{\partial x_i}\left(\frac{\partial f(x)}{\partial x_j}\right), \quad 1 \leq i, j \leq n.$$

If all second partial derivatives of f exist, then f is said to be *twice differentiable*; if, furthermore, all second partial derivatives of f are continuous, we say that f is *twice continuously differentiable*. The *Hessian matrix* $\nabla^2 f(x)$ of f at x is the $n \times n$ matrix defined as

$$\left(\nabla^2 f(x)\right)_{ij} = \frac{\partial^2 f(x)}{\partial x_i x_j}, \quad 1 \leq i, j \leq n.$$

Let C be a convex subset of $\mathbb{R}^n$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is called *convex* over the set C if

$$f(\alpha x + (1 - \alpha)y) \leq \alpha f(x) + (1 - \alpha) f(y), \qquad \text{for all } \; x, y \in C, \alpha \in [0, 1].$$

The function f is called *strictly convex* over the set C if the above inequality is strict for all $x, y \in C$ with $x \neq y$ and all $\alpha \in (0, 1)$.

Let f be twice continuously differentiable over $\mathbb{R}^n$. If $\nabla^2 f(x)$ is positive semidefinite for every $x \in C$, then f is convex over C. If $\nabla^2 f(x)$ is positive definite for every $x \in C$, then f is strictly convex over C, [6, Proposition B.4].

It follows the Taylor's Theorem for multivariable functions.

Theorem A.1 *[49, Theorem 1.2.2] Let $f : \mathbb{R}^n \to \mathbb{R}$ be twice continuously differentiable in a neighborhood of a point $x \in \mathbb{R}^n$. Then for $p \in \mathbb{R}^n$ and $\|p\|$ sufficiently small*

$$f(x+p) = f(x) + \nabla f(x)^T p + \frac{1}{2} p^T \nabla^2 f(x) p + o(\|p\|^2),$$

where $\|\cdot\|$ denotes a vector norm in $\mathbb{R}^n$.

Now we consider a vector-valued function $F : \mathbb{R}^n \to \mathbb{R}^m$ and let $(F(x))_i = F_i(x)$ with $F_i : \mathbb{R}^n \to \mathbb{R}$, $i = 1, \dots, m$. We say that F is continuously differentiable if each component $F_i, i = 1, \dots, m$, is continuously differentiable. The derivative of F at x is called the *Jacobian matrix* of F at x and it is denoted by $J(x)$ where

$$(J(x))_{i,j} = \frac{\partial F_i(x)}{\partial x_j}, \quad i = 1, \dots,\ m,\ j = 1, \dots, n.$$

It follows the Mean Value Theorem for vector-valued functions.

Theorem A.2 *[17, Lemma 4.1.9] Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be continuously differentiable in an open convex set $D \subset \mathbb{R}^n$. For any x, $x + p \in D$,*

$$F(x+p) - F(x) = \int_0^1 J(x + tp) p \, dt.$$

Finally, let $\|\cdot\|$ denote a vector norm and the induced matrix norm. We say that a matrix-valued function $G : \mathbb{R}^n \to \mathbb{R}^{m \times n}$ is *Lipschitz continuous* in an open set $D \subset \mathbb{R}^n$ with Lipschitz constant γ, if there exists a positive constant γ such that for all $x, y \in D$

$$\|G(x) - G(y)\| \le \gamma \|x - y\|. \tag{A.1}$$

A.3 Nonlinear least-squares: local theory

Let consider the nonlinear least-squares problem

$$\min_{x \in \mathbb{R}^n} \theta(x) = \frac{1}{2} \|\Theta(x)\|^2, \tag{LS}$$

where $\theta : \mathbb{R}^n \to \mathbb{R}$ and $\Theta : \mathbb{R}^n \to \mathbb{R}^m$. In this section we collect results for the local convergence of the Gauss-Newton and the Levenberg-Marquardt methods presented in Section 2.2.1.

Theorem A.3 establishes local convergence results for the Gauss-Newton method assuming that the Jacobian J of Θ is full column rank at a local minimizer x^*.

Theorem A.3 *[17, Theorem 10.2.1, Corollary 10.2.2] Let $m \ge n$, $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ and let $\theta = \frac{1}{2}\|\Theta\|^2$ be twice continuously differentiable in an open convex set $D \subset \mathbb{R}^n$. Let the Jacobian J of Θ be Lipschitz continuous on D with Lipschitz constant γ and $\|J(x)\| \le \alpha$, $\forall x \in D$. Assume that there exist $x^* \in D$ and $\lambda, \sigma \ge 0$ such that $\nabla\theta(x^*) = J(x^*)^T \Theta(x^*) = 0$, λ is the smallest eigenvalue of $J(x^*)^T J(x^*)$, and*

$$\|(J(x) - J(x^*))^T \Theta(x^*)\| \le \sigma \|x - x^*\|,$$

for all $x \in D$. If $\sigma < \lambda$, then, for any $c \in (1, \lambda/\sigma)$, there exists $\epsilon > 0$ such that for all $x_0 \in B_\epsilon(x^*)$, the sequence generated by Gauss-Newton (2.7) is well-defined, converges to x^*, and satisfies

$$\|x_{k+1} - x^*\| \leq \frac{c\sigma}{\lambda}\|x_k - x^*\| + \frac{c\alpha\gamma}{2\lambda}\|x_k - x^*\|^2,$$

and

$$\|x_{k+1} - x^*\| \leq \frac{c\sigma + \lambda}{2\lambda}\|x_k - x^*\| < \|x_k - x^*\|.$$

If $\Theta(x^*) = 0$, then $\{x_k\}$ converges q-quadratically to x^*.

The local properties of the normal flow method are proved in Theorem A.4 assuming the Jacobian J full row rank at x^* such that $\Theta(x^*) = 0$.

Theorem A.4 *[70, Theorem 2.1] Let $m \leq n$ and $\Theta : \mathbb{R}^n \to \mathbb{R}^m$ be continuously differentiable in an open convex set $D \subset \mathbb{R}^n$. Suppose that in the set D the Jacobian J of Θ is full rank, Lipschitz continuous with Lipschitz constant γ and such that $\|J(x)^+\| \leq \mu$. Let $\eta > 0$ and D_η be defined as $D_\eta = \{x \in D : \|y - x\| < \eta \Rightarrow y \in D\}$. Then there is an $\epsilon > 0$ depending only on γ, μ and η such that if $x_0 \in D_\eta$ and $\|\Theta(x_0)\| < \epsilon$, then the sequence generated by the normal flow method (2.8) is well-defined and converges to a point $x^* \in D$ such that $\Theta(x^*) = 0$. Furthermore, there is a constant β for which*

$$\|x_{k+1} - x^*\| \leq \beta\|x_k - x^*\|^2, \qquad k = 0, 1, \ldots.$$

Theorem A.5 gives the convergence properties of the Levenberg-Marquardt method which are similar to those of the Gauss-Newton method given in Theorem A.3.

Theorem A.5 *[17, Theorem 10.2.6] Let the conditions of Theorem A.3 be satisfied, and let the sequence $\{\mu_k\}$ of nonnegative scalars be bounded by $b > 0$. If $\sigma < \lambda$, then for any $c \in (1, (\lambda + b)/(\sigma + b))$, there exists $\epsilon > 0$ such that for all $x_0 \in B_\epsilon(x^*)$, the sequence generated by the Levenberg-Marquardt*

$$x_{k+1} = x_k - (J_k^T J_k + \mu_k I_n)^{-1} J_k^T \Theta_k,$$

is well-defined and satisfies

$$\|x_{k+1} - x^*\| \leq \frac{c(\sigma + b)}{(\lambda + b)}\|x_k - x^*\| + \frac{c\alpha\gamma}{2(\lambda + b)}\|x_k - x^*\|^2,$$

and

$$\|x_{k+1} - x^*\| \leq \frac{c(\sigma + b) + (\lambda + b)}{2(\lambda + b)}\|x_k - x^*\| < \|x_k - x^*\|.$$

If $\Theta(x^*) = 0$ and $\mu_k = O(\|J_k^T \Theta_k\|)$, then $\{x_k\}$ converges q-quadratically to x^*.

A.4 The pseudoinverse

Any matrix $A \in \mathbb{R}^{m\times n}$ can be written as

$$A = U\Sigma V^T, \tag{A.2}$$

where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times n}$ are orthogonal matrices and $\Sigma \in \mathbb{R}^{m\times n}$ is a diagonal matrix, $\Sigma = diag(\sigma_1, \ldots .\sigma_p)$, with $p = \min\{m, n\}$ and $\sigma_i \geq 0,\ i = 1, \ldots p$. The nonnegative scalars $\sigma_i, i = 1, \ldots, p$, are called the *singular values* of A and (A.2) is called the *singular value decomposition* (SVD). Assume that $\sigma_1 \geq \sigma_1 \geq \cdots \geq \sigma_p$, so that σ_1 denotes the largest singular value of A.

If A has rank r and $r > 0$, then A has exactly r strictly positive singular values so that $\sigma_r > 0$ and $\sigma_{r+1} = \cdots = \sigma_p = 0$. If A has full rank, all its singular values are nonzero. The singular values of A are the square roots of the eigenvalues of A^TA (if $m \geq n$) or of AA^T (if $m < n$). If A is symmetric, its singular values are the absolute values of its eigenvalues [28].

It follows a perturbation result for singular values of arbitrary matrices.

Theorem A.6 *[42, Corollary 7.3.8] Let $A, B \in \mathbb{R}^{m\times n}$ and let $p = \min\{m, n\}$. If σ_i and ψ_i, $i = 1, \ldots, p$, are the singular values of A and B respectively then*

$$|\sigma_i - \psi_i| \leq \|A - B\|, \quad i = 1, \ldots, p.$$

A classical generalization of the inverse that exists for any nonzero matrix A is the *pseudoinverse*, denoted by A^+. Let r be the rank of A and let (A.2) be its SVD. The matrix $A^+ \in \mathbb{R}^{n\times m}$ is defined as

$$A^+ = V\Sigma^+U^T, \tag{A.3}$$

where $\Sigma^+ \in \mathbb{R}^{n\times m}$ is a diagonal matrix, $\Sigma^+ = diag(\sigma_1^+, \ldots .\sigma_p^+)$, with

$$\sigma_i^+ = \begin{cases} \dfrac{1}{\sigma_i} & \text{for } i = 1, \ldots, r, \\ 0 & \text{for } i = r+1, \ldots, p. \end{cases}$$

For the matrix 2-norm, it is straightforward to show that $\|A\| = \sigma_1$. The value of the 2-norm of A^+ is also related to the singular values of A. When A has rank $r > 0$, then σ_r is the smallest nonzero singular value of A. Since the nonzero diagonal entries of A^+ are the reciprocals of the nonzero singular values of A, the relation (A.3) shows that σ_r is the largest singular value of A^+. It follows immediately that

$$\|A^+\| = \frac{1}{\sigma_r}. \tag{A.4}$$

Finally, if A has full column rank then

$$A^+ = (A^TA)^{-1}A^T \text{ and } A^+A = I_n.$$

On the other hand, if A has full row rank then

$$A^+ = A^T(AA^T)^{-1} \text{ and } AA^+ = I_m.$$

Bibliography

[1] S. Bellavia, M.Macconi, B. Morini, *An affine scaling trust-region approach to bound-constrained nonlinear systems*, Appl. Numer. Math., 44 (2003), pp. 257-280.

[2] S. Bellavia, B. Morini, *An interior global method for nonlinear systems with simple bounds*, Optim. Methods Softw., 20 (2005), pp. 1-22.

[3] S. Bellavia, M. Macconi, B. Morini, *STRSCNE: A Scaled Trust Region Solver for Constrained Nonlinear Equations*, Comput. Optim. Appl., 98 (2004), pp. 31-50.

[4] S. Bellavia, B. Morini, *Subspace trust-region methods for large bound-constrained nonlinear equations*, SIAM J. Numer. Anal., 44 (2006), pp. 1535-1555.

[5] L. Bencini, R. Fantacci, L. Maccari, *Analytical model for performance analysis of IEEE 802.11 DCF mechanism in multi-radio wireless networks*, in Proceedings of International Communications Conference 2010, to appear.

[6] D.P. Bertsekas, *Nonlinear programming*, Athena Scientific, 1999.

[7] A. Bjork, *Numerical methods for least squares problems*, SIAM, Philiadephia, 1934.

[8] M.A. Branch, T.F. Coleman, Y. Li, *A subspace, interior and conjugate gradient method for large-scale bound-constrained minimization problems*, SIAM J. Sci. Comput., 21 (1999), pp. 1-23.

[9] J.V. Burke, M.C. Ferris, *A Gauss-Newton method for convex composite optimization*, Math. Program., 71 (1995), pp. 179-194.

[10] C. Cartis, N.I.M. Gould, Ph.L. Toint, *Trust-region and other regularisations of linear least-squares problems*, BIT, 49 (2009), pp. 21-53.

[11] T.F. Coleman, Y. Li, *An interior trust-region approach for nonlinear minimization subject to bounds*, SIAM J. Optim., 6 (1996), pp. 418-445.

[12] A.R. Conn, N.I.M. Gould, Ph.L. Toint, *Trust-region methods*, SMPS/SIAM Series on Optimization, 2000.

[13] J.W. Daniel, *Newton's method for nonlinear inequalities*, Numer. Math., 6 (1973), pp. 381-387.

[14] H. Dan, N. Yamashita, M. Fukushima, *Convergence properties of the inexact Levenberg-Marquardt method under local error bound conditions*, Optim. Methods Softw., 17 (2002), pp. 605-626.

[15] R.S. Dembo, S.C. Eisenstat, T. Steihaug, *Inexact Newton methods*, SIAM J. Numer. Anal., 19 (1982), pp. 400-408.

[16] J.E. Dennis, M. El-Alem, K. Williamson, *A trust-region approach to nonlinear systems of equalities and inequalities*, SIAM J. Optim., 9 (1999), pp. 291-315.

[17] J.E. Dennis, R.B. Schnabel, *Numerical methods for unconstrained optimization and nonlinear equations*, Prentice Hall, Englewood Cliffs, NJ, 1983.

[18] E.D. Dolan, J.J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., 91 (2002), 201-213.

[19] E.D. Dolan, J.J. Moré, T.S. Munson, *Optimality measures for performance profiles*, SIAM J. Optim., 16 (2006), pp. 891-909.

[20] J. Fan, Y.X. Yuan, *On the quadratic convergence of the Levenberg-Marquardt method without nonsingularity assumption*, Computing, 74 (2005), pp. 23-39.

[21] R. Fletcher, S. Leyffer, *Filter-type algorithms for solving systems of algebraic equations and inequalities*, High Performance Algorithms and Software for Nonlinear Optimization, G. Di Pillo and A. Murli, editors, Kluwer Academic Publishers, 2003, pp. 259-278.

[22] R. Fletcher, S. Leyffer *Nonlinear programming without a penalty function*, Math. Program., 91 (2002), pp. 239-270.

[23] R. Fletcher, S. Leyffer, Ph.L. Toint, *A brief history of filter methods*, SIAG/OPT Views-and-News, 18 (2007), pp. 2-12.

[24] C.A. Floudas, P.M. Pardalos, C. Adjiman, W.R. Esposito, Z.H. Gümüs, S.T. Harding, J.L. Klepeis, C.A. Meyer, C.A. Schweiger, *Handbook of test problems in local and global optimization*, Kluwer Academic Publishers, Nonconvex Optimization and its Applications, 33, 1999.

[25] J.B. Francisco, N. Krejić, J.M. Martínez, *An interior-point method for solving box-constrained underdetermined nonlinear systems*, J. Comput. Appl. Math., 177 (2005), pp. 67-88.

[26] S.A. Gabriel, J.S. Pang, *A trust region method for constrained nonsmooth equations*, in: W.W. Hager, D.W. Hearn, P.M. Pardalos (Eds.), Large Scale Optimization-State of the Art, Kluwer Academic Press, The Netherlands, 1994, pp. 155-181.

[27] U.M. Garcia-Palomares, A. Restuccia, *A global quadratic algorithm for solving a system of mixed equalities and inequalities*, Math. Program., 21 (1981), pp. 290-300.

[28] P.E. Gill, W. Murray, M.H. Wright, *Numerical linear algebra and optimization*, Volume I, Addison-Wesley, 1991.

[29] P.E. Gill, W. Murray, M.H. Wright, *Practical optimization*, Academic Press, 1981.

[30] G. H. Golub, W. Kahan, *Calculating the singular values and pseudo-inverse of a matrix*, SIAM J. Numer. Anal., 2 (1965), pp. 205-224.

[31] N.I.M. Gould, S. Leyffer, Ph.L. Toint, *A multidimensional filter algorithm for nonlinear equations and nonlinear least-squares*, SIAM J. Optim., 15 (2004), pp. 17-38.

[32] N.I.M. Gould, S. Lucidi, M. Roma, Ph. L.Toint *Solving the trust-region subproblem using the Lanczos method*, SIAM J. Optim., 9 (1999), pp. 504-525.

[33] N.I.M. Gould, D. Orban, Ph.L. Toint, *CUTEr, a constrained and unconstrained testing environment, revisited*, ACM Trans. Math. Software, 29 (2003), pp. 373-394.

[34] N.I.M. Gould, Ph.L. Toint, *FILTRANE: a Fortran 95 filter-trust-region package for solving nonlinear least-squares and nonlinear feasibility problems*, ACM Trans. Math. Software, 33 (2007), pp. 3-25.

[35] N.I.M. Gould, D.Orban, Ph.L. Toint, *GALAHAD - a library of thread-safe Fortran 90 packages for large-scale nonlinear optimization*, ACM Trans. Math. Software, 29 (2003), pp. 353-372.

[36] S. Gratton, M. Mouffe, Ph.L. Toint, *Stopping rules and backward error analysis for bound-constrained optimization*, Report 09/13, Department of Mathematics, FUNDP-University of Namur, Namur, Belgium, 2009.

[37] M. Heinkenschloss, M. Ulbrich, S. Ulbrich, *Superlinear and quadratic convergence of affine-scaling interior-point Newton methods for problems with simple bounds without strict complementarity assumptions*, Math. Program., 86 (1999), pp. 615-635.

[38] M.R. Hestenes, *Pseudoinverses and conjugate gradients*, Communications of the ACM, 18 (1975), pp. 40-43.

[39] M.R. Hestenes, E. Stiefel, *Methods of Conjugate Gradients for solving linear systems*, J. Res. Natl. Bur. Stand., 49 (1952), pp. 409-436.

[40] N.J. Higham, *The Matrix Computation Toolbox*, http://www.ma.man.ac.uk/~higham/mctoolbox.

[41] W. Hock, K. Schittkowski, *Test examples for nonlinear programming codes*, Lecture Notes in Economics and Mathematical Systems, Vol. 187, 1981.

[42] R.A. Horn, C.R. Johnson, *Matrix analysis*, The Cambridge University Press, 1985.

[43] C. Jia, D. Zhu *An affine scaling interior algorithm via Lanczos path for solving bound-constrained nonlinear systems*, Appl. Math. Comput., 195 (2008), pp. 558-575.

[44] C. Kanzow, *An active set-type Newton method for constrained nonlinear systems*, Complementarity: Applications, Algorithms and Extensions, M.C. Ferris, O.L.Mangasarian, J.S. Pang eds, Kluwer Academic Publishers, (2001), pp.179-200.

[45] C. Kanzow, A. Klug, *On affine-scaling interior-point Newton methods for nonlinear minimization with bound constraints*, Comput. Optim. Appl., 35 (2006), pp. 177-197.

[46] C. Kanzow, A. Klug, *An interior-point affine-scaling trust-region method for semismooth equations with box constraints*, Comput. Optim. Appl., 37 (2007), pp. 329-353.

[47] C. Kanzow, S. Petra, *Projected filter trust region methods for a semismooth least-squares formulation of mixed complementarity problems*, Optim. Method Soft., 22 (2007), pp. 713-735.

[48] C. Kanzow, N. Yamashita, M. Fukushima, *Levenberg-Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints*, J. Comput. Appl. Math., 172 (2004), pp. 375-397.

[49] C.T. Kelley, *Iterative methods for optimization*, Frontiers in Applied Mathematics, SIAM, 1999.

[50] D.N. Kozakevich, J.M. Martinez, S.A. Santos, *Solving nonlinear systems of equations with simple bounds*, Comput. Appl. Math., 16 (1997), pp. 215-235.

[51] D.H. Li, M. Fukushima, L. Qi, N. Yamashita, *Regularized Newton methods for convex minimization problems with singular solutions*, Comput. Optim. Appl., 28 (2004), pp. 131-147.

[52] M. Macconi, B. Morini, M. Porcelli, *Trust-region quadratic methods for nonlinear systems of mixed equalities and inequalities*, Applied Numer. Math., 59 (2009), pp. 859-876.

[53] M. Macconi, B. Morini, M. Porcelli, *A Gauss-Newton method for solving bound-constrained underdetermined nonlinear systems*, Optim. Methods Softw., 24 (2009), pp. 219-235.

[54] *Optimization Toolbox, Matlab 7*, The MathWorks, Natick, MA.

[55] R.D.C. Monteiro, J.-S. Pang, *A potential reduction Newton method for constrained equations*, SIAM J. Optim., 9 (1999), pp. 729-754.

[56] 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.

[57] J.J. Moré, D.C. Sorensen, *Computing a trust-region step*, SIAM J. Sci. Stat. Comput., 4 (1983), pp. 553-572.

[58] B. Morini, M. Porcelli, *Tresnei, a Matlab trust-region solver for systems of nonlinear equalities and inequalities*, Report 09/3, Dipartimento di Energetica, Università di Firenze.

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

[60] C.C. Paige, M.A. Saunders, *LSQR: an algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 43-71.

[61] C.C. Paige, M.A. Saunders, *ALGORITHM 583: LSQR: an algorithm for sparse linear equations and sparse least squares*, ACM Trans. Math. Software, 8 (1982), pp. 195-209.

[62] J.S. Pang, *Error bounds in mathematical programming*, Math. Program., 79 (1997), pp. 299-332.

[63] L. Qi, X.J. Tong, D.H. Li, *An active-set projected trust-region algorithm for box-constrained nonsmooth equations*, J. Optim. Theory Appl., 120 (2004), pp. 627-649.

[64] M. Shacham, N. Brauner, M. Cutlib, *A web-based library for testing performance of numerical software for solving nonlinear algebraic equations*, Comp. & Chem. Eng., 26 (2002), pp. 547-554.

[65] C. Shen, W. Xue, D. Pu *Global convergence of a tri-dimensional filter SQP algorithm based on the line search method*, Applied Numer. Math., 59 (2009), pp. 235-250.

[66] T. Steihaug, *The conjugate gradient method and trust regions in large scale optimization*, SIAM J. Numer. Anal., 20 (1983), pp. 626-637.

[67] A. Thekale, *Trust-region methods for simulation based nonlinear optimization*, Department of Mathemtics, University of Erlangen-Nuremberg, Germany, 2009.

[68] Ph.L. Toint, *Towards an efficient sparsity exploiting Newton method for minimization*, in Sparse Matrices and Their Uses (I. S. Duff, ed.), Academic Press, London (1981), pp. 57-88.

[69] X.J. Tong, L.Qi, *On the convergence of a trust-region method for solving constrained nonlinear equations with degenerate solutions*, J. Optim. Theory Appl., 123 (2004), pp. 187-211.

[70] H.F. Walker, L.T. Watson, *Least-squares secant update methods for underdetermined systems*, SIAM J. Numer. Math., 27 (1990), pp. 1227-1262.

[71] M. Ulbrich, *Nonmonotone trust-region methods for bound-constrained semismooth equations with applications to nonlinear mixed complementarity problems*, SIAM J. Optim., 11 (2000), pp. 889-917.

[72] N. Yamashita, M. Fukushima, *On the rate of convergence of the Levenberg- Marquardt method*, Computing, 15 (2001), pp. 239-249.

[73] D. Zhu, *Affine scaling interior LevenbergMarquardt method for bound-constrained semismooth equations under local error bound conditions*, J. Comput. Appl. Math., 219 (2008), pp. 198 -215.