Available online at www.sciencedirect.com
SCIENCE DIRECT®

ELSEVIER

Journal of Computational and Applied Mathematics 173 (2005) 321–343

JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS

www.elsevier.com/locate/cam

Levenberg–Marquardt methods with strong local convergence properties for solving nonlinear equations with convex constraints☆

Christian Kanzow[a], Nobuo Yamashita[b], Masao Fukushima[b,*]

[a] *Institute of Applied Mathematics and Statistics, University of Würzburg, Am Hubland, 97074 Würzburg, Germany*
[b] *Department of Applied Mathematics and Physics, Graduate School of Informatics, Kyoto University, Kyoto 606-8501, Japan*

Received 18 September 2003; received in revised form [illegible] February 2004

Abstract

We consider the problem of finding a solution of a constrained (and not necessarily square) system of equations, i.e., we consider systems of nonlinear equations and want to find a solution that belongs to a certain feasible set. To this end, we present two Levenberg–Marquardt-type algorithms that differ in the way they compute their 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. Both methods can be globalized in an easy way. Some numerical results for the second method indicate that the algorithm works quite well in practice.

Keywords: Constrained equations; Levenberg–Marquardt method; Projected gradients; Quadratic convergence; Error bounds

1. Introduction

In this paper, we consider the problem of finding a solution of the constrained system of nonlinear equations

$$F(x) = 0, \quad x \in X, \tag{1}$$

☆ This research was supported in part by a Grant-in-Aid for Scientific Research from the Ministry of Education, Science, Sports, and Culture of Japan.
* Corresponding author. Tel.: +81-757535519; fax: +81-757534756.
E-mail addresses: kanzow@mathematik.uni-wuerzburg.de (C. Kanzow), nobuo@i.kyoto-u.ac.jp (N. Yamashita), fuku@i.kyoto-u.ac.jp (M. Fukushima).

0377-0427/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2004.03.015

where $X \subseteq \mathbb{R}^n$ is a nonempty, closed and convex set and $F: \mathcal{O} \to \mathbb{R}^m$ is a given mapping defined on an open neighbourhood $\mathcal{O}$ of the set X. Note that the dimensions n and m do not necessarily coincide. We denote by X^* the set of solutions to (1).

The solution of an unconstrained square system of nonlinear equations, where $X = \mathbb{R}^n$ and $n = m$ in (1), is a classical problem in mathematics for which many well-known solution techniques like Newton's method, quasi-Newton methods, Gauss–Newton methods, Levenberg–Marquardt methods etc., are available, see, e.g., [20,5,15] for three standard books on this subject.

The solution of a constrained (and possibly nonsquare) system of equations like problem (1), however, has not been the subject of intense research. In fact, the authors are currently only aware of the recent papers [10,16,13,14,19,23,22,1,21] that deal with constrained (typically box constrained) systems of equations. Most of these papers describe algorithms that have certain global and local fast convergence properties under a nonsingularity assumption at the solution.

The nonsingularity assumption implies that the solution is locally unique. Here, we present some Levenberg–Marquardt-type algorithms that are locally quadratically convergent under a weaker assumption that, in particular, allows the solution set to be (locally) nonunique. To this end, we replace the nonsingularity assumption by an error bound condition. This is motivated by the recent paper [24] that deals with unconstrained equations only. See also [4,8] for some subsequent related results for the unconstrained case.

On the other hand, the possibility of dealing with constrained equations is very important. In fact, systems of nonlinear equations arising in several applications are often constrained. For example, in chemical equilibrium systems (see, e.g., [17,18]), the variables correspond to the concentration of certain elements that are naturally nonnegative. Furthermore, in many economic equilibrium problems, the mapping F is not defined everywhere (see, e.g., [7]) so that one is urged to impose suitable constraints on the variables. Finally, engineers often have a good guess regarding the area where they expect their solution to lie; such a priori knowledge can then easily be incorporated by adding suitable constraints to the system of equations.

The organization of this paper is as follows: Section 2 describes a constrained Levenberg–Marquardt method for the solution of problem (1). It is shown that this method has some nice local convergence properties under fairly mild assumptions. We also note that the method can be globalized quite easily. The main disadvantage of this method is that it has to solve relatively complicated subproblems at each iteration, namely (strictly convex) quadratic programs in the special case where set X is polyhedral, and convex minimization problems in the general case.

In order to avoid this drawback, we present a variant of the constrained Levenberg–Marquardt method in Section 3 (called the projected Levenberg–Marquardt method) that solves only a system of linear equations per iteration. This method is shown to have essentially the same local (and global) convergence properties as the method of Section 2. Numerical results for this method are presented in Section 4. We conclude the paper with some remarks in Section 5.

The notation used in this paper is standard: The Euclidean norm is denoted by $\|\cdot\|$, $B_\delta(x) := \{y \in \mathbb{R}^n \mid \|y - x\| \leqslant \delta\}$ is the closed ball centered at x with radius $\delta > 0$, $\mathrm{dist}(y, X^*) := \inf\{\|y - x\| \mid x \in X^*\}$ denotes the distance from a point y to the solution set X^*, and $P_X(x)$ is the projection of a point $x \in \mathbb{R}^n$ onto the feasible set X.

2. Constrained Levenberg–Marquardt method

This section describes and investigates a constrained Levenberg–Marquardt method for the solution of the constrained system of nonlinear equations (1). The algorithm and the assumptions will be given in detail in Section 2.1. The convergence of the distance from the iterates to the solution set will be discussed in Section 2.2, while Section 2.3 considers the local behavior of the iterates themselves. A globalized version of the Levenberg–Marquardt method is given in Section 2.4.

2.1. Algorithm and assumptions

For solving (1) we consider the related optimization problem

$$\min f(x) \quad \text{s.t.} \quad x \in X, \tag{2}$$

where

$$f(x) := \|F(x)\|^2$$

denotes the natural merit function corresponding to the mapping F. A Gauss–Newton-type method for this (not necessarily square) system of equations generates a sequence $\{x^k\}$ by setting $x^{k+1} := x^k + d^k$, where d^k is a solution of the linearized subproblem

$$\min f^k(d) \quad \text{s.t.} \quad x^k + d \in X \tag{3}$$

with the objective function

$$f^k(d) := \|F(x^k) + H_k d\|^2,$$

where matrix $H_k \in \mathbb{R}^{m\times n}$ is an approximation to the (not necessarily existing) Jacobian $F'(x^k)$. However, since we allow the solution set of problem (1) to be nonunique and nonisolated, we replace subproblem (3) by a regularized problem of the form

$$\min \theta^k(d) \quad \text{s.t.} \quad x^k + d \in X \tag{4}$$

with the objective function

$$\theta^k(d) := \|F(x^k) + H_k d\|^2 + \mu_k \|d\|^2, \tag{5}$$

where μ_k is a positive parameter. Note that θ^k is a strictly convex quadratic function. Hence the solution of subproblem (4) always exists uniquely.

Finally, we arrive at the following method for the solution of the constrained system of nonlinear equations (1).

Algorithm 2.1 (Constrained Levenberg–Marquardt Method: Local Version).
(S.0) Choose $x^0 \in X$, $\mu > 0$, and set $k := 0$.
(S.1) If $F(x^k) = 0$, STOP.
(S.2) Choose $H_k \in \mathbb{R}^{m\times n}$, set $\mu_k := \mu\|F(x^k)\|^2$, and compute d^k as the solution of (4).
(S.3) Set $x^{k+1} := x^k + d^k$, $k \leftarrow k+1$, and go to (S.1).

Note that the algorithm is well-defined and that all iterates x^k belong to the feasible set X. To establish our (local) convergence results for Algorithm 2.1, we need the following assumptions.

Assumption 2.2. The solution set X^* of problem (1) is nonempty. For some solution $x^* \in X^*$, there exist constants $\delta > 0$, $c_1 > 0$, $c_2 > 0$ and $L > 0$ such that the following inequalities hold:

$$c_1 \operatorname{dist}(x, X^*) \leqslant \|F(x)\| \quad \forall x \in B_\delta(x^*) \cap X, \tag{6}$$

$$\|F(x) - F(x^k) - H_k(x - x^k)\| \leqslant c_2 \|x - x^k\|^2 \quad \forall x, x^k \in B_\delta(x^*) \cap X, \tag{7}$$

$$\|F(x) - F(y)\| \leqslant L\|x - y\| \quad \forall x, y \in B_\delta(x^*) \cap X. \tag{8}$$

Assumption (6) is a local error bound condition and known to be much weaker than the more standard nonsingularity of the Jacobian $F'(x^*)$ in the case where this Jacobian exists and is a square matrix (i.e., if F is differentiable and $n=m$). For example, this local error bound condition is satisfied when F is affine and X is polyhedral. To see this, let $F(x) := Ax + a$ and $X := \{x \mid Bx \leqslant b\}$ with appropriate matrices A, B and vectors a, b. Due to Hoffman's [illegible] famous error bound result, there exists $\tau > 0$ such that

$$\tau \operatorname{dist}(x, X^*) \leqslant \|F(x)\| + \|P_X(x)\|. \tag{9}$$

If $x \in B_\delta(x^*) \cap X$ for some $x^* \in X^*$, then $P_X(x) = 0$. So, (9) reduces to $\tau \operatorname{dist}(x, X^*) \leqslant \|F(x)\|$, which implies condition (6).

Furthermore, assumption (7) may be viewed as a smoothness condition on F together with a requirement on the choice of matrix H_k. For example, this condition is satisfied with the choice $H_k := F'(x^k)$ if F is continuously differentiable with F' being locally Lipschitzian.

Finally, assumption (8) only says that F is locally Lipschitzian in a neighbourhood of the solution x^*. Of course, this condition is automatically satisfied if F is a continuously differentiable function.

2.2. Local convergence of distance function

Throughout this subsection, we suppose that Assumption 2.2 holds. The constants δ, c_1, c_2, and L that appear in the subsequent analysis are always the constants from Assumption 2.2.

Our aim is to show that Algorithm 2.1 is locally quadratically convergent in the sense that the distance from the iterates x^k to the solution set X^* goes down to zero with a quadratic rate. In order to verify this result, we need to prove a couple of technical lemmas. These lemmas can be derived by suitable modifications of the corresponding unconstrained results in [24].

Lemma 2.3. *There exist constants $c_3 > 0$ and $c_4 > 0$ such that the following inequalities hold for each $x^k \in B_{\delta/2}(x^*) \cap X$:*

(a) $\|d^k\| \leqslant c_3 \operatorname{dist}(x^k, X^*)$,
(b) $\|F(x^k) + H_k d^k\| \leqslant c_4 \operatorname{dist}(x^k, X^*)^2$.

Proof. (a) Let $\bar{x}^k \in X^*$ denote the closest solution to x^k so that

$$\|x^k - \bar{x}^k\| = \operatorname{dist}(x^k, X^*). \tag{10}$$

Since d^k is the global minimum of subproblem (4) and $x^k + \bar{d}^k \in X$ holds for the vector $\bar{d}^k := \bar{x}^k - x^k$, we have

$$\theta^k(d^k) \leqslant \theta^k(\bar{d}^k) = \theta^k(\bar{x}^k - x^k). \tag{11}$$

Furthermore, since $x^k \in B_{\delta/2}(x^*)$ by assumption, we obtain

$$\|\bar{x}^k - x^*\| \leqslant \|\bar{x}^k - x^k\| + \|x^k - x^*\| \leqslant \|x^* - x^k\| + \|x^k - x^*\| \leqslant \delta$$

so that $\bar{x}^k \in B_\delta(x^*) \cap X$. Moreover, the definition of μ_k in Algorithm 2.1 together with ([illegible]) and (10) gives

$$\mu_k = \mu\|F(x^k)\|^2 \geqslant \mu c_1^2 \,\mathrm{dist}(x^k, X^*)^2 = \mu c_1^2 \|x^k - \bar{x}^k\|^2. \tag{12}$$

Using (10), (11), (12) and (7), we obtain from the definition of the function θ^k in (5) that

$$\begin{aligned}
\|d^k\|^2 &\leqslant \frac{1}{\mu_k}\,\theta^k(d^k) \leqslant \frac{1}{\mu_k}\,\theta^k(\bar{x}^k - x^k) \\
&= \frac{1}{\mu_k}(\|F(x^k) + H_k(\bar{x}^k - x^k)\|^2 + \mu_k\|\bar{x}^k - x^k\|^2) \\
&= \frac{1}{\mu_k}\|F(x^k) - \underbrace{F(\bar{x}^k)}_{=0} - H_k(x^k - \bar{x}^k)\|^2 + \|\bar{x}^k - x^k\|^2 \\
&\leqslant \frac{1}{\mu_k}\, c_2^2\|x^k - \bar{x}^k\|^4 + \|x^k - \bar{x}^k\|^2 \\
&\leqslant \frac{c_2^2}{\mu c_1^2}\,\|x^k - \bar{x}^k\|^2 + \|x^k - \bar{x}^k\|^2 = \left(\frac{c_2^2}{\mu c_1^2} + 1\right)\mathrm{dist}(x^k, X^*)^2.
\end{aligned}$$

Therefore, statement (a) holds with $c_3 := \sqrt{(c_2^2/\mu c_1^2) + 1}$.

(b) The definition of θ^k in ([illegible]) implies

$$\|F(x^k) + H_k d^k\|^2 \leqslant \theta^k(d^k). \tag{13}$$

On the other hand, from ([illegible]), (5) and (7), we have

$$\begin{aligned}
\theta^k(d^k) &\leqslant \theta^k(\bar{x}^k - x^k) = \|F(x^k) - F(\bar{x}^k) - H_k(x^k - \bar{x}^k)\|^2 + \mu_k\|\bar{x}^k - x^k\|^2 \\
&\leqslant c_2^2\|x^k - \bar{x}^k\|^4 + \mu_k\|x^k - \bar{x}^k\|^2.
\end{aligned} \tag{14}$$

Since ([illegible]) yields

$$\mu_k = \mu\|F(x^k)\|^2 = \mu\|F(x^k) - F(\bar{x}^k)\|^2 \leqslant \mu L^2\|x^k - \bar{x}^k\|^2$$

we obtain from (13) and (14) that

$$\begin{aligned}
\|F(x^k) + H_k d^k\|^2 \leqslant \theta^k(d^k) &\leqslant c_2^2\|x^k - \bar{x}^k\|^4 + \mu_k\|x^k - \bar{x}^k\|^2 \\
&\leqslant c_2^2\|x^k - \bar{x}^k\|^4 + \mu L^2\|x^k - \bar{x}^k\|^4 \\
&= (c_2^2 + \mu L^2)\|x^k - \bar{x}^k\|^4.
\end{aligned}$$

Hence statement (b) holds with $c_4 := \sqrt{c_2^2 + \mu L^2}$. □

The next result is a major step in verifying local quadratic convergence of the distance function.

Lemma 2.4. *Assume that both* x^{k-1} *and* x^k *belong to the ball* $B_{\delta/2}(x^*)$ *for each* $k \in \mathbb{N}$. *Then there is a constant* $c_5 > 0$ *such that*

$$\operatorname{dist}(x^k, X^*) \leqslant c_5 \operatorname{dist}(x^{k-1}, X^*)^2$$

for each $k \in \mathbb{N}$.

Proof. Since $x^k, x^{k-1} \in B_{\delta/2}(x^*)$ and $x^k = x^{k-1} + d^{k-1}$, we obtain from (7) that

$$\|F(x^{k-1} + d^{k-1})\| - \|F(x^{k-1}) + H_{k-1} d^{k-1}\|$$
$$\leqslant \|F(x^{k-1}) - F(x^{k-1} + d^{k-1}) + H_{k-1} d^{k-1}\| \leqslant c_2 \|d^{k-1}\|^2.$$

Using the error bound assumption (6) and Lemma 2.3, we therefore obtain

$$\begin{aligned} c_1 \operatorname{dist}(x^k, X^*) \leqslant \|F(x^k)\| &= \|F(x^{k-1} + d^{k-1})\| \\ &\leqslant \|F(x^{k-1}) + H_{k-1} d^{k-1}\| + c_2 \|d^{k-1}\|^2 \\ &\leqslant c_4 \operatorname{dist}(x^{k-1}, X^*)^2 + c_2 c_3^2 \operatorname{dist}(x^{k-1}, X^*)^2 \\ &= (c_4 + c_2 c_3^2) \operatorname{dist}(x^{k-1}, X^*)^2, \end{aligned}$$

and this completes the proof by setting $c_5 := (c_4 + c_2 c_3^2)/c_1$. □

The next result shows that the assumption of Lemma 2.4 is satisfied if the starting point x^0 in Algorithm 2.1 is chosen sufficiently close to the solution set X^*. Let

$$r := \min\left\{ \frac{\delta}{2(1 + 2c_3)}, \frac{1}{2c_5} \right\}. \tag{15}$$

Lemma 2.5. *Assume that the starting point* $x^0 \in X$ *used in Algorithm* 2.1 *belongs to the ball* $B_r(x^*)$, *where* r *is defined by* (15). *Then all iterates* x^k *generated by Algorithm* 2.1 *belong to the ball* $B_{\delta/2}(x^*)$.

Proof. The proof is by induction on k. We start with $k=0$. By assumption, we have $x^0 \in B_r(x^*)$. Since $r \leqslant \delta/2$, this implies $x^0 \in B_{\delta/2}(x^*)$. Now let $k \geqslant 0$ be arbitrarily given and assume that $x^l \in B_{\delta/2}(x^*)$ for all $l = 0, \ldots, k$. In order to show that x^{k+1} belongs to $B_{\delta/2}(x^*)$, first note that

$$\begin{aligned} \|x^{k+1} - x^*\| &= \|x^k + d^k - x^*\| \leqslant \|x^k - x^*\| + \|d^k\| \\ &= \|x^{k-1} + d^{k-1} - x^*\| + \|d^k\| \leqslant \|x^{k-1} - x^*\| + \|d^{k-1}\| + \|d^k\| \\ &\vdots \quad \vdots \\ &\leqslant \|x^0 - x^*\| + \sum_{l=0}^{k} \|d^l\| \leqslant r + c_3 \sum_{l=0}^{k} \operatorname{dist}(x^l, X^*), \end{aligned}$$

where the last inequality follows from Lemma 2.3. Since Lemma 2.4 implies

$$\operatorname{dist}(x^l, X^*) \leqslant c_5 \operatorname{dist}(x^{l-1}, X^*)^2 \quad l = 1, \ldots, k$$

we have

$$\begin{aligned}\operatorname{dist}(x^l,X^*) &\leqslant c_5 \operatorname{dist}(x^{l-1},X^*)^2 \leqslant c_5c_5^2 \operatorname{dist}(x^{l-2},X^*)^{2^2}\\ &\;\;\vdots \quad \vdots\\ &\leqslant c_5c_5^2\cdots c_5^{2^l-1}\operatorname{dist}(x^0,X^*)^{2^l} = c_5^{2^l-1}\operatorname{dist}(x^0,X^*)^{2^l}\\ &\leqslant c_5^{2^l-1}\|x^0-x^*\|^{2^l} \leqslant c_5^{2^l-1}r^{2^l}\end{aligned}$$

for all $l=0,\ldots,k$. Using $r \leqslant 1/(2c_5)$, we therefore get

$$\begin{aligned}\|x^{k+1}-x^*\| &\leqslant r + c_3\sum_{l=0}^{k} c_5^{2^l-1}r^{2^l} = r + c_3 r\sum_{l=0}^{k} c_5^{2^l-1}r^{2^l-1}\\ &\leqslant r + c_3 r\sum_{l=0}^{k}\left(\frac{1}{2}\right)^{2^l-1} \leqslant r + c_3 r\sum_{l=0}^{\infty}\left(\frac{1}{2}\right)^{l} = r + 2c_3 r \leqslant \frac{\delta}{2},\end{aligned}$$

where the last inequality follows from definition (15) of r. This completes the induction. □

We now obtain the following quadratic convergence result for the distance function as an immediate consequence of Lemmas 2.4 and 2.5.

Theorem 2.6. *Let Assumption 2.2 be satisfied and $\{x^k\}$ be a sequence generated by Algorithm 2.1 with starting point $x^0 \in B_r(x^*)$, where r is defined by (15). Then the sequence $\{\operatorname{dist}(x^k,X^*)\}$ converges to zero quadratically, i.e., the iterates x^k approach the solution set X^* locally quadratically.*

Theorem 2.6 is the main result in this subsection and shows that the constrained Levenberg–Marquardt method of Algorithm 2.1 is locally quadratically convergent under fairly mild assumptions.

2.3. Local convergence of iterates

The aim of this subsection is to investigate the local behaviour of the sequence $\{x^k\}$ generated by Algorithm 2.1. To this end, we also assume throughout this subsection that the conditions in Assumption 2.2 are satisfied. Moreover, the constants δ and c_i, $i=1,\ldots,5$ will be those from the previous subsections, i.e., from Assumption 2.2 and Lemmas 2.3–2.5.

In view of Theorem 2.6, we know that the distance $\operatorname{dist}(x^k,X^*)$ from the iterates x^k to the solution set X^* converges to zero locally quadratically. However, this says little about the behaviour of the sequence $\{x^k\}$ itself. In this subsection, we will see that this sequence converges to a solution of (1), and that the rate of convergence is also locally quadratic.

We start by showing that the sequence is convergent.

Theorem 2.7. *Let Assumption 2.2 be satisfied and $\{x^k\}$ be a sequence generated by Algorithm 2.1 with starting point $x^0 \in B_r(x^*)$, where r is defined by (15). Then the sequence $\{x^k\}$ converges to a solution $\bar{x}$ of (1) belonging to the ball $B_{\delta/2}(x^*)$.*

Proof. Since the entire sequence $\{x^k\}$ remains in the closed ball $B_{\delta/2}(x^*)$ by Lemma 2.5, every limit point of this sequence belongs to this set, too. Hence it remains to show that the sequence $\{x^k\}$ converges. To this end, we first note that, for any positive integers k and m such that $k > m$, we have

$$\begin{aligned}\|x^k - x^m\| &= \|x^{k-1} + d^{k-1} - x^m\| \leqslant \|x^{k-1} - x^m\| + \|d^{k-1}\| \\ &= \|x^{k-2} + d^{k-2} - x^m\| + \|d^{k-1}\| \leqslant \|x^{k-2} - x^m\| + \|d^{k-2}\| + \|d^{k-1}\| \\ &\vdots \quad \vdots \\ &\leqslant \sum_{l=m}^{k-1} \|d^l\| \leqslant \sum_{l=m}^{\infty} \|d^l\|.\end{aligned}$$

Now, as in proof of Lemma 2.5, we have

$$\|d^l\| \leqslant c_3 \operatorname{dist}(x^l, X^*) \leqslant c_3 c_5^{2^l-1} r^{2^l} \leqslant c_3 r (\tfrac{1}{2})^{2^l-1} \leqslant c_3 r (\tfrac{1}{2})^l,$$

where the first inequality follows from Lemma 2.3 and the third inequality follows from $r \leqslant 1/(2c_5)$. Consequently, we get $\|x^k - x^m\| \leqslant c_3 r \sum_{l=m}^{\infty} (\frac{1}{2})^l \to 0$ as $m \to \infty$. This means $\{x^k\}$ is a Cauchy sequence and hence convergent. □

In order to prove that the sequence $\{x^k\}$ converges locally quadratically, we need some further preparatory results.

Lemma 2.8. *Let $x^0 \in B_r(x^*)$ and $\{x^k\}$ be a sequence generated by Algorithm 2.1. Then there is a constant $c_6 > 0$ such that $\operatorname{dist}(x^k, X^*) \leqslant c_6 \|d^k\|$ for all $k \in \mathbb{N}$ sufficiently large.*

Proof. In view of Theorem 2.6, we have $\operatorname{dist}(x^{k+1}, X^*) \leqslant \frac{1}{2} \operatorname{dist}(x^k, X^*)$ for all $k \in \mathbb{N}$ sufficiently large. Letting $\bar{x}^{k+1} \in X^*$ denote the closest solution to x^{k+1}, we then obtain

$$\begin{aligned}\|d^k\| &= \|x^k - x^{k+1}\| \geqslant \|x^k - \bar{x}^{k+1}\| - \|\bar{x}^{k+1} - x^{k+1}\| \\ &\geqslant \operatorname{dist}(x^k, X^*) - \operatorname{dist}(x^{k+1}, X^*) \geqslant \operatorname{dist}(x^k, X^*) - \tfrac{1}{2}\operatorname{dist}(x^k, X^*) = \tfrac{1}{2}\operatorname{dist}(x^k, X^*)\end{aligned}$$

for all k large enough. □

The next result shows that the length of the search direction d^k goes down to zero locally quadratically.

Lemma 2.9. *Let $x^0 \in B_r(x^*)$ and $\{x^k\}$ be a sequence generated by Algorithm 2.1. Then there is a constant $c_7 > 0$ such that $\|d^{k+1}\| \leqslant c_7 \|d^k\|^2$ for all $k \in \mathbb{N}$ sufficiently large.*

Proof. In view of Lemmas 2.3, 2.4, and 2.8, we have

$$\|d^{k+1}\| \leqslant c_3 \operatorname{dist}(x^{k+1}, X^*) \leqslant c_3 c_5 \operatorname{dist}(x^k, X^*)^2 \leqslant c_3 c_5 c_6^2 \|d^k\|^2$$

for all $k \in \mathbb{N}$ sufficiently large. Setting $c_7 := c_3 c_5 c_6^2$ gives the desired result. □

We next show that the length of the search direction d^k is eventually in the same order as the distance from the current iterate x^k to the limit point $\bar{x}$ of the sequence $\{x^k\}$.

Lemma 2.10. *Let $x^0 \in B_r(x^*)$ and $\{x^k\}$ be a sequence generated by Algorithm 2.1 and converging to $\bar{x}$. Then there exist constants $c_8 > 0$ and $c_9 > 0$ such that*

$$c_8\|x^k - \bar{x}\| \leqslant \|d^k\| \leqslant c_9\|x^k - \bar{x}\|$$

for all $k \in \mathbb{N}$ sufficiently large.

Proof. The right inequality holds with $c_9 := c_3$ since Lemma 2.3 implies

$$\|d^k\| \leqslant c_3 \operatorname{dist}(x^k, X^*) \leqslant c_3\|x^k - \bar{x}\|$$

for all $k \in \mathbb{N}$. In order to verify the left inequality, let $k \in \mathbb{N}$ be sufficiently large so that Lemma 2.9 applies and $c_7\|d^k\| \leqslant 1$ holds. Without loss of generality, we may also assume that $\|d^{k+1}\| \leqslant \frac{1}{2}\|d^k\|$ holds. We can then apply Lemma 2.9 successively to obtain

$$\begin{array}{lllllll}
\|d^{k+2}\| & \leqslant & c_7\|d^{k+1}\|^2 & \leqslant & (\frac{1}{2})^2 c_7\|d^k\|^2 & \leqslant & (\frac{1}{2})^2\|d^k\|, \\
\|d^{k+3}\| & \leqslant & c_7\|d^{k+2}\|^2 & \leqslant & (\frac{1}{2})^4 c_7\|d^k\|^2 & \leqslant & (\frac{1}{2})^3\|d^k\|, \\
\|d^{k+4}\| & \leqslant & c_7\|d^{k+3}\|^2 & \leqslant & (\frac{1}{2})^6 c_7\|d^k\|^2 & \leqslant & (\frac{1}{2})^4\|d^k\|, \\
& \vdots & & \vdots & & \vdots &
\end{array}$$

i.e., $\|d^{k+j}\| \leqslant (\frac{1}{2})^j\|d^k\|$ for all $j = 0, 1, 2, \ldots$. Since

$$x^{k+l} = x^k + \sum_{j=0}^{l-1} d^{k+j} \quad \text{and} \quad \bar{x} = \lim_{l\to\infty} x^{k+l}$$

we therefore get

$$\begin{aligned}
\|x^k - \bar{x}\| &= \left\| x^k - \lim_{l\to\infty} x^{k+l} \right\| = \left\| \lim_{l\to\infty} \sum_{j=0}^{l-1} d^{k+j} \right\| \\
&= \lim_{l\to\infty} \left\| \sum_{j=0}^{l-1} d^{k+j} \right\| \leqslant \lim_{l\to\infty} \sum_{j=0}^{l-1} \|d^{k+j}\| \\
&= \sum_{j=0}^{\infty} \|d^{k+j}\| \leqslant \|d^k\| \sum_{j=0}^{\infty} \left(\frac{1}{2}\right)^j = 2\|d^k\|.
\end{aligned}$$

Setting $c_8 := \frac{1}{2}$ gives the desired result. □

As a consequence of the previous lemmas, we now obtain our main local convergence result of this subsection.

Theorem 2.11. *Let Assumption* 2.2 *be satisfied and* $\{x^k\}$ *be a sequence generated by Algorithm* 2.1 *with starting point* $x^0 \in B_r(x^*)$ *and limit point* $\bar{x}$. *Then the sequence* $\{x^k\}$ *converges locally quadratically to* $\bar{x}$.

Proof. Using Lemmas 2.9 and 2.10, we immediately obtain

$$c_8\|x^{k+1} - \bar{x}\| \leqslant \|d^{k+1}\| \leqslant c_7\|d^k\|^2 \leqslant c_7c_9^2\|x^k - \bar{x}\|^2$$

for all $k \in \mathbb{N}$ sufficiently large. This shows that $\{x^k\}$ converges locally quadratically to the limit point $\bar{x}$. □

2.4. Globalized method

So far, we have presented only a local version of the constrained Levenberg–Marquardt method. Although this is the main emphasis of this paper, we also present, for the sake of completeness, a globalized version of Algorithm 2.1. The globalization given here is quite simple and might not be the best choice from the computational point of view. Nevertheless, we can show that it preserves the nice local properties of Algorithm 2.1. Throughout this subsection, we assume that the mapping F is continuously differentiable.

The globalized Levenberg–Marquardt method is based on a simple descent condition for the function $\|F(x)\|$: If a full Levenberg–Marquardt step gives a sufficient decrease of this merit function, we accept this point as the new iterate. Otherwise we switch to a projected gradient step, see, e.g., Bertsekas [3] for more details on projected gradients. Formally, the globalized method looks as follows. (Recall that we define $f(x) := \|F(x)\|^2$.)

Algorithm 2.12 (Constrained Levenberg–Marquardt Method: Globalized Version).

(S.0) Choose $x^0 \in X, \mu > 0, \beta, \sigma, \gamma \in (0,1)$, and set $k := 0$.

(S.1) If $F(x^k) = 0$, STOP.

(S.2) Choose $H_k \in \mathbb{R}^{m\times n}$, set $\mu_k := \mu\|F(x^k)\|^2$, and compute d^k as the solution of (4).

(S.3) If

$$\|F(x^k + d^k)\| \leqslant \gamma\|F(x^k)\|, \tag{16}$$

then set $x^{k+1} := x^k + d^k$, $k \leftarrow k+1$, and go to (S.1); otherwise go to (S.4).

(S.4) Compute a stepsize $t_k = \max\{\beta^\ell \mid \ell = 0,1,2,\ldots\}$ such that

$$f(x^k(t_k)) \leqslant f(x^k) + \sigma\nabla f(x^k)^{\mathrm{T}}(x^k(t_k) - x^k),$$

where $x^k(t) := P_X[x^k - t\nabla f(x^k)]$. Set $x^{k+1} := x^k(t_k), k \leftarrow k+1$, and go to (S.1).

The convergence properties of Algorithm 2.12 are summarized in the following theorem.

Theorem 2.13. *Let* $\{x^k\}$ *be a sequence generated by Algorithm* 2.12. *Then any accumulation point of this sequence is a stationary point of* (2). *Moreover, if an accumulation point* x^* *of the sequence* $\{x^k\}$ *is a solution of* (1) *and Assumption* 2.2 *is satisfied at this point, then the entire sequence* $\{x^k\}$ *converges to* x^*, *the rate of convergence is locally quadratic, and the sequence* $\{\mathrm{dist}(x^k, X^*)\}$ *also converges locally quadratically.*

Based on our previous results, the proof can be carried out in exactly the same way as that of Theorem 3.1 in [24]. We therefore skip the details here.

3. Projected Levenberg–Marquardt method

This section deals with another Levenberg–Marquardt method for the solution of constrained nonlinear systems. The main difference from Algorithm 2.1 lies in the fact that the search direction can be obtained by the solution of a single system of linear equations rather than a constrained optimization problem. This method is shown to have the same convergence properties as the Levenberg–Marquardt method of Algorithm 2.1.

The organization of this section is similar to the previous one. We first state the algorithm and assumptions in Section 3.1. Then we investigate the local behaviour of the distance function in Section 3.2. Section 3.3 deals with the local behaviour of the iterates. Finally, Section 3.4 contains a simple globalization strategy for the modified Levenberg–Marquardt method.

3.1. Algorithm and assumptions

We consider again the constrained system of nonlinear equations (1). In the previous section, we presented a constrained Levenberg–Marquardt method that generates a sequence $\{x^k\}$ by

$$x^{k+1} := x^k + d^k \quad k = 0, 1, \ldots,$$

where d^k is the solution of the constrained optimization problem

$$\min \theta^k(d) \quad \text{s.t.} \quad x^k + d \in X$$

with θ^k being defined by [illegible].

In this section, we adopt a different approach that uses the formula

$$x^{k+1} := P_X(x^k + d_U^k) \quad k = 0, 1, \ldots,$$

where d_U^k is the unique solution of the *unconstrained* (hence the subscript 'U') subproblem

$$\min \theta^k(d_U) \quad d_U \in \mathbb{R}^n.$$

We call this the projected Levenberg–Marquardt method since the unconstrained step gets projected onto the feasible region X. Note that, whenever the projection can be carried out efficiently (like in the box constrained case), this method needs a significantly less amount of work per iteration since the strict convexity of the function θ^k ensures that d_U^k is a global minimum of this function if and only if $\nabla\theta^k(d_U^k) = 0$, i.e., if and only if d_U^k is the unique solution of the system of linear equations

$$(H_k^T H_k - \mu_k I) d_U = -H_k^T F(x^k). \tag{17}$$

Specifically we consider the following algorithm.

Algorithm 3.1 (Projected Levenberg–Marquardt Method: Local Version).

(S.0) Choose $x^0 \in X, \mu > 0$, and set $k := 0$.

(S.1) If $F(x^k) = 0$, STOP.

(S.2) Choose $H_k \in \mathbb{R}^{m\times n}$, set $\mu_k := \mu\|F(x^k)\|^2$, and compute d_U^k as the solution of (17).
(S.3) Set $x^{k+1} := P_X(x^k + d_U^k)$, $k \leftarrow k+1$, and go to (S.1).

Note that the algorithm is well-defined since the coefficient matrix in (17) is always symmetric positive definite. Furthermore, all iterates x^k belong to the feasible set X.

The following assumption is supposed to hold throughout this section.

Assumption 3.2. The solution set X^* of problem (1) is nonempty. For some solution $x^* \in X^*$, there exists constants $\varepsilon > 0$, $\kappa_1 > 0$, $\kappa_2 > 0$ and $L > 0$ such that the following inequalities hold:

$$\kappa_1 \operatorname{dist}(x, X^*) \leqslant \|F(x)\| \quad \forall x \in B_\varepsilon(x^*), \tag{18}$$

$$\|F(x) - F(x^k) - H_k(x - x^k)\| \leqslant \kappa_2\|x - x^k\|^2 \quad \forall x, x^k \in B_\varepsilon(x^*), \tag{19}$$

$$\|F(x) - F(y)\| \leqslant L\|x - y\| \quad \forall x, y \in B_\varepsilon(x^*). \tag{20}$$

We tacitly assume that the constant $\varepsilon > 0$ in Assumption 3.2 is taken sufficiently small so that the mapping F is defined in the entire ball $B_\varepsilon(x^*)$. Note that this is always possible since F is assumed to be defined on an open set $\mathcal{O}$ containing the feasible region X.

Apart from this, the only difference between Assumptions 2.1 and 3.2 lies in the fact that we now assume that conditions (18)–(20) hold in the entire ball $B_\varepsilon(x^*)$, whereas before it was only assumed that the corresponding conditions (6)–(8) hold in the intersection $B_\delta(x^*) \cap X$. The reason for this slight modification is that we sometimes have to apply conditions (18)–(20) to the vector $x^k + d_U^k$ that may lie outside X.

Without the restriction on X, condition (18) is more restrictive than the corresponding condition (6). Whenever there exists a point x such that $F(x) = 0$ and $x \notin X$, (18) may fail even if F is affine and X is polyhedral. Nevertheless, condition (18) is still significantly weaker than the nonsingularity of the Jacobian of F. To see this, consider the example with $F : \mathbb{R}^2 \to \mathbb{R}$ and $X \subseteq \mathbb{R}^2$ being defined by

$$F(x) = \sqrt{x_1^2 + x_2^2} - 1 \quad \text{and} \quad X = \{x \mid -1 \leqslant x_1 \leqslant 1, -1 \leqslant x_2 \leqslant 0\}, \tag{21}$$

respectively. Note that the solution set of $F(x) = 0$ without the constraint is the unit circle, while the solution set of the constrained equation $F(x) = 0$, $x \in X$, is the lower half of the unit circle. By substituting $x = (r\cos\theta, r\sin\theta)$ with $r \geqslant 0$, we have $|F(x)| = |r - 1|$. It is easy to see that $\operatorname{dist}(x, X^*) = |r - 1|$ when x is an interior point of X. Therefore (18) holds on the interior of X. However, when $x^* = (-1, 0)^{\mathrm{T}}$, which is a boundary point of X, (18) fails since $F(x) = 0$ but $\operatorname{dist}(x, X^*) > 0$ for any x such that $r = 1$ and $0 < \theta < \pi$. On the other hand, when $x^* = (0, -1)^{\mathrm{T}}$, which is also a boundary point of X, (18) is satisfied for sufficiently small $\varepsilon > 0$.

3.2. Local convergence of distance function

This subsection deals with the behaviour of the sequence $\{\operatorname{dist}(x^k, X^*)\}$. The analysis is similar to that of Section 2.2, and many of our results can be found in the related paper [24] that deals with the convergence properties of a Levenberg–Marquardt method for the solution of unconstrained systems of equations. We therefore skip some of the proofs here.

Lemma 3.3. *There exist constants $\kappa_3 > 0$ and $\kappa_4 > 0$ such that the following inequalities hold for each $x^k \in B_{\varepsilon/2}(x^*)$:*

(a) $\|d_U^k\| \leqslant \kappa_3 \operatorname{dist}(x^k, X^*)$,
(b) $\|F(x^k) + H_k d_U^k\| \leqslant \kappa_4 \operatorname{dist}(x^k, X^*)^2$.

Proof. The proof is similar to Lemma 2.3 and may also be found in [2…]. □

We next state the counterpart of Lemma 2.4. Note, however, that the vector $x^{k-1} + d_U^{k-1}$ is no longer equal to the next iterate x^k in the method considered here. Hence the assumption in the following result is somewhat different from the assumption in the corresponding result in Lemma 2.4.

Lemma 3.4. *Assume that both x^{k-1} and $x^{k-1} + d_U^{k-1}$ belong to the ball $B_{\varepsilon/2}(x^*)$ for each $k \in \mathbb{N}$. Then there is a constant $\kappa_5 > 0$ such that $\operatorname{dist}(x^k, X^*) \leqslant \kappa_5 \operatorname{dist}(x^{k-1}, X^*)^2$ for each $k \in \mathbb{N}$.*

Proof. The definition of x^k and the nonexpansiveness of the projection operator imply that

$$
\begin{aligned}
\kappa_1 \operatorname{dist}(x^k, X^*) &= \kappa_1 \operatorname{dist}(P_X(x^{k-1} + d_U^{k-1}), X^*) \\
&= \kappa_1 \inf_{\bar{x} \in X^*} \|P_X(x^{k-1} + d_U^{k-1}) - \bar{x}\| \\
&= \kappa_1 \inf_{\bar{x} \in X^*} \|P_X(x^{k-1} + d_U^{k-1}) - P_X(\bar{x})\| \\
&\leqslant \kappa_1 \inf_{\bar{x} \in X^*} \|x^{k-1} + d_U^{k-1} - \bar{x}\| \\
&= \kappa_1 \operatorname{dist}(x^{k-1} + d_U^{k-1}, X^*) \leqslant \|F(x^{k-1} + d_U^{k-1})\|,
\end{aligned} \tag{22}
$$

where the last inequality follows from (18) together with our assumption that $x^{k-1} + d_U^{k-1} \in B_{\varepsilon/2}(x^*)$. Now, using (19) as well as x^{k-1}, $x^{k-1} + d_U^{k-1} \in B_{\varepsilon/2}(x^*)$, we have

$$
\begin{aligned}
\|F(x^{k-1} + d_U^{k-1})\| &- \|F(x^{k-1}) + H_{k-1} d_U^{k-1}\| \\
&\leqslant \|F(x^{k-1}) - F(x^{k-1} + d_U^{k-1}) + H_{k-1} d_U^{k-1}\| \leqslant \kappa_2 \|d_U^{k-1}\|^2.
\end{aligned} \tag{23}
$$

Using (22), (23) and Lemma 3.3, we obtain

$$
\begin{aligned}
\kappa_1 \operatorname{dist}(x^k, X^*) &\leqslant \|F(x^{k-1}) + H_{k-1} d_U^{k-1}\| + \kappa_2 \|d_U^{k-1}\|^2 \\
&\leqslant \kappa_4 \operatorname{dist}(x^{k-1}, X^*)^2 + \kappa_2 \kappa_3^2 \operatorname{dist}(x^{k-1}, X^*)^2 \\
&= (\kappa_4 + \kappa_2 \kappa_3^2) \operatorname{dist}(x^{k-1}, X^*)^2.
\end{aligned}
$$

This completes the proof by setting $\kappa_5 := (\kappa_4 + \kappa_2 \kappa_3^2)/\kappa_1$. □

The next result is the counterpart of Lemma 2.5 and states that the assumptions in Lemma 3.4 are satisfied if the starting point x^0 is chosen sufficiently close to the solution set. Let

$$r := \min\left\{\frac{\varepsilon}{2(1+2\kappa_3)}, \frac{1}{2\kappa_5}\right\}. \tag{24}$$

Lemma 3.5. *Assume that the starting point $x^0 \in X$ used in Algorithm 3.1 belongs to the ball $B_r(x^*)$, where x^* denotes a solution of (1) satisfying Assumption 3.2 and r is defined by (24). Then x^{k-1}, $x^{k-1}+d_U^{k-1} \in B_{\varepsilon/2}(x^*)$ holds for all $k \in \mathbb{N}$.*

Proof. The proof is by induction on k. We start with $k=1$. By assumption, we have $x^0 \in B_r(x^*)$. Since $r \leqslant \varepsilon/2$, this implies $x^0 \in B_{\varepsilon/2}(x^*)$. Furthermore, we obtain from Lemma 3.3

$$\begin{aligned}\|x^0 + d_U^0 - x^*\| &\leqslant \|x^0 - x^*\| + \|d_U^0\| \leqslant r + \|d_U^0\| \\ &\leqslant r + \kappa_3 \operatorname{dist}(x^0, X^*) \leqslant r + \kappa_3\|x^0 - x^*\| \leqslant (1+\kappa_3)r.\end{aligned}$$

Since $(1+\kappa_3)r \leqslant \varepsilon/2$, it follows that $x^0 + d_U^0 \in B_{\varepsilon/2}(x^*)$.

Now let $k \geqslant 1$ be arbitrarily given and assume that x^{l-1}, $x^{l-1} + d_U^{l-1} \in B_{\varepsilon/2}(x^*)$ for all $l = 1, \ldots, k$. We have to show that x^k and $x^k + d_U^k$ belong to $B_{\varepsilon/2}(x^*)$. Since $x^{k-1} + d_U^{k-1} \in B_{\varepsilon/2}(x^*)$, we immediately obtain $x^k = P_X(x^{k-1} + d_U^{k-1}) \in B_{\varepsilon/2}(x^*)$ from the inequality

$$\|x^k - x^*\| = \|P_X(x^{k-1} + d_U^{k-1}) - P_X(x^*)\| \leqslant \|x^{k-1} + d_U^{k-1} - x^*\|.$$

To see that $x^k + d_U^k \in B_{\varepsilon/2}(x^*)$, first note that

$$\begin{aligned}\|x^k + d_U^k - x^*\| &\leqslant \|x^k - x^*\| + \|d_U^k\| \\ &= \|x^{k-1} + d_U^{k-1} - x^*\| + \|d_U^k\| \\ &\leqslant \|x^{k-1} - x^*\| + \|d_U^{k-1}\| + \|d_U^k\| \\ &\;\;\vdots \\ &\leqslant \|x^0 - x^*\| + \sum_{l=0}^{k} \|d_U^l\| \leqslant r + \kappa_3 \sum_{l=0}^{k} \operatorname{dist}(x^l, X^*),\end{aligned}$$

where the last inequality follows from Lemma 3.3. Using Lemma 3.4, the induction can then be completed by following the arguments in the proof of Lemma 2.5. □

We are now able to state our main local convergence result of this subsection. It is an immediate consequence of Lemmas 3.4 and 3.5.

Theorem 3.6. *Let Assumption 3.2 be satisfied and $\{x^k\}$ be a sequence generated by Algorithm 3.1 with starting point $x^0 \in B_r(x^*)$, where r is defined by (24). Then the sequence $\{\operatorname{dist}(x^k, X^*)\}$ converges to zero locally quadratically.*

3.3. Local convergence of iterates

This subsection deals with the local behaviour of the sequence $\{x^k\}$ itself. In order to investigate its behaviour, we suppose that Assumption 3.2 holds throughout this subsection. Our first result states that the sequence $\{x^k\}$ generated by Algorithm 3.1 is convergent.

Theorem 3.7. *Let Assumption* 3.2 *be satisfied and* $\{x^k\}$ *be a sequence generated by Algorithm* 3.1 *with starting point* $x^0 \in B_r(x^*)$, *where* r *is defined by* (24). *Then the sequence* $\{x^k\}$ *converges to a solution* $\bar{x}$ *of* (1) *belonging to the ball* $B_{\varepsilon/2}(x^*)$.

Proof. Similar to the proof of Theorem 2.7, we verify that $\{x^k\}$ is a Cauchy sequence. Indeed, for any integers k and m such that $k > m$, we have

$$\begin{aligned}
\|x^k - x^m\| &= \|P_X(x^{k-1} + d_U^{k-1}) - P_X(x^m)\| \\
&\leqslant \|x^{k-1} + d_U^{k-1} - x^m\| \leqslant \|x^{k-1} - x^m\| + \|d_U^{k-1}\| \\
&= \|P_X(x^{k-2} + d_U^{k-2}) - P_X(x^m)\| + \|d_U^{k-1}\| \\
&\leqslant \|x^{k-2} + d_U^{k-2} - x^m\| + \|d_U^{k-1}\| \\
&\leqslant \|x^{k-2} - x^m\| + \|d_U^{k-2}\| + \|d_U^{k-1}\| \\
&\vdots \quad \vdots \\
&\leqslant \sum_{l=m}^{k-1} \|d_U^l\| \leqslant \sum_{l=m}^{\infty} \|d_U^l\|.
\end{aligned}$$

The rest of the proof is identical to that of Theorem 2.7. □

We next want to show that the sequence $\{x^k\}$ is locally quadratically convergent. To this end, we begin with the following preliminary result.

Lemma 3.8. *Let* $x^0 \in B_r(x^*)$ *and* $\{x^k\}$ *be a sequence generated by Algorithm* 3.1. *Then there is a constant* $\kappa_6 > 0$ *such that* $\operatorname{dist}(x^k, X^*) \leqslant \kappa_6 \|d_U^k\|$ *for all* $k \in \mathbb{N}$ *sufficiently large.*

Proof. The proof is a modification of that of Lemma 2.8. First note that Theorem 3.6 implies that $\operatorname{dist}(x^{k+1}, X^*) \leqslant \frac{1}{2}\operatorname{dist}(x^k, X^*)$ for all $k \in \mathbb{N}$ sufficiently large. Let $\bar{x}^{k+1}$ be the closest solution to x^{k+1}, i.e., $\operatorname{dist}(x^{k+1}, X^*) = \|x^{k+1} - \bar{x}^{k+1}\|$. Then we obtain from the nonexpansiveness of the projection operator

$$\begin{aligned}
\|d_U^k\| &= \|x^k + d_U^k - x^k\| \geqslant \|P_X(x^k + d_U^k) - P_X(x^k)\| \\
&= \|x^{k+1} - x^k\| \geqslant \|\bar{x}^{k+1} - x^k\| - \|x^{k+1} - \bar{x}^{k+1}\| \\
&\geqslant \operatorname{dist}(x^k, X^*) - \operatorname{dist}(x^{k+1}, X^*) \geqslant \operatorname{dist}(x^k, X^*) - \tfrac{1}{2}\operatorname{dist}(x^k, X^*) = \tfrac{1}{2}\operatorname{dist}(x^k, X^*)
\end{aligned}$$

for all $k \in \mathbb{N}$ large enough. □

The next result shows that the length of the unconstrained search direction d_U^k goes down to zero locally quadratically.

Lemma 3.9. *Let $x^0 \in B_r(x^*)$ and $\{x^k\}$ be a sequence generated by Algorithm* 3.1. *Then there is a constant $\kappa_7 > 0$ such that $\|d_U^{k+1}\| \leqslant \kappa_7 \|d_U^k\|^2$ for all $k \in \mathbb{N}$ sufficiently large.*

Proof. Lemmas 3.3, 3.4, and 3.8 immediately imply

$$\|d_U^{k+1}\| \leqslant \kappa_3 \operatorname{dist}(x^{k+1}, X^*) \leqslant \kappa_3 \kappa_5 \operatorname{dist}(x^k, X^*)^2 \leqslant \kappa_3 \kappa_5 \kappa_6^2 \|d_U^k\|^2$$

for all $k \in \mathbb{N}$ sufficiently large. The desired result then follows by setting $\kappa_7 := \kappa_3 \kappa_5 \kappa_6^2$. □

We next state the counterpart of Lemma 2.10 that relates the length of d_U^k with the distance from the iterates x^k to their limit point $\bar{x}$.

Lemma 3.10. *Let $x^0 \in B_r(x^*)$ and $\{x^k\}$ be a sequence generated by Algorithm* 3.1 *and converging to $\bar{x}$. Then there exist constants $\kappa_8 > 0$ and $\kappa_9 > 0$ such that*

$$\kappa_8 \|x^k - \bar{x}\| \leqslant \|d_U^k\| \leqslant \kappa_9 \|x^k - \bar{x}\|$$

for all $k \in \mathbb{N}$ sufficiently large.

Proof. Lemma 3.3(a) yields the right inequality with $\kappa_9 := \kappa_3$. We will show the left inequality. Following the proof of Lemma 2.10 and exploiting Lemma 3.9 (instead of Lemma 2.9), we can show that the following inequality holds for some sufficiently large (but fixed) index $k \in \mathbb{N}$:

$$\|d_U^{k+j}\| \leqslant \left(\tfrac{1}{2}\right)^j \|d_U^k\| \quad \text{for all } j = 0, 1, 2, \ldots$$

Furthermore, the nonexpansiveness of the projection operator yields

$$\begin{aligned}
\|x^k - x^{k+l}\| &= \|P_X(x^k) - P_X(x^{k+l-1} + d_U^{k+l-1})\| \leqslant \|x^k - x^{k+l-1} - d_U^{k+l-1}\| \\
&\leqslant \|x^k - x^{k+l-1}\| + \|d_U^{k+l-1}\| \\
&\;\;\vdots \qquad \vdots \\
&\leqslant \sum_{j=0}^{l-1} \|d_U^{k+j}\|.
\end{aligned}$$

Since $\bar{x} = \lim_{l \to \infty} x^{k+l}$, we therefore obtain from the continuity of the norm

$$\begin{aligned}
\|x^k - \bar{x}\| &= \lim_{l \to \infty} \|x^k - x^{k+1}\| \leqslant \lim_{l \to \infty} \sum_{j=0}^{l-1} \|d^{k+j}\| \\
&\leqslant \|d_U^k\| \lim_{l \to \infty} \sum_{j=0}^{l-1} \left(\frac{1}{2}\right)^j = \|d_U^k\| \sum_{j=0}^{\infty} \left(\frac{1}{2}\right)^j = 2\|d_U^k\|.
\end{aligned}$$

Since this holds for an arbitrary (sufficiently large) $k \in \mathbb{N}$, we obtain the desired result by setting $\kappa_8 := 1/2$. □

Using Lemmas 3.9 and 3.10, we get the following local convergence result for the iterates x^k in exactly the same way as in the proof of the corresponding Theorem 2.11.

Theorem 3.11. *Let Assumption* 3.2 *be satisfied and* $\{x^k\}$ *be a sequence generated by Algorithm* 3.1 *with starting point* $x^0 \in B_r(x^*)$ *and limit point* $\bar{x}$. *Then the sequence* $\{x^k\}$ *converges locally quadratically to* $\bar{x}$.

Hence it turns out that the projected Levenberg–Marquardt method of Algorithm 3.1 has essentially the same local convergence properties as the constrained Levenberg–Marquardt method of Algorithm 2.1.

3.4. Globalized method

Although we are mainly interested in the local behaviour of the projected Levenberg–Marquardt method, we can globalize this method in a simple way by introducing a projected gradient step whenever the full projected Levenberg–Marquardt step does not provide a sufficient decrease for $\|F(x)\|$. The globalization strategy is therefore very similar to the one discussed in Section 2.4. Assuming that F is continuously differentiable, we may formally state the algorithm as follows.

Algorithm 3.12 (Projected Levenberg–Marquardt Method: Globalized Version).

(S.0) Choose $x^0 \in X$, $\mu > 0$, $\beta, \sigma, \gamma \in (0,1)$, and set $k := 0$.

(S.1) If $F(x^k) = 0$, STOP.

(S.2) Choose $H_k \in \mathbb{R}^{m \times n}$, set $\mu_k := \mu\|F(x^k)\|^2$, and compute d_U^k as the solution of (17).

(S.3) If

$$\|F(P_X(x^k + d_U^k))\| \le \gamma\|F(x^k)\|, \tag{25}$$

then set $x^{k+1} := P_X(x^k + d_U^k)$, $k \leftarrow k+1$, and go to (S.1); otherwise go to (S.4).

(S.4) Compute a stepsize $t_k = \max\{\beta^\ell \mid \ell = 0, 1, 2, \ldots\}$ such that

$$f(x^k(t_k)) \le f(x^k) + \sigma\nabla f(x^k)^{\mathrm{T}}(x^k(t_k) - x^k),$$

where $x^k(t) := P_X[x^k - t\nabla f(x^k)]$. Set $x^{k+1} := x^k(t_k)$, $k \leftarrow k+1$, and go to (S.1).

Algorithm 3.12 has the advantage of having simpler subproblems than Algorithm 2.12. However, this advantage is realized only if the projections onto the feasible set X can be computed in a convenient manner, which is particularly the case when X is described by some box constraints.

Based on the previous results, it is not difficult to see that the counterpart of Theorem 2.13 also holds for Algorithm 3.12. We skip the details here.

4. Numerical results

We have implemented Algorithm 3.12 in MATLAB and tested it on a number of examples from different areas. The implementation differs slightly from the description of Algorithm 3.12.

Specifically, Algorithm 3.12 considers two types of steps only, namely Levenberg–Marquardt and projected gradient steps, whereas our implementation uses the following three types of steps:

- LM-step (Levenberg–Marquardt step): This is used when the descent condition (25) is satisfied, i.e., (S.3) is carried out.
- LS-step (line search step): This step occurs if condition (25) is not satisfied but the search direction $s^k := P_X(x^k + d^k) - x^k$ is a descent direction for f in the sense that $\nabla f(x^k)^T s^k \leqslant -\rho \|s^k\|^p$ for some constants $\rho > 0$ and $p > 1$. We then use an Armijo-type line search to reduce f along the direction s^k.
- PG-step (projected gradient step): If neither an LM-step nor an LS-step can be used, we apply a projected gradient step as described in (S.4) of Algorithm 3.12.

It is easy to see that this modification does not change the local and global convergence properties of Algorithm 3.12.

The parameters used for our test runs are $\beta = 0.9$, $\sigma = 10^{-4}$, $\gamma = 0.99995$, $\rho = 10^{-8}$, $p = 2.1$. For the Levenberg–Marquardt parameter, we initially take $\mu_0 := \frac{1}{2} \cdot 10^{-4}\|F(x^0)\|^2$ and then use the update $\mu_{k+1} := \min\{\mu_k, \|F(x^{k+1})\|^2\}$, which is motivated by the local convergence analysis. Furthermore, we always take $H_k := F'(x^k)$ since all our test examples are smooth. The computation of the search direction d_U^k from the linear system (17) is done by a Cholesky factorization. Alternatively, (17) could be replaced by an equivalent linear least squares problem which then could be solved by suitable orthogonal transformations (Householder or Givens). Finally, we terminate the iteration if $\|F(x^k)\| \leqslant \varepsilon$ or $k \geqslant k_{\max}$ or $t_k \leqslant t_{\min}$ with $\varepsilon = 10^{-5}$, $k_{\max} = 100$ and $t_{\min} = 10^{-12}$. The computational results obtained with these parameters are shown in Tables 1–6.

Tables 1 and 2 give the results for some square systems of equations. All these systems have some bound constraints. For example, many of the test examples come from chemical equilibrium problems where the components of the vector x correspond to chemical concentrations, so that these problems have some nonnegativity constraints. Other examples are obtained from complementarity problems

$$G(x) - y = 0, \quad x \geqslant 0, \quad y \geqslant 0, \quad x_i y_i = 0 \ \forall i.$$

Also some constrained optimization problems are solved by applying the algorithm to the corresponding KKT conditions.

The starting point x^0 for all test examples is the vector of lower bounds except for those examples which arise from complementarity or optimization problems. For the latter problems we used the standard starting point from the literature (filled with zero Lagrange multipliers).

The columns of Table 1 contain the name of the test problem (together with a hint to the literature that, however, is usually not the original reference for that particular example), the dimension $n\,(=m)$ of this example, the number of iterations, the number of LM-, LS- and PG-steps, the number of function evaluations as well as the final value of the merit function f. Table 2 has a similar structure except that the first column gives the value of a parameter for the particular problem (we use all three different parameters given in [9]).

Table 3 states the results obtained for some underdetermined systems taken from [4]. The columns have a similar meaning to those of Table 1 except that we added one more column that gives the dimension m of the corresponding (nonsquare) system.

Table 1
Numerical results for different test problems (square systems)

Test problem, source	n	iter	LM/LS/PG	F-eval.	$f(x)$
Himmelblau function, [9, 14.1.1]	2	8	8/0/0	9	1.1e − 11
Equilibrium combustion, [9, 14.1.2]	5	10	6/4/0	11	5.2e − 11
Bullard–Biegler system, [9, 14.1.3]	2	11	9/2/0	40	9.5e − 15
Ferraris–Tronconi system, [9, 14.1.4]	2	3	3/0/0	[illegible]	8.9e − 15
Brown's almost lin. syst., [9, 14.1.5]	5	10	10/0/0	[illegible]	9.1e − 16
Robot kinematics system, [9, 14.1.6]	8	5	5/0/0	6	[illegible] − 19
Circuit design problem, [9, 14.1.7]	9	—	—/—/—	—	—
Chem. equil. system, [18, system 1]	11	15	13/1/1	64	6.5e − 11
Chem. equil. system, [18, system 2]	5	—	—/—/—	—	—
Combust. system (Lean case), [17]	10	7	5/2/0	[illegible]	2.0e − 11
Combust. system (Rich case), [17]	10	—	—/—/—	—	—
Kojima–Shindo problem, [7]	4	5	[illegible]	21	3.1e − 13
Josephy problem, [7]	4	11	[illegible]	80	9.5e − 21
Mathiesen problem, [7]	4	3	[illegible]	4	2.0e − 16
Hock–Schittkowski 34, [11]	16	8	7/1/0	32	7.6e − 18
Hock–Schittkowski 35, [11]	8	[illegible]	2/0/0	3	1.2e − 13
Hock–Schittkowski 66, [11]	16	[illegible]	[illegible]	253	3.4e − 11
Hock–Schittkowski 76, [11]	14	[illegible]	[illegible]	428	7.1e − 11

Table 2
Numerical results for test problem 14.1.9 from [9] (Smith steady-state temperature)

ΔH	n	iter	LM/LS/PG	F-eval.	$f(x)$
−50,000	1	[illegible]	3/0/0	4	2.8e − 15
−35,958	1	[illegible]	3/0/0	4	2.9e − 17
−35,510	1	3	3/0/0	4	2.3e − 17

Table 3
Numerical results for some underdetermined systems from [4]

Test problem, source	n	m	iter	LM/LS/PG	F-eval.	$f(x)$
Linear system, [4, Problem 2]	100	50	3	3/0/0	4	1.3e − 11
Linear system, [4, Problem 2]	200	100	6	6/0/0	7	1.8e − 14
Linear system, [4, Problem 2]	300	150	13	13/0/0	14	7.8e − 29
Quadratic system, [4, Problem 4]	100	50	11	11/0/0	12	1.2e − 11
Quadratic system, [4, Problem 4]	200	100	26	26/0/0	27	5.0e − 12
Quadratic system, [4, Problem 4]	300	150	72	72/0/0	73	2.6e − 15

Finally, Tables 4–6 contain numerical results for some parameter-dependent problems where the starting point of a problem is equal to the solution of the previous problem, i.e., we apply Algorithm 3.12 in the framework of a path-following method. Note, however, that the

Table 4
Numerical results for test problem 14.1.8 from [9] (CSTR)

R	n	iter	LM/LS/PG	F-eval.	$f(x)$
0.995	2	8	8/0/0	9	1.6e − 10
0.990	2	9	9/0/0	10	8.5e − 11
0.985	2	9	9/0/0	10	1.7e − 10
0.980	2	10	10/0/0	11	1.2e − 19
0.975	2	11	11/0/0	12	1.1e − 10
0.970	2	12	12/0/0	13	[illegible]
0.965	2	13	13/0/0	[illegible]	1.8e − 10
0.960	2	15	15/0/0	[illegible]	1.9e − 10
0.955	2	18	18/0/0	19	2.0e − 10
0.950	2	24	24/0/0	25	1.5e − 10
0.945	2	—	—/—/—	—	—
0.940	2	—	—/—/—	—	—
0.935	2	—	—/—/—	[illegible]	—

Table 5
Numerical results for Chandrasekhar H-equation, see [5]

c	n	iter	LM/LS/PG	F-eval.	$f(x)$
0.5	100	4	[illegible]	5	4.1e − 11
0.6	100	4	4/0/0	5	2.3e − 11
0.7	100	5	5/0/0	6	1.3e − 10
0.8	100	[illegible]	9/0/0	10	5.1e − 11
0.9	100	[illegible]	3/92/0	383	1.6e − 10
0.99	100	[illegible]	97/1/1	102	1.7e − 10

Table 6
Numerical results for chemical equilibrium problem (propane), see [6]

c	n	iter	LM/LS/PG	F-eval.	$f(x)$
3.0	[illegible]	14	14/0/0	15	1.0e − 10
3.1	[illegible]	11	7/2/2	177	1.6e − 10
3.2	10	2	2/0/0	3	6.4e − 13
3.3	10	2	2/0/0	3	3.0e − 15
3.4	10	2	2/0/0	3	1.1e − 15
3.5	10	2	2/0/0	3	2.9e − 15
3.6	10	2	2/0/0	3	1.4e − 15
3.7	10	2	2/0/0	3	2.2e − 15
3.8	10	2	2/0/0	3	1.9e − 15
3.9	10	2	2/0/0	3	2.3e − 15
4.0	10	2	2/0/0	3	2.8e − 15

dependence of these problems on the corresponding parameters might be nonsmooth, e.g., the number of (known) solutions in the example given in Table 4 varies significantly with the values of parameters.

Interestingly, our method is also able to solve the counterexample from (21) which does not satisfy the local error bound assumption (18) at the solution point $x^* := (-1, 0)$ (recall also that this example has a connected solution set, hence the above x^* is not locally unique). For example, taking starting points like $(-2, 0)$, $(-2, 1)$ or $(-3, 1)$, our method terminates with $x^* = (-1, 0)$ after one or two iterations only.

To summarize the results shown in the tables, we were able to solve most of the test problems without any difficulties. Only in a few cases, we were not able to find an approximate solution (the same is true for the method of [1], which has also been tested on many of the examples used here). This is typically due to the fact that the step size gets too small (except for the circuit design problem in Table 1, for which we observed convergence to a nonoptimal stationary point). For some examples, we also needed a relatively large number of function evaluations (at least compared to the number of iterations), but this is mainly due to the fact that the stepsize reduction factor β was chosen equal to 0.9 (both for LS- and PG-steps). Taking a smaller value of β typically reduces the number of function evaluations, but increases the number of iterations. For example, applying our method with $\beta = 0.5$ to the three problems `Josephy`, `Hock–Schittkowski 66` and `Hock–Schittkowski 76`, it takes 15, 70 and 57 iterations, respectively, but only 49, 136 and 161 function evaluations, cf. Table 1.

Of course, the behaviour of our method also depends on the choice of the Levenberg–Marquardt parameter. However, since we have to use updates of the form $\mu_k = O(\|F(x^k)\|^2)$ in order to be consistent with our theory, the definition of μ_k is somewhat restricted. In fact, the entire behaviour of our algorithm does not change much if we use modified updates of the form $\mu_{k+1} := \min\{\mu_k, \mu\|F(x^{k+1})\|^2\}$ for some constant $\mu > 0$, e.g., taking $\mu = 0.1$ does not change a single iteration number for any of the test examples from Table 1.

We close this section with some comments in order to compare our method with those from [1,13,22]. To this end, we first want to stress that these three methods can be applied to nonlinear systems of equations with box constraints only, whereas our method is much more general and allows convex constraints. Furthermore, our method can also be applied to nonsquare problems like those from Table 3. This is not possible for the methods developed in [1,13,22]. Furthermore, the local convergence analysis for all methods from [1,13,22] is based on a nonsingularity assumption which implies that the solution is locally unique. The methods from [1,22] also have to solve more complicated subproblems (trust region subproblems, quadratic programs), although the implementation of the method from [1] and described in more detail in [2] is based on a dogleg-type strategy and therefore solves only one linear system of equations per iteration like our method or the one from [13].

On the other hand, the main focus of this paper is on the local convergence behaviour, and the globalization has been included only for the sake of completeness. While the globalization in [13] is very similar, the methods from [1,22] use more sophisticated globalization strategies and therefore seem to have a slightly better numerical behaviour, at least if their local assumptions are satisfied. For example, the method from [1] was able to solve the `Chemical equilibrium system (System 2)`, whereas the method from [13] produced an additional error on the `Bullard–Biegler system`.

5. Final remarks

This paper has described two Levenberg–Marquardt-type methods for the solution of a constrained system of equations. Both methods have been shown to possess a local quadratic rate of convergence under a suitable error bound condition. This property is motivated by the recent research for unconstrained equations in [24] and seems to be much stronger than that of any other method for constrained equations known to the authors.

The globalization strategy used in this paper is quite standard and can certainly be improved, although the numerical results indicate that the method works quite well with this strategy. However, numerical experiments were carried out for the case of box constraints only since otherwise the computation of the projections onto the feasible set becomes very expensive and, in fact, dominates the overall cost of the algorithm. The question of how to deal with a general convex set X in a numerically efficient way deserves further study.

Acknowledgements

The authors would like to thank Stefania Bellavia for sending us some of the test problems.

References

[1] S. Bellavia, M. Macconi, B. Morini, An affine scaling trust-region approach to bound-constrained nonlinear systems, Appl. Numer. Math. 44 (2003) 257–280.
[2] S. Bellavia, M. Macconi, B. Morini, STRSCNE: a scaled trust-region solver for constrained nonlinear systems, Comput. Optim. Appl., to appear.
[3] D.P. Bertsekas, Nonlinear Programming, Athena Scientific, Massachusetts, 1995.
[4] H. Dan, N. Yamashita, M. Fukushima, Convergence properties of the inexact Levenberg–Marquardt method under local error bound conditions, Optim. Meth. Software 17 (2002) 605–626.
[5] J.E. Dennis, R.B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Prentice-Hall, Englewood Cliffs, NJ, 1983.
[6] P. Deuflhard, A. Hohmann, Numerische Mathematik, Walter de Gruyter, Berlin, Germany, 1991.
[7] S.P. Dirkse, M.C. Ferris, MCPLIB: a collection of nonlinear mixed complementarity problems, Optim. Meth. Software 5 (1995) 319–345.
[8] J.Y. Fan, Y.X. Yuan, On the convergence of a new Levenberg–Marquardt method, Technical Report, AMSS, Chinese Academy of Sciences, Beijing, China, 2001.
[9] C.A. Floudas, P.M. Pardalos, C.S. Adjiman, W.R. Esposito, Z.H. Gumus, S.T. Harding, J.L. Klepeis, C.A. Meyer, C.A. Schweiger, Handbook of Test Problems in Local and Global Optimization, Nonconvex Optimization and its Applications, Vol. 33, Kluwer Academic Publishers, The Netherlands, 1999.
[10] 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.
[11] W. Hock, K. Schittkowski, Test Examples for Nonlinear Programming Codes, Lecture Notes in Economics and Mathematical Systems, Vol. 187, Springer, New York, 1981.
[12] A.J. Hoffman, On approximate solutions of systems of linear inequalities, J. Nat. Bureau Standards 49 (1952) 263–265.
[13] C. Kanzow, An active set-type Newton method for constrained nonlinear systems, in: M.C. Ferris, O.L. Mangasarian, J.-S. Pang (Eds.), Complementarity: Applications, Algorithms and Extensions, Kluwer Academic Publishers, The Netherlands, 2001, pp. 179–200.

[14] C. Kanzow, Strictly feasible equation-based methods for mixed complementarity problems, Numer. Math. 89 (2001) 135–160.
[15] C.T. Kelley, Iterative Methods for Linear and Nonlinear Equations, SIAM, Philadelphia, 1995.
[16] D.N. Kozakevich, J.M. Martinez, S.A. Santos, Solving nonlinear systems of equations with simple bounds, Comput. Appl. Math. 16 (1997) 215–235.
[17] K. Meintjes, A.P. Morgan, A methodology for solving chemical equilibrium systems, Appl. Math. Comput. 22 (1987) 333–361.
[18] K. Meintjes, A.P. Morgan, Chemical equilibrium systems as numerical test problems, ACM Trans. Math. Software 16 (1990) 143–151.
[19] R.D.C. Monteiro, J.-S. Pang, A potential reduction Newton method for constrained equations, SIAM J. Optim. 9 (1999) 729–754.
[20] J.M. Ortega, W.C. Rheinboldt, Iterative Solution of Nonlinear Equations in Several Variables, Academic Press, New York, 1970.
[21] L. Qi, X.-J. Tong, D.-H. Li, An active-set projected trust region algorithm for box constrained nonsmooth equations, J. Optim. Theory Appl., to appear.
[22] M. Ulbrich, Nonmonotone trust-region method for bound-constrained semismooth equations with applications to nonlinear mixed complementarity problems, SIAM J. Optim. 11 (2001) 889–917.
[23] T. Wang, R.D.C. Monteiro, J.-S. Pang, An interior point potential reduction method for constrained equations, Math. Programming 74 (1996) 159–195.
[24] N. Yamashita, M. Fukushima, On the rate of convergence of the Levenberg–Marquardt method, Computing 15 (Suppl) (2001) 239–249.