

Article

# On a Nonsmooth Gauss–Newton Algorithms for Solving Nonlinear Complementarity Problems

Marek J. Śmietański 

Faculty of Mathematics and Computer Science, University of Lodz, Banacha 22, 90-238 Łódź, Poland; marek.smietanski@wmii.uni.lodz.pl

Received: 25 June 2020; Accepted: 31 July 2020; Published: 4 August 2020



**Abstract:** In this paper, we propose a new version of the generalized damped Gauss–Newton method for solving nonlinear complementarity problems based on the transformation to the nonsmooth equation, which is equivalent to some unconstrained optimization problem. The B-differential plays the role of the derivative. We present two types of algorithms (usual and inexact), which have superlinear and global convergence for semismooth cases. These results can be applied to efficiently find all solutions of the nonlinear complementarity problems under some mild assumptions. The results of the numerical tests are attached as a complement of the theoretical considerations.

**Keywords:** Gauss–Newton method; nonsmooth equations; nonsmooth optimization; nonlinear complementarity problem; B-differential; superlinear convergence; global convergence

## 1. Introduction

Let  $F : R^n \rightarrow R^n$  and let  $F_i, i = 1, \dots, n$  denote the components of  $F$ . The nonlinear complementarity problem (NCP) is to find  $x \in R^n$  such that

$$x \geq 0, F(x) \geq 0 \text{ and } x^T F(x) = 0. \quad (1)$$

The  $i$ th component of a vector  $x$  is represented by  $x_i$ . Solving (1) is equivalent to solving a nonlinear equation  $G(x) = 0$ , where the operator  $G : R^n \rightarrow R^n$  is defined by

$$G(x) = \begin{bmatrix} \varphi(x_1, F_1(x)) \\ \dots \\ \varphi(x_n, F_n(x)) \end{bmatrix}$$

with some special function  $\varphi$ . Function  $\varphi$  may have one of the following forms:

$$\begin{aligned} \varphi_1(a, b) &= \min\{a, b\}; \\ \varphi_2(a, b) &= \sqrt{a^2 + b^2} - a - b; \\ \varphi_3(a, b) &= \theta(|a - b|) - \theta(a) - \theta(b), \end{aligned}$$

where  $\theta : R \rightarrow R$  is any strictly increasing function with  $\theta(0) = 0$ , see [1].

The (NCP) problem is one of the fundamental problems of mathematical programming, operations research, economic equilibrium models, and in engineering sciences. A lot of interesting and important applications can be found in the papers of Harker and Pang [2] and Ferris and Pang [3]. We can find the most essential applications in:

- engineering—optimal control problems, contact or structural mechanics problems, structural design problems, or traffic equilibrium problems,
- equilibrium modeling—general equilibrium (in production or consumption), invariant capital stock, or game-theoretic models.

We borrow a technique used in solving some smooth problems. If  $g$  is a merit function of  $G$ , i.e.,  $g(x) = \frac{1}{2}G(x)^T G(x)$ , then any stationary point of  $g(x)$  is a least-squares solution of the equation  $G(x) = 0$ . Then, algorithms for minimization are equivalent to algorithms for solving equations. The usual Gauss–Newton method (known also as the differential corrections method), presented by Ortega and Rheinboldt [4] in the smooth case, has the form

$$x^{(k+1)} = x^{(k)} - \left[ G'(x^{(k)})^T G'(x^{(k)}) \right]^{-1} G'(x^{(k)})^T G(x^{(k)}). \tag{2}$$

Local convergence properties of the Gauss–Newton method was discussed by Chen and Li [5], but only for some smooth case. The Levenberg–Marquardt method is also considered, which is a modified Gauss–Newton method, in some papers, e.g., [6] or [7]. Moreover, some comparison of semismooth algorithms for solving (NCP) problems has been made in [8].

In practice, we may also consider the damped Gauss–Newton method

$$x^{(k+1)} = x^{(k)} - \omega_k \left[ G'(x^{(k)})^T G'(x^{(k)}) + \lambda_k I \right]^{-1} G'(x^{(k)})^T G(x^{(k)}) \tag{3}$$

with parameters  $\omega_k$  and  $\lambda_k$ . Parameter  $\omega_k$  may be chosen to ensure suitable decrease of  $g$ . If  $\lambda_k$  is positive for all  $k$ , then the inverse matrix in (3) always exists because  $G'(x^{(k)})^T G'(x^{(k)})$  is a symmetric and positive semidefinite matrix. The method (3) has the important advantage: the search direction always exists, even if  $G'(x)$  is singular. Naturally, in the case of nonsmooth equations, some additional assumptions are needed to allow the use of some line search strategies and to ensure the global convergence. Because, in some cases, a function  $G$  is nondifferentiable, so the equation  $G(x) = 0$  will be nonsmooth, whereby the method (3) may be useless. Some version of the Gauss–Newton method for solving complementarity problems was also introduced by Xiu and Zhang [9] for generalized problems, but only for linear ones. Thus, for solving nonsmooth and nonlinear problems, we propose two new versions of a damped Gauss–Newton algorithm based on B-differential. The usual generalized method is a relevant extension of the work by Subramanian and Xiu [10] for a nonsmooth case. In turn, an inexact version is related to the traditional approach, which was widely studied, e.g., in [11]. In recent years, various versions of the Gauss–Newton method were discussed, although most frequently for solving nonlinear least-squares problems, e.g., in [12,13].

The paper is organized as follows: in the next section, we review some notions needed, such as B-differential, BD-regularity, semismoothness, etc. (Section 2.1). Next, we propose a new optimization problem-based methods for the NCP, transforming the NCP into an unconstrained minimization problem by employing a function  $\varphi_3$  (Section 2.2). We state its global convergence and superlinear convergence rate under appropriate conditions. In Section 3, we present the results of numerical tests.

## 2. Materials and Methods

### 2.1. Preliminaries

If  $F$  is Lipschitz continuous, the Rademacher’s theorem [14] implies that  $F$  is almost everywhere differentiable. Let the set of points, where  $F$  is differentiable, be denoted by  $D_F$ . Then, the B-differential (the Bouligand differential) of  $F$  at  $x$  (introduced in [15]) is

$$\partial_B F(x) = \left\{ \lim_{x^{(n)} \rightarrow x} F'(x^{(n)}), x^{(n)} \in D_F \right\},$$

where  $F'(x)$  denotes the usual Jacobian of  $F$  at  $x$ . The generalized Jacobian of  $F$  at  $x$  in the sense of Clarke [14] is

$$\partial F(x) = \text{conv} \partial_B F(x)$$

We say that  $F$  is BD-regular at  $x$ , if  $F$  is locally Lipschitz at  $x$  and if all  $V \in \partial_B F(x)$  are nonsingular (regularity on account of B-differential). Qi proved (Lemma 2.6, [15]) that, if  $F$  is BD-regular at  $x$ , then a neighborhood  $N$  of  $x$  and a constant  $C > 0$  exist such that, for any  $y \in N$  and  $V \in \partial_B F(y)$ ,  $V$  is nonsingular and

$$\|V^{-1}\| \leq C.$$

Throughout this paper,  $\|\cdot\|$  denotes the 2-norm.

The notion of semismoothness was originally introduced for functionals by Mifflin [16]. The following definition is taken from Qi and Sun [17]. A function  $F$  is semismooth at a point  $x$ , if  $F$  is locally Lipschitzian at  $x$  and

$$\lim_{V \in \partial F(x+th'), h' \rightarrow h, t \downarrow 0} Vh'$$

exists for any  $h \in R^n$ .  $F$  is also said semismooth at  $x$ , if it is directionally differentiable at  $x$  and

$$Vh - F'(x, h) = o(\|h\|).$$

Scalar products and sums of semismooth functions are still semismooth functions. Piecewise smooth functions and maximum of a finite number of smooth functions are also semismooth. The semismoothness is the almost usually seen assumption on  $F$  in papers dealing with nonsmooth equations because it implies some important properties for convergence analysis of methods in nonsmooth optimization.

If for any  $V \in \partial F(x + h)$ , as  $h \rightarrow 0$

$$Vh - F'(x, h) = O(\|h\|^{1+p}),$$

where  $0 < p \leq 1$ , then we say  $F$  is  $p$ -order semismooth at  $x$ . Clearly,  $p$ -order semismoothness implies semismoothness. If  $p = 1$ , then the function  $F$  is called strongly semismooth. Piecewise  $C^2$  functions are examples of strongly semismooth functions.

Qi and Sun [17] remarked that, if  $F$  is semismooth at  $x$ , then, for any  $h \rightarrow 0$

$$F(x + h) - F(x) - F'(x; h) = o(\|h\|),$$

and, if  $F$  is  $p$ -order semismooth at  $x$ , then for any  $h \rightarrow 0$

$$F(x + h) - F(x) - F'(x; h) = O(\|h\|^{1+p}).$$

**Remark 1.** Strong semismoothness of the appropriate function usually implies quadratic convergence of method instead of the superlinear one for semismooth function.

In turn, Pang and Qi [18] proved that semismoothness of  $F$  at  $x$  implies that

$$\sup_{V \in \partial F(x+h)} \{F(x + h) - F(x) - Vh\} = o(\|h\|).$$

Moreover, if  $F$  is  $p$ -order semismooth at  $x$ , then

$$\sup_{V \in \partial F(x+h)} \{F(x+h) - F(x) - Vh\} = O(\|h\|^{1+p}).$$

2.2. The Algorithm and Its Convergence

Consider nonlinear equation  $G(x) = 0$  defined by  $\varphi_3$ . The equivalence of solving this equation and problem (NCP) is described by the following theorem:

**Theorem 1** (Mangasarian [1]). *Let  $\theta$  be any strictly increasing function from  $R$  into  $R$ , that is,  $a > b \Leftrightarrow \theta(a) > \theta(b)$ , and let  $\theta(0) = 0$ . Then,  $x$  solves the complementarity problem (1) if and only if*

$$\theta(|F_i(x) - x_i|) - \theta(F_i(x)) - \theta(x_i) = 0, \quad i = 1, 2, \dots, n. \tag{4}$$

For the convenience, denote

$$G_i(x) := \theta(|F_i(x) - x_i|) - \theta(F_i(x)) - \theta(x_i) \tag{5}$$

for  $i = 1, 2, \dots, n$ .

We assume that the function  $\theta$  in Theorem 1 has the form

$$\theta(\xi) = \xi |\xi|.$$

Let  $G(x)$  be the associated function. We define function  $g$  in the following way:

$$g(x) = \frac{1}{2} \|G(x)\|^2,$$

which allows for solving system  $G(x) = 0$  based on solving the nonlinear least-square problem

$$\min_x g(x). \tag{6}$$

Let us note that  $x^*$  solves  $G(x) = 0$  if and only if it is a stationary point of  $g$ . Thus, from Theorem 1,  $x^*$  solves (1).

**Remark 2.** *On the other hand, the first-order optimality conditions for problem (6) are equivalent to the nonlinear system*

$$\nabla g(x) = G'(x)^T G(x) = 0,$$

where  $\nabla g$  is the gradient of  $g$ , provided  $G$  is differentiable and  $G'$  is the Jacobian matrix of  $G$ .

The continuous differentiability of the merit function  $g$  for some kind of nonsmooth functions was established by Ulbrich in the following lemma:

**Lemma 1** (Ullbrich, [19]). *Assume that the function  $G : R^n \supset D \rightarrow R^n$  is semismooth, or, stronger,  $p$ -order semismooth,  $0 < p \leq 1$ , then the merit function  $\frac{1}{2} \|G(x)\|^2$  is continuously differentiable on  $D$  with gradient  $\nabla g(x) = V^T G(x)$ , where  $V \in \partial G(x)$  is arbitrary.*

**Lemma 2.** *For any  $x \in R^n$ , let  $A_x = V_x^T V_x$ , where  $V_x \in \partial_B G(x)$ . Suppose that  $\nabla g(x) \neq 0$ . Then, given  $\lambda > 0$ , the direction  $d$  given by*

$$(A_x + \lambda I)d = \nabla g(x)$$

*is an ascent direction for  $g$ . In particular, there is a positive  $w$  such that  $g(x - wd) < g(x)$ .*

**Proof.** There exist constants  $\beta \geq 0$  and  $\gamma > 0$  such that

$$\beta \|\mathbf{h}\|^2 \leq \mathbf{h}^T \mathbf{A}_x \mathbf{h} \leq \gamma \|\mathbf{h}\|^2 \text{ for all } \mathbf{h} \in \mathbb{R}^n,$$

because  $\mathbf{A}_x$  defined as  $\mathbf{V}_x^T \mathbf{V}_x$  is symmetric and positive semidefinite.

It follows that

$$(\beta + \lambda) \|\mathbf{h}\|^2 \leq \mathbf{h}^T (\mathbf{A}_x + \lambda \mathbf{I}) \mathbf{h} \leq (\gamma + \lambda) \|\mathbf{h}\|^2 \text{ for all } \mathbf{h} \in \mathbb{R}^n.$$

Since  $\nabla g(\mathbf{x}) \neq 0$ ,  $\mathbf{d} \neq 0$ . If we take  $\mathbf{h} = \mathbf{d}$ , we obtain

$$\mathbf{d}^T \nabla g(\mathbf{x}) \geq (\beta + \lambda) \|\mathbf{d}\|^2 > 0.$$

It follows that  $\nabla g(\mathbf{x}) \mathbf{d} > 0$  and that  $\mathbf{d}$  is a ascent direction for  $g$  (Section 8.2.1 in [4]).  $\square$

Now, we present the generalized version of the damped Gauss–Newton method for solving the nonlinear complementarity problem.

---

**Algorithm 1:** The damped Gauss–Newton method for solving NCP

---

Let  $\beta, \delta \in (0, 1)$  be given. Let  $\mathbf{x}^{(0)}$  be a starting point. Given  $\mathbf{x}^{(k)}$ , the steps for obtaining  $\mathbf{x}^{(k+1)}$  are:

Step 1: If  $\nabla g(\mathbf{x}^{(k)}) = 0$ , then stop. Otherwise, choose any matrix  $\mathbf{V}_k \in \partial_B G(\mathbf{x}^{(k)})$  and let

$$\mathbf{A}_k = \mathbf{V}_k^T \mathbf{V}_k.$$

Step 2: Let  $\lambda_k = g(\mathbf{x}^{(k)})$ .

Step 3: Find  $\mathbf{d}^{(k)}$  that is a solution of the linear system

$$(\mathbf{A}_k + \lambda_k \mathbf{I}) \mathbf{d}^{(k)} = \nabla g(\mathbf{x}^{(k)}).$$

Step 4: Compute the smallest nonnegative integer  $m_k$  such that

$$g(\mathbf{x}^{(k)} + \beta^m \mathbf{d}^{(k)}) - g(\mathbf{x}^{(k)}) \leq -\delta \beta^m \nabla g(\mathbf{x}^{(k)})^T \mathbf{d}^{(k)}$$

and set

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \beta^{m_k} \mathbf{d}^{(k)}.$$


---

**Remark 3.** (i) In Step 2, letting  $\lambda_k = g(\mathbf{x}^{(k)})$  is one of the simplest strategy because then  $\{\lambda_k\}$  converges to 0. (ii) The line search step (Step 4) in the algorithm follows the Armijo rule.

**Theorem 2.** Let  $\mathbf{x}^{(0)}$  be a starting point and  $\{\mathbf{x}^{(k)}\}$  be a sequence generated by Algorithm 1. Assume that:

(a)  $\sup_k \|\mathbf{V}_k\| < \infty$  for all  $\mathbf{V}_k \in \partial_B G(\mathbf{x}^{(k)})$ ;

(b)  $\nabla g(\mathbf{x})$  is Lipschitzian with a constant  $L_g > 0$  on the level set  $L = \{\mathbf{x} : g(\mathbf{x}) \leq g(\mathbf{x}^{(0)})\}$ .

Then, the generalized damped Gauss–Newton method described by Algorithm 1 is well defined and either  $\{\mathbf{x}^{(k)}\}$  terminates at a stationary point of  $g$ , or else every accumulation point of  $\{\mathbf{x}^{(k)}\}$ , if it exists, is a stationary point of  $g$ .

**Proof.** The proof is almost the same as Theorem 2.1 in [10], providing appropriately modified assumptions.  $\square$

For the nonsmooth case, the alternative condition may be considered instead of Lipschitz continuity of  $\nabla g(\mathbf{x})$  (similar as in [10]). Thus, we have the following convergence theorem:

**Theorem 3.** Let  $\mathbf{x}^{(0)}$  be a starting point and  $\{\mathbf{x}^{(k)}\}$  be a sequence generated by Algorithm 1. Assume that:

- (a) the level set  $L = \{\mathbf{x} : g(\mathbf{x}) \leq g(\mathbf{x}^{(0)})\}$  is bounded;
- (b)  $G$  is semismooth on  $L$ .

Then, the generalized damped Gauss–Newton method described by Algorithm 1 is well defined and either  $\{\mathbf{x}^{(k)}\}$  terminates at a stationary point of  $g$ , or else every accumulation point of  $\{\mathbf{x}^{(k)}\}$ , if it exists, is a stationary point of  $g$ .

Now, we take up the rate of convergence of the considered algorithm. The following theorem shows suitable conditions in various cases.

**Theorem 4.** Suppose that  $\mathbf{x}^*$  is a solution of problem (1),  $G$  is semismooth, and  $G$  is BD-regular at  $\mathbf{x}^*$ . Then, there exists a neighborhood  $N_*$  of  $\mathbf{x}^*$  such that, if  $\mathbf{x}^{(0)} \in N_*$  and the sequence  $\{\mathbf{x}^{(k)}\}$  is generated by Algorithm 1, we have:

- (i)  $\mathbf{x}^{(k)} \in N_*$  for all  $k$  and the sequence  $\{\mathbf{x}^{(k)}\}$  is linear convergent to  $\mathbf{x}^*$ ;
- (ii) if  $\delta < 0.5$ , then the convergence is at least superlinear;
- (iii) If  $G$  is strongly semismooth, then the convergence is quadratic.

**Proof.** The proof of similar theorem given by Subramanian and Xiu [10] is based on three lemmas, which have the same assumptions as theorem. Now, we present these lemmas in versions adapted to our nonsmooth case. □

**Lemma 3.** Assume that  $\mathbf{d}_x$  is a solution of the equation

$$(\mathbf{A}_x + \lambda_x \mathbf{I})\mathbf{d}_x = \nabla g(\mathbf{x}),$$

where

$$\lambda_x = g(\mathbf{x}) \text{ and } \mathbf{A}_x = \mathbf{V}_x^T \mathbf{V}_x$$

for some matrix  $\mathbf{V}_x$  taken from  $\partial_B G(\mathbf{x})$ . Then, there is a neighborhood  $D_1$  of  $\mathbf{x}^*$  such that, for all  $\mathbf{x} \in D_1$ ,

$$\|\mathbf{x} - \mathbf{d}_x - \mathbf{x}^*\| = o(\|\mathbf{x} - \mathbf{x}^*\|).$$

**Lemma 4.** There is a neighborhood  $D_2$  of  $\mathbf{x}^*$  such that, for all  $\mathbf{x} \in D_2$ ,

- (a)  $g(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T \mathbf{A}_*(\mathbf{x} - \mathbf{x}^*) + o(\|\mathbf{x} - \mathbf{x}^*\|^2)$ ,
- (b)  $g(\mathbf{x}) = \frac{1}{2}(\mathbf{x} - \mathbf{x}^*)^T \mathbf{A}_x(\mathbf{x} - \mathbf{x}^*) + o(\|\mathbf{x} - \mathbf{x}^*\|^2)$ .

**Lemma 5.** Suppose that the conditions of Lemma 1 hold. Then, there is a neighborhood  $D_3$  of  $\mathbf{x}^*$  such that, for all  $\mathbf{x} \in D_3$ ,

$$g(\mathbf{x} - \mathbf{d}_x) - g(\mathbf{x}) + \frac{1}{2} \nabla g(\mathbf{x})^T \mathbf{d}_x \leq o(\|\mathbf{x} - \mathbf{x}^*\|^2).$$

The proofs of Lemmas 5 and 6 are almost the same as in [10]; however, in the proof of Lemma 4, we have to take into account the semismoothness and to use Lemma 1 to obtain the desired result.

At the same time, in a similar way, we may show a suitable rate of convergence.

Now, we consider the inexact version of the considered method, which computes an approximate step, using the nonnegative sequence of forcing terms to control the level of accuracy.

For the above inexact version of the algorithm, we can state the analogous theorems which are equivalents of Theorems 2–4. Based on our previous results, the proof can be carried out almost in the

same way as that of theorems for the 'exact' version of the method. However, the condition (7), implied by inexactness given in Step 3 of Algorithm 2, has to be considered. Thus, we omit both theorems as proofs here.

---

**Algorithm 2:** The inexact version of the damped Gauss-Newton method for solving NCP

---

Let  $\beta, \delta, \theta \in (0, 1)$  and  $\eta_k \in [0, 1]$  for all  $k$  given. Let  $\mathbf{x}^{(0)} \in R^n$  be a starting point. Given  $\mathbf{x}^{(k)}$ , the steps for obtaining  $\mathbf{x}^{(k+1)}$  are:

Step 1: If  $\nabla g(\mathbf{x}^{(k)}) = 0$ , then stop. Otherwise, choose any matrix  $\mathbf{V}_k \in \partial_B G(\mathbf{x}^{(k)})$  and let  $\mathbf{A}_k = \mathbf{V}_k^T \mathbf{V}_k$ .

Step 2: Let  $\lambda_k = g(\mathbf{x}^{(k)})$ .

Step 3: Find  $\mathbf{d}^{(k)}$  that is a solution of the linear system

$$\|(\mathbf{A}_k + \lambda_k \mathbf{I})\mathbf{d}^{(k)} + \nabla g(\mathbf{x}^{(k)})\| \leq \eta_k \|\nabla g(\mathbf{x}^{(k)})\|. \tag{7}$$

Step 4: If

$$\|\nabla g(\mathbf{x}^{(k)} + \mathbf{d}^{(k)})\| \leq \theta \|\nabla g(\mathbf{x}^{(k)})\|$$

then let

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \mathbf{d}^{(k)}$$

and go to Step 1.

Step 5: Compute the smallest nonnegative integer  $m_k$  such that

$$g(\mathbf{x}^{(k)} + \beta^m \mathbf{d}^{(k)}) - g(\mathbf{x}^{(k)}) \leq -\delta \beta^m \nabla g(\mathbf{x}^{(k)})^T \mathbf{d}^{(k)}$$

and set

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} + \beta^{m_k} \mathbf{d}^{(k)}$$

and go to Step 1.

---

### 3. Numerical Results

In this section, we present results of our numerical experiments, obtained by coding both algorithms in Code:Blocks. We use double precision on an Intel Core i7 3.2 GHz running under the Windows Server 2016 operating system. We applied the generalized damped Gauss–Newton method to solve three nonlinear complementarity problems. In the following examples:  $N_1$  and  $N_2$  denote the number of performed iterations to satisfy the stopping criterion  $\|\mathbf{x}^{(k+1)} - \mathbf{x}^{(k)}\| < 10^{-7}$ , using Algorithms 1 and 2, respectively. The forcing terms in Algorithm 2 were chosen as follows:  $\eta_k = (10k)^{-1}$  for all  $k$ .

**Example 1** (from Kojima and Shindo [20]). Let the function  $F : R^4 \rightarrow R^4$  have the form

$$\begin{aligned} F^1(\mathbf{x}) &= 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6, \\ F^2(\mathbf{x}) &= 2x_1^2 + x_1 + x_2^2 + 10x_3 + 2x_4 - 2, \\ F^3(\mathbf{x}) &= 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 9x_4 - 9, \\ F^4(\mathbf{x}) &= x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3. \end{aligned}$$

Problem (NCP) with the above function  $F$  has two solutions:

$$\mathbf{x}^* = (1, 0, 3, 0)^T \text{ and } \mathbf{x}^{**} = (\sqrt{6}/2, 0, 0, 0.5)^T$$

for which

$$F(\mathbf{x}^*) = (0, 31, 0, 4)^T \text{ and } F(\mathbf{x}^{**}) = \left(0, 2 + \frac{\sqrt{6}}{2}, 0, 0\right)^T.$$

Thus,  $\mathbf{x}^*$  is a non-degenerate solution of (NCP) because

$$L := \left\{i : x_i^* = 0, F^i(\mathbf{x}^*) = 0\right\} = \emptyset,$$

but  $\mathbf{x}^{**}$  is a degenerate solution.

Depending upon the starting point, we obtained the convergence iteration process to both solutions (see Table 1 or Figure 1).

Table 1. Results for Example 1.

$\mathbf{x}^{(0)}$	$N_1$	$N_2$	Solution
$(1, 0, 0, 0)^T$	9	11	$\mathbf{x}^{**}$
$(0, 0, 1, 0)^T$	failed	18	$\mathbf{x}^{**}$
$(0, 0, 0, 1)^T$	failed	failed	-
$(1, 0, 1, 0)^T$	7	10	$\mathbf{x}^*$
$(1, 0, 0, 1)^T$	7	9	$\mathbf{x}^{**}$
$(1, 0, 1, -5)^T$	6	8	$\mathbf{x}^{**}$

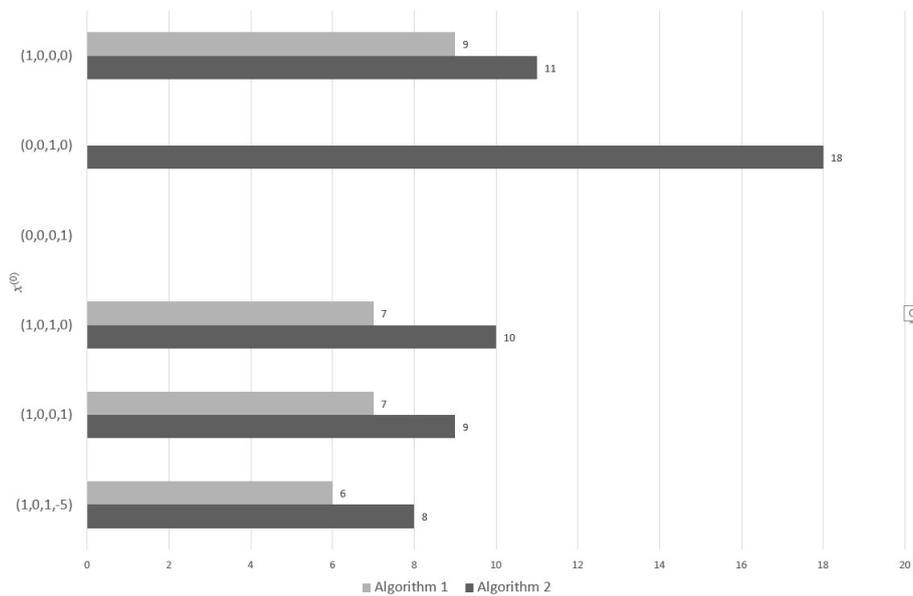


Figure 1. Number of iterations for various starting points (for Example 1).

Example 2. Let function  $F : R^2 \rightarrow R^2$  be defined as follows:

$$F(\mathbf{x}) = \begin{bmatrix} 2x_1 + x_2^2 - 6 \\ -x_1^2 + 4x_1 + \frac{1}{2}x_2 - 3 \end{bmatrix}.$$

Then, problem (NCP) has two solutions:

- non-degenerate

$$\mathbf{x}^* = (0,6)^T \text{ for which } F(\mathbf{x}^*) = (30,0)^T$$

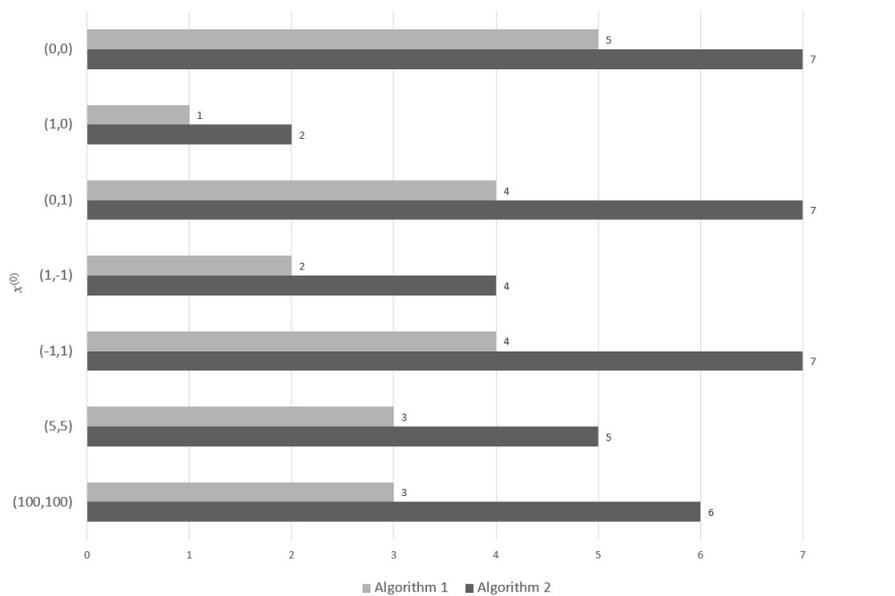
- degenerate

$$\mathbf{x}^{**} = (3,0)^T \text{ for which } F(\mathbf{x}^{**}) = (0,0)^T.$$

Similar to Example 1, we obtained the convergence iteration process for both solutions, depending on the starting point (see Table 2 or Figure 2).

**Table 2.** Results for Example 2.

$\mathbf{x}^{(0)}$	$N_1$	$N_2$	Solution
$(0,0)^T$	5	7	$\mathbf{x}^*$
$(1,0)^T$	1	2	$\mathbf{x}^{**}$
$(0,1)^T$	4	7	$\mathbf{x}^*$
$(1,-1)^T$	2	4	$\mathbf{x}^{**}$
$(-1,1)^T$	4	7	$\mathbf{x}^{**}$
$(5,5)^T$	3	5	$\mathbf{x}^*$
$(100,100)^T$	3	6	$\mathbf{x}^*$



**Figure 2.** Number of iterations for various starting points (for Example 2.)

**Example 3** (from Jiang and Qi [21]). Let function  $F : R^n \rightarrow R^n$  has the form  $F(\mathbf{x}) = M\mathbf{x} + \mathbf{q}$ , where

$$M = \begin{bmatrix} 4 & -1 & 0 & \dots & 0 & 0 \\ -1 & 4 & -1 & \dots & 0 & 0 \\ 0 & -1 & 4 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 4 & -1 \\ 0 & 0 & 0 & \dots & -1 & 4 \end{bmatrix}, \mathbf{q} = (-1, \dots, -1)^T.$$

Because  $F$  is strictly monotonic, the proper problem (NCP) has exactly one solution.

Calculations have been made for various  $n$  with one starting point  $\mathbf{x}^{(0)} = (0, \dots, 0)^T$ . For all tests, we obtain the same number of iterations  $N_1 = 3$  and  $N_2 = 4$ .

#### 4. Conclusions

We have given the nonsmooth version of the damped generalized Gauss–Newton method presented by Subramanian and Xu [10]. The generalized Newton algorithms related to the Gauss–Newton method are well-known important tools for solving nonsmooth equations, which arise from various nonlinear problems such as nonlinear complementarity or variational inequality. These algorithms are especially useful when the problem has many variables. We have proved that the sequences generated by the methods are superlinearly convergent under mild assumptions. Clearly, the semismoothness and BD-regularity are sufficient to obtain only a superlinear convergence of methods, while strong semismoothness even gives quadratic convergence. However, if function  $G$  is not semismooth or BD-regular or the gradient of  $g$  is not Lipschitzian, the Gauss–Newton methods may be useless.

The performance of both methods was evaluated in terms of the number of iterations required. The analysis of the numerical results seems to indicate that the methods are usually reliable for solving semismooth problems. The results show that the inexact approach can produce a noticeable slowdown by the number of iterations (compare  $N_1$  and  $N_2$  in Figures 1 and 2). In turn, an important advantage is that the algorithms allow us to find various solutions to the problem (this can be observed in two examples: the first and second one). However, if there are many solutions of the problem, then the relationship between the starting point and the obtained solution may be unpredictable.

Clearly, traditional numerical algorithms aren't the only method for solving the nonlinear complementarity problems, regardless of the degree of nonsmoothness. Except for the methods presented in the paper and mentioned in the Introduction, some computational intelligence algorithms can be used to solve (NCP) problems, i.e., monarch butterfly optimization (see [22,23]), the earthworm optimization algorithm (see [24]), the elephant herding optimization (see [25,26]), or the moth search algorithm (see [27,28]). All of these approaches are bio-inspired metaheuristic algorithms.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The author declares no conflict of interest.

#### References

1. Mangasarian, O.L. Equivalence of the complementarity problem to a system of nonlinear equations. *SIAM J. Appl. Math.* **1976**, *31*, 89–92. [CrossRef]
2. Harker, P.T.; Pang, J.S. Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Math. Program.* **1990**, *48*, 161–220. [CrossRef]
3. Ferris, M.C.; Pang, J.S. Engineering and economic applications of complementarity problems. *SIAM Rev.* **1997**, *39*, 669–713. [CrossRef]
4. Ortega, J.M.; Rheinboldt, W.C. *Iterative Solution of Nonlinear Equations in Several Variables*; Academic Press: New York, NY, USA, 1970.
5. Chen, J.; Li, W. Local convergence of Gauss–Newton's like method in weak conditions. *J. Math. Anal. Appl.* **2006**, *324*, 381–394. [CrossRef]
6. Fan, J.; Pan, J. On the convergence rate of the inexact Levenberg-Marquardt method. *J. Ind. Manag. Optim.* **2011**, *7*, 199–210. [CrossRef]
7. Yamashita, N.; Fukushima, M. On the Rate of Convergence of the Levenberg-Marquardt Method. *J. Comput. Suppl.* **2001**, *15*, 227–238.
8. De Luca, T.; Facchinei, F.; Kanzow, C.T. A theoretical and numerical comparison of some semismooth algorithms for complementarity problems. *Comp. Optim. Appl.* **2000**, *16*, 173–205. [CrossRef]
9. Xiu, N.; Zhang, J. A smoothing Gauss–Newton method for the generalized HLCP. *J. Comput. Appl. Math.* **2001**, *129*, 195–208. [CrossRef]
10. Subramanian, P.K.; Xiu, N.H. Convergence analysis of Gauss–Newton method for the complementarity problem. *J. Optim. Theory Appl.* **1997**, *94*, 727–738. [CrossRef]
11. Martínez, J.M.; Qi, L. Inexact Newton method for solving nonsmooth equations. *J. Comput. Appl. Math.* **1995**, *60*, 127–145. [CrossRef]

12. Bao, J.F.; Li, C.; Shen, W.P.; Yao, J.C.; Guu, S.M. Approximate Gauss–Newton methods for solving underdetermined nonlinear least squares problems. *App. Num. Math.* **2017**, *111*, 92–110. [[CrossRef](#)]
13. Cartis, C.; Roberts, L. A derivative-free Gauss–Newton method. *Math. Program. Comput.* **2019**, *11*, 631–674. [[CrossRef](#)]
14. Clarke, F.H. *Optimization and Nonsmooth Analysis*; John Wiley & Sons: New York, NY, USA, 1983.
15. Qi, L. Convergence analysis of some algorithms for solving nonsmooth equations. *Math. Oper. Res.* **1993**, *18*, 227–244. [[CrossRef](#)]
16. Mifflin, R. Semismooth and semiconvex functions in constrained optimization. *SIAM J. Control Optim.* **1977**, *15*, 142–149. [[CrossRef](#)]
17. Qi, L.; Sun, D. A nonsmooth version of Newton’s method. *Math. Program.* **1993**, *58*, 353–367. [[CrossRef](#)]
18. Pang, J.S.; Qi, L. Nonsmooth equations: Motivation and algorithms. *SIAM J. Optim.* **1993**, *3*, 443–465. [[CrossRef](#)]
19. Ulbrich, M. Nonmonotone trust-region methods for bound-constrained semismooth systems of equations with applications to nonlinear mixed complementarity problems. *SIAM J. Optim.* **2001**, *11*, 889–917. [[CrossRef](#)]
20. Kojima, M.; Shindo, S. Extensions of Newton and quasi-Newton methods to systems of PC<sup>1</sup>-equations. *J. Oper. Res. Soc. Jpn.* **1986**, *29*, 352–374. [[CrossRef](#)]
21. Jiang, H.; Qi, L. A new nonsmooth equations approach to nonlinear complementarity problems. *SIAM J. Control Optim.* **1997**, *35*, 178–193. [[CrossRef](#)]
22. Feng, Y.; Wang, G.; Li, W.; Li, N. Multi-strategy monarch butterfly optimization algorithm for discounted 0–1 knapsack problem. *Neural Comput. Appl.* **2018**, *30*, 3019–3036. [[CrossRef](#)]
23. Feng, Y.; Yu, X.; Wang, G. A novel monarch butterfly optimization with global position updating operator for large-scale 0–1 knapsack problems. *Mathematics* **2019**, *7*, 1056. [[CrossRef](#)]
24. Wang, G.; Deb, S.; Coelho, L.D. Earthworm optimisation algorithm: A bio-inspired metaheuristic algorithm for global optimisation problems. *Int. J. Biol. Inspired Comput.* **2018**, *12*, 1–22. [[CrossRef](#)]
25. Wang, G.; Deb, S.; Coelho, L.D. Elephant herding optimization. In Proceedings of the 2015 3rd International Symposium on Computational and Business Intelligence (ISCBI), Bali, Indonesia, 7–9 December 2015; pp. 1–5.
26. Wang, G.; Deb, S.; Gao, X.; Coelho, L.D. A new metaheuristic optimisation algorithm motivated by elephant herding behaviour. *Int. J. Biol. Inspired Comput.* **2016**, *8*, 394–409. [[CrossRef](#)]
27. Feng, Y.; Wang, G. Binary moth search algorithm for discounted 0-1 knapsack problem. *IEEE Access* **2018**, *6*, 10708–10719. [[CrossRef](#)]
28. Wang, G. Moth search algorithm: A bio-inspired metaheuristic algorithm for global optimization problems. *Memet. Comput.* **2018**, *10*, 151–164. [[CrossRef](#)]



© 2020 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).