A nonmonotone semismooth inexact Newton method

SILVIA BONETTINI[*], FEDERICA TINTI

Dipartimento di Matematica, Università di Modena e Reggio Emilia, Italy

Abstract

In this work we propose a variant of the inexact Newton method for the solution of semismooth nonlinear systems of equations. We introduce a nonmonotone scheme, which couples the inexact features with the nonmonotone strategies. For the nonmonotone scheme, we present the convergence theorems. Finally, we show how we can apply these strategies in the variational inequalities context and we present some numerical examples.
Keywords: Semismooth Systems, Inexact Newton Methods, Nonmonotone Convergence, Variational Inequality Problems, Nonlinear Programming Problems.

1 Introduction

The inexact Newton method is one of the methods for the solution of the nonlinear system of equations

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

where $F : \mathbb{R}^n \to \mathbb{R}^n$ is a continuously differentiable function. The idea of this method was presented firstly in [4], with local convergence properties; then in [7] the authors proposed a global version of the method. Furthermore, the inexact Newton method was proposed also for the solution of nonsmooth equations (see for example [12], [8]).

[*]Corresponding author. Email bonettini.silvia@unimo.it
This research was supported by the Italian Ministry for Education, University and Research(MIUR), FIRB Project RBAU01JYPN.

An inexact Newton method is every method which generates a sequence satisfying the following properties,

$$\|F(x_k) + G(x_k, s_k)\| \le \eta_k \|F(x_k)\| \tag{2}$$

and

$$\|F(x_k + s_k)\| \le (1 - \beta(1 - \eta_k))\|F(x_k)\| \tag{3}$$

where $G : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ is a given iteration function, $x_{k+1} = x_k + s_k$, η_k is the *forcing term*, i.e. a scalar parameter chosen in the interval $[0, 1)$, and β is a positive number fixed in $(0, 1)$.

In the smooth case, we could choose $G(x, s) = \nabla F(x)^T s$, where $\nabla F(x)^T$ is the Jacobian matrix of F, while if F is just semismooth, we can take $G(x, s) = Hs$ where H is a matrix of the B-subdifferential of F at x (for the definition of semismooth function and B-subdifferential see for example [17] and [18]).

The quantity $\|F(x_k)\|$ on the right hand side of the condition (2) represents the residual of the nonlinear system (1) at the iterate k. Furthermore, the condition (2) implies that the vector s_k is an approximate solution of the equation

$$F(x_k) + G(x_k, s) = 0, \tag{4}$$

since the left hand side of (2) is the residual of (4), and the tolerance of such approximation is given by the term $\eta_k \|F(x_k)\|$.

The condition (3) implies that the ratio of the norms of the vector F computed in two successive iterates is less than $(1 - \beta(1 - \eta_k))$, a quantity less than one.

It is worth to stressing that both conditions are depending by the forcing term η_k.

There are many advantages in the algorithms with inexact features, from the theoretical and practical point of view. Indeed, global convergence theorems can be proved under some standard assumptions. Furthermore, the condition (2) tells us that an adaptive tolerance can be introduced in the solution of the iteration function equation (4), saving unnecessary computations when we are far from the solution.

A further relaxation on the requirements can be obtained by allowing nonmonotone choices. The nonmonotone strategies (see for example [10]) are well known in the literature for their effectiveness in the choice of the steplength in many line–search algorithms.

In [1] the nonmonotone convergence has been proved in the smooth case for an inexact Newton line–search algorithm and the numerical experience shows that the nonmonotone strategies can be useful in this kind of algorithm to avoid the stagnation of the iterates in a neighborhood of some "critical" points.

In this paper we propose to modify the general inexact Newton scheme (2) and (3) in a nonmonotone way, by substituting (2) and (3) with the following conditions

$$\|F(x_k) + G(x_k, s_k)\| \le \eta_k \|F(x_{\ell(k)})\| \tag{5}$$

and

$$\|F(x_k + s_k)\| \le (1 - \beta(1 - \eta_k))\|F(x_{\ell(k)})\| \tag{6}$$

where F is a semismooth function and, given $N \in \mathbb{N}$, $x_{\ell(k)}$ is the element with the following property

$$\|F(x_{\ell(k)})\| = \max_{0 \le j \le \min(N,k)} \|F(x_{k-j})\|. \tag{7}$$

The nonmonotone conditions (5)-(6) can be considered as a generalization of the global method for smooth equations presented in [1], and in this work, we provide convergence theorems under analogous assumptions.

In the following section we recall some basic definitions and some results for the semismooth functions; in section 3 we describe the general scheme of our nonmonotone semismooth inexact method and we prove the convergence theorems; in section 4 we apply the method to a particular semismooth system arising from variational inequalities and nonlinear programming problems and, in section 5, we report the numerical results.

2 The semismooth case

Now we consider the nonlinear system of equations (1) with a nonsmooth operator F; in particular we focus on the case in which the system (1) is semismooth.

In order to introduce the semismooth notion, we also report the B-subdifferential and the generalized gradient definitions. We consider a vector-valued function $F : \mathbb{R}^n \to \mathbb{R}^n$, with $F(x) = [f_1(x), \ldots, f_n(x)]^T$ and we assume that, for each component f_i, a Lipschitz condition near a given point x holds. This means that a function $F : \mathbb{R}^n \to \mathbb{R}^n$ is said locally Lipschitz near a given point $x \in \mathbb{R}^n$, if there exists a positive number δ such that each f_i satisfies

$$\|f_i(x_1) - f_i(x_2)\| \le l_i \|x_1 - x_2\| \qquad \forall x_1, x_2 \in N_\delta(x), \quad l_i \in \mathbb{R}$$

where $N_\delta(x)$ is the set $\{y \in \mathbb{R}^n : \|y - x\| < \delta\}$ and $L = (l_1, \ldots, l_n)$ is called rank of F.
The function $F : \mathbb{R}^n \to \mathbb{R}^n$ is locally Lipschitz if, for any $x \in \mathbb{R}^n$, F is locally Lipschitz near x.
Rademacher's Theorem asserts that F is differentiable almost everywhere (i.e. each f_i is differentiable almost everywhere) on any neighborhood of x in which F is a locally Lipschitz function.

We denote with Ω_F the set of points at which F fails to be differentiable.

Definition 2.1 B-subdifferential ([17])
Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a locally Lipschitz function near a given point x (with $x \in \Omega_F$). The B-subdifferential of F at x is

$$\partial_B F(x) = \{Z \in \mathbb{R}^{n\times n} : \exists\{x^k\} \not\subseteq \Omega_F, \text{with} \lim_{x^k \to x} \nabla F(x^k)^T = Z\}.$$

Definition 2.2 (**Clarke's generalized Jacobian ([3])**)
Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a locally Lipschitz function near a given point x. Clarke's generalized Jacobian of F at x is

$$\partial F(x) = co\partial_B F(x)$$

where *co* denotes the convex combinations in the space $\mathbb{R}^{n\times n}$.

Remark: Clarke's generalized Jacobian is the convex hull of all matrices Z obtained as the limit of sequence of the form $\nabla F(x_i)^T$ where $x_i \to x$ and $x_i \notin \Omega_F$.

Now we can finally define the semismooth function, as follows.

Definition 2.3 ([18]) Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be locally Lipschitzian near a given point $x \in \mathbb{R}^n$. We say that F is **semismooth** at x if

$$\lim_{\substack{Z \in \partial F(x+tv') \\ v' \to v, t \downarrow 0}} Zv'$$

exists for all $v \in \mathbb{R}^n$.

The following definition of a BD-regular vector plays a crucial role in establishing global convergence results of several iterative methods.

Definition 2.4 ([16]) Let $F : \mathbb{R}^n \to \mathbb{R}^n$. We say that a point $x \in \mathbb{R}^n$ is BD–regular for F (F is BD–regular at x) if F is locally Lipschitz near x and if all the elements in the B–subdifferential $\partial_B F(x)$ are nonsingular.

The next results play an important role in establishing the global convergence of the semismooth Newton methods.

Proposition 2.1 ([17]) If $F : \mathbb{R}^n \to \mathbb{R}^n$ is BD–regular at x_*, then there exists a positive number δ and a constant $K > 0$ such that for all $x \in N_\delta(x_*)$ and all $H \in \partial_B F(x)$, H is nonsingular and

$$\|H^{-1}\| \leq K$$

Proposition 2.2 ([16]) If $F : \mathbb{R}^n \to \mathbb{R}^n$ is semismooth at a point $x \in \mathbb{R}^n$ then for any $\epsilon > 0$ there exists a $\delta > 0$ such that

$$\|F(y) - F(x) - H \cdot (y - x)\| \leq \epsilon \|y - x\|$$

for all $H \in \partial_B F(y)$, for all $y \in N_\delta(x)$.

3 Nonmonotone semismooth inexact Newton methods

We define a nonmonotone semismooth Newton method every method which generates a sequence $\{x_k\}$ such that

$$\|F(x_k) + H_k s_k\| \leq \eta_k \|F(x_{\ell(k)})\| \tag{8}$$

and

$$\|F(x_k + s_k)\| \leq (1 - \beta(1 - \eta_k))\|F(x_{\ell(k)})\| \tag{9}$$

where $x_{\ell(k)}$ is defined in (7), $x_{k+1} = x_k + s_k$, $H_k \in \partial_B F(x_k)$, $\eta_k \in [0, 1)$ and β is a positive parameter fixed in $(0, 1)$.
We will call the vector s_k which satisfies (8) *nonmonotone semismooth inexact Newton step* at the level η_k.
For a sequence satisfying (8) and (9) it is possible to prove the following convergence result which is fundamental for the convergence proofs of the algorithms presented in the following.
In [7, Theorem 3.3] and [19] an analogous result can be found for the smooth case.

Theorem 3.1 Let $F : \mathbb{R}^n \to \mathbb{R}^n$ be a locally Lipschitz function. Let $\{x_k\}$ be a sequence such that $\lim_{k\to\infty} F(x_k) = 0$ and for each k the following conditions hold:

$$\|F(x_k) + H_k s_k\| \leq \eta \|F(x_{\ell(k)})\|, \tag{10}$$

$$\|F(x_{k+1})\| \leq \|F(x_{\ell(k)})\|, \tag{11}$$

where $H_k \in \partial_B F(x_k)$, $s_k = x_{k+1} - x_k$ and $\eta < 1$. If x_* is an accumulation point [1] of $\{x_k\}$, then $F(x_*) = 0$. Furthermore, if F is semismooth at x_* and F is BD-regular at x_* , then the sequence $\{x_k\}$ converges to x_*.

Proof. If x_* is an accumulation point of the sequence $\{x_k\}$, there exists a subsequence $\{x_{k_j}\}$ of $\{x_k\}$ convergent to x_*. By the continuity of F, we obtain

$$F(x_*) = F\left(\lim_{j\to\infty} x_{k_j}\right) = \lim_{j\to\infty} F(x_{k_j}) = 0.$$

Furthermore, since $\{x_{\ell(k)}\}$ is a subsequence of $\{x_k\}$, also the sequence $\{F(x_{\ell(k)})\}$ converges to zero when k diverges. From Proposition 2.1, there exist $\delta > 0$ and a constant K such that each $H \in \partial_B F(x)$ is nonsingular and $\|H^{-1}\| \leq K$ for any $x \in N_\delta(x_*)$; we can suppose that δ is sufficiently small such that Proposition 2.2 implies

$$\|F(y) - F(x_*) - H_y(y - x_*)\| \leq \frac{1}{2K}\|y - x_*\|$$

for $y \in N_\delta(x_*)$ and for any $H_y \in \partial_B F(y)$. Then for any $y \in N_\delta(x_*)$ we have

$$\begin{array}{rcl} \|F(y)\| & = & \|H_y(y - x_*) + F(y) - F(x_*) - H_y(y - x_*)\| \\ & \geq & \|H_y(y - x_*)\| - \|F(y) - F(x_*) - H_y(y - x_*)\| \\ & \geq & \frac{1}{K}\|y - x_*\| - \frac{1}{2K}\|y - x_*\| \\ & = & \frac{1}{2K}\|y - x_*\|. \end{array}$$

Then

$$\|y - x_*\| \leq 2K\|F(y)\| \tag{12}$$

holds for any $y \in N_\delta(x_*)$. Now let $\epsilon \in (0, \frac{\delta}{4})$ and since x_* is an accumulation point of $\{x_k\}$, there exists a k sufficiently large that

$$x_k \in N_{\frac{\delta}{2}}(x_*)$$

[1] We say that x_* ia an accumulation point for the sequence $\{x_k\}$ if for any positive number δ there exists an index j such that $\|x_j - x_*\| < \delta$.
We say that the sequence $\{x_k\}$ converges to a point x_* (or, equivalently, that x_* is the limit point of the sequence $\{x_k\}$) if for any positive number δ there exists an integer n such that $\|x_j - x_*\| < \delta$ for any $j \geq n$.

and

$$x_{\ell(k)} \in S_\epsilon \equiv \left\{ y : \|F(y)\| < \frac{\epsilon}{K(1+\eta)} \right\}.$$

Note that since $x_{\ell(k)} \in S_\epsilon$ then also $x_{k+1} \in S_\epsilon$ because $\|F(x_{k+1})\| \leq \|F(x_{\ell(k)})\|$. For the direction s_k, by (10), (11) and since $\|H_k^{-1}\| \leq K$, the following inequality holds:

$$\begin{aligned} \|s_k\| &\leq \|H_k^{-1}\|(\|F(x_k)\| + \|F(x_k) + H_k s_k\|) \\ &\leq K(\|F(x_{\ell(k)})\| + \eta\|F(x_{\ell(k)})\|) \\ &= K(1+\eta)\|F(x_{\ell(k)})\| < \epsilon < \tfrac{\delta}{2}. \end{aligned}$$

Since $s_k = x_{k+1} - x_k$, the previous inequality implies $\|x_{k+1} - x_*\| < \delta$ and from (12) we obtain

$$\|x_{k+1} - x_*\| \leq 2K\|F(x_{k+1})\| < 2K\frac{\epsilon}{K(1+\eta)} < \frac{\delta}{2}$$

that implies $x_{k+1} \in N_{\frac{\delta}{2}}(x_*)$. Therefore $x_{\ell(k+1)} \in S_\epsilon$, since $\|F(x_{\ell(k+1)})\| \leq \|F(x_{\ell(k)})\|$. It follows that, for any j sufficiently large, $x_j \in N_\delta(x_*)$, and from (12)

$$\|x_j - x_*\| \leq 2K\|F(x_j)\|.$$

Since $F(x_j)$ converges to 0 we can conclude that x_j converges to x_*. $\square$

3.1 A line–search semismooth inexact Newton algorithm

In this section we describe a line–search algorithm: once computed a semismooth inexact Newton step, the steplengh is reduced by a backtracking procedure until an acceptance rule is satisfied.
In the remaining of the section, we prove that the proposed algorithm is well defined.

Algorithm 3.1

Step 1 Choose $x_0 \in \mathbb{R}^n$, $\beta \in (0,1)$, $0 < \theta_{min} < \theta_{max} < 1$, $\eta_{max} \in (0,1)$. Set $k = 0$;

Step 2 (Search direction)
Select an element $H_k \in \partial_B F(x_k)$.
Determine $\bar{\eta}_k \in [0, \eta_{max}]$ and $\bar{s}_k$ that satisfy

$$\|H_k \bar{s}_k + F(x_k)\| \leq \bar{\eta}_k \|F(x_{\ell(k)})\|;$$

Step 3 (Linesearch)

While $\|F(x_k + \alpha_k \bar{s}_k)\| > (1 - \alpha_k \beta(1 - \bar{\eta}_k))\|F(x_{\ell(k)})\|$

Step 3.a Choose $\theta \in [\theta_{min}, \theta_{max}]$;

Step 3.b Set $\alpha_k = \theta \alpha_k$;

End

Step 4 Set $x_{k+1} = x_k + \alpha_k \bar{s}_k$;

Step 5 Set $k = k + 1$ and go to **Step 2**.

The steplength is represented by the damping parameter α_k which is reduced until the backtracking condition

$$\|F(x_k + \alpha_k \bar{s}_k)\| \leq (1 - \alpha_k \beta(1 - \bar{\eta}_k))\|F(x_{\ell(k)})\| \tag{13}$$

is satisfied. Condition (13) is more general than the Armijo condition employed for example in [8], since it does not require the differentiability of the merit function $\Psi(x) = 1/2\|F(x)\|^2$.
The final inexact Newton step is given by $s_k = \alpha_k \bar{s}_k$, and it satisfies conditions (8) and (9) with forcing term $\eta_k = 1 - \alpha_k(1 - \bar{\eta}_k)$.
We will simply assume that at each iterate k it is possible to compute the vector $\bar{s}_k$ which is an inexact Newton step at the level $\bar{\eta}_k$ (see for example the assumption A1 in [12] for a sufficient condition). The next lemma shows that, under the previous assumption, the sequence generated by Algorithm 3.1 satisfies conditions (8) and (9).

Lemma 3.1 Let $\beta \in (0, 1)$; suppose that there exist $\bar{\eta} \in [0, 1)$, $\bar{s}$ satisfying

$$\|F(x_k) + H_k \bar{s}\| \leq \bar{\eta}\|F(x_{\ell(k)})\|.$$

Then, there exist $\alpha_{max} \in (0, 1]$ and a vector s such that

$$\|F(x_k) + H_k s\| \leq \eta \|F(x_{\ell(k)})\| \tag{14}$$

$$\|F(x_k + s)\| \leq (1 - \beta\alpha(1 - \eta))\|F(x_{\ell(k)})\| \tag{15}$$

hold for any $\alpha \in (0, \alpha_{max}]$, where $\eta \in [\bar{\eta}, 1)$, $\eta = (1 - \alpha(1 - \bar{\eta}))$.

Proof. Let $s = \alpha \bar{s}$. Then we have

$$\begin{aligned}
\|F(x_k) + H_k s\| &= \|F(x_k) - \alpha F(x_k) + \alpha F(x_k) + \alpha H_k \bar{s}\| \\
&\leq (1 - \alpha)\|F(x_k)\| + \alpha\|F(x_k) + H_k \bar{s}\| \\
&\leq (1 - \alpha)\|F(x_{\ell(k)})\| + \alpha\bar{\eta}\|F(x_{\ell(k)})\| \\
&= \eta\|F(x_{\ell(k)})\|,
\end{aligned}$$

so (14) is proved. Now let

$$\varepsilon = \frac{(1-\beta)(1-\bar{\eta})}{\|\bar{s}\|}\|F(x_{\ell(k)})\|, \tag{16}$$

and $\delta > 0$ be sufficiently small (see Proposition 2.2) that

$$\|F(x_k + s) - F(x_k) - H_k s\| \le \varepsilon \|s\| \tag{17}$$

whenever $\|s\| \le \delta$. Choosing $\alpha_{max} = \min(1, \frac{\delta}{\|\bar{s}\|})$, for any $\alpha \in (0, \alpha_{max}]$ we have $\|s\| \le \delta$ and then, using (16) and (17), we obtain the following inequality

$$\begin{aligned}
\|F(x_k + s)\| &\le \|F(x_k + s) - F(x_k) - H_k s\| + \|F(x_k) + H_k s\| \\
&\le \varepsilon\alpha\|\bar{s}\| + \eta\|F(x_{\ell(k)})\| \\
&= ((1-\beta)(1-\bar{\eta})\alpha + (1-\alpha(1-\bar{\eta})))\|F(x_{\ell(k)}\| \\
&= (1-\beta\alpha(1-\bar{\eta}))\|F(x_{\ell(k)})\| \\
&\le (1-\beta\alpha(1-\eta))\|F(x_{\ell(k)})\|,
\end{aligned}$$

that completes the proof. □

A consequence of the previous lemma is that the backtracking loop at the step 3 of Algorithm 3.1, at each iterate k, terminates in a finite number of steps. Indeed, at each iterate k the backtracking condition (13) is satisfied for $\alpha < \alpha_{max}$, where α_{max} depends on k. Since the value of α_k is reduced by a factor $\theta < \theta_{max} < 1$, then there exists a positive integer p such that $(\theta_{max})^p < \alpha_{max}$ and so *the while loop* terminates at most after p steps.
When, at some iterate k, it is impossible to determine the next point x_{k+1} satisfying (8) and (9), we say that the algorithm *breaks down*. Then, Lemma 3.1 yields that assuming that it is possible to compute the semismooth inexact Newton step $\bar{s}_k$ satisfying (8), then Algorithm 3.1 does not break down and it is well defined.

3.2 Convergence Analysis

The next theorem proves, under appropriate assumptions, that the sequence $\{x_k\}$ generated by Algorithm 3.1 converges to a solution of the system (1). The proof is carried out by showing that $\lim_{k\to+infty} \|F(x_k)\| = 0$, so that the convergence of the sequence is ensured by Theorem 3.1.

Theorem 3.2 Suppose that $\{x_k\}$ is the sequence generated by Algorithm 3.1, with $2\beta < 1 - \eta_{max}$. Assume that the following conditions hold:

A1 There exists an accumulation point x_* of the sequence $\{x_k\}$, such that F is semismooth and BD-regular at x_*;

A2 At each iterate k it is possible to find a forcing term $\bar{\eta}_k$ and a vector $\bar{s}_k$ such that the inexact residual condition (8) is satisfied;

A3 For every sequence $\{x_k\}$ converging to x_*, every convergent sequence $\{s_k\}$ and every sequence $\{\lambda_k\}$ of positive scalars converging to zero,

$$\limsup_{k\to+\infty} \frac{\Psi(x_k + \lambda_k s_k) - \Psi(x_{\ell(k)})}{\lambda_k} \le \lim_{k\to+\infty} F(x_k)^T H_k s_k,$$

where $\Psi(x) = 1/2\|F(x)\|^2$, whenever the limit on the left-hand side exists;

A4 For every sequence $\{x_{k_j}\}$ such that α_{k_j} converges to zero, then $\|\bar{s}_{k_j}\|$ is bounded.

Then, $F(x_*) = 0$ and the sequence $\{x_k\}$ converges to x_*.

Proof. Assumption A1 implies that the norm of the vector $\|\bar{s}_k\|$ is bounded in a neighborhood of the point x_*. Indeed, from Proposition 2.1, there exists a positive number δ and a constant K such that $\|H_k^{-1}\| \le K$ for any $H_k \in \partial_B F(x_k)$, for any $x_k \in N_\delta(x_*)$.
Thus, the following conditions hold:

$$\begin{aligned}
\|\bar{s}_k\| &\le \|H_k^{-1}\|(\|F(x_k)\| + \|F(x_k) + H_k\bar{s}_k\|) \\
&\le K(\|F(x_{\ell(k)})\| + \eta_{max}\|F(x_{\ell(k)})\|) \\
&= K(1+\eta_{max})\|F(x_{\ell(k)})\| \\
&\le K(1+\eta_{max})\|F(x_0)\|.
\end{aligned}$$

Furthermore, the condition A2 ensures that the Algorithm 3.1 does not break down, thus it generates an infinite sequence.
Now we consider separately the two following cases:

a) There exists a set of indices K such that $\{x_k\}_{k\in K}$ converges to x_* and $\liminf_{k\to+\infty, k\in K} \alpha_k = 0$;

b) For any subsequence $\{x_k\}_{k\in K}$ converging to x_* we have $\liminf_{k\to+\infty, k\in K} \alpha_k = \tau > 0$.

a)Since $\|F(x_{\ell(k)})\|$ is a monotone nonincreasing, bounded sequence, then there exists $L \geq 0$ such that

$$L = \lim_{k\to\infty} \|F(x_{\ell(k)})\|. \tag{18}$$

From the definition (7) it follows that $\|F(x_{\ell(k)})\| \geq \|F(x_k)\|$, thus

$$L \geq \lim_{k\to+\infty, k\in I} \|F(x_k)\| \tag{19}$$

where $\{x_k\}_{k\in I}$ is a subsequence of $\{x_k\}$ such that the limit of the sequence $\{\|F(x_k)\|\}_{k\in I}$ exists.
Since α_k is the final value after the backtracking loop, we must have

$$\|F(x_k + \frac{\alpha_k}{\theta}\bar{s}_k)\| > \left(1 - \frac{\alpha_k}{\theta}\beta(1-\bar{\eta}_k)\right)\|F(x_{\ell(k)})\| \tag{20}$$

which yields

$$\lim_{k\to+\infty, k\in K} \|F(x_k + \frac{\alpha_k}{\theta}\bar{s}_k)\| \geq \lim_{k\to+\infty, k\in K} \left(1 - \frac{\alpha_k}{\theta}\beta(1-\bar{\eta}_k)\right)\|F(x_{\ell(k)})\|. \tag{21}$$

If we choose K as the set of indices with the property a), exploiting the continuity of F, recalling that $\bar{\eta}_k$ is bounded, that $\|\bar{s}_k\|$ is bounded and subsequencing to ensure the existence of the limit of α_k , we obtain $\|F(x_*)\| \geq L$. On the other hand, from (19) we have also that $L \geq \|F(x_*)\|$, thus it follows that

$$L = \|F(x_*)\|. \tag{22}$$

Furthermore, by squaring both sides of (20), we obtain the following inequalities

$$\begin{aligned}
\|F(x_k + \frac{\alpha_k}{\theta}\bar{s}_k)\|^2 &> \left(1 - \frac{\alpha_k}{\theta}\beta(1-\bar{\eta}_k)\right)^2\|F(x_{\ell(k)})\|^2 \\
&\geq \left(1 - 2\frac{\alpha_k}{\theta}\beta(1-\bar{\eta}_k)\right)\|F(x_{\ell(k)})\|^2.
\end{aligned}$$

This yields

$$\|F(x_k + \frac{\alpha_k}{\theta}\bar{s}_k)\|^2 - \|F(x_{\ell(k)})\|^2 > -2\frac{\alpha_k}{\theta}\beta(1-\bar{\eta}_k)\|F(x_{\ell(k)})\|^2. \tag{23}$$

Dividing both sides by $\frac{\alpha_k}{\theta}$, passing to the limit and exploiting the assumption A4, we obtain

$$\begin{aligned}
\lim_{k\to+\infty, k\in K} F(x_k)^T H_k s_k &\geq \lim_{k\to+\infty, k\in K} \frac{\|F(x_k + \frac{\alpha_k}{\theta}\bar{s}_k)\|^2 - \|F(x_{\ell(k)})\|^2}{\frac{\alpha_k}{\theta}} \\
&\geq \lim_{k\to+\infty, k\in K} -2\beta(1-\bar{\eta}_k)\|F(x_{\ell(k)})\|^2. \qquad (24)
\end{aligned}$$

Since (22) holds and taking into account that $\bar{\eta}_k \geq 0$, we have

$$\lim_{k\to+\infty, k\in K} F(x_k)^T H_k s_k \geq -2\beta \|F(x_*)\|^2. \tag{25}$$

On the other hand, we have

$$\begin{aligned} F(x_k)^T H_k \bar{s}_k &= F(x_k)^T \left[-F(x_k) + F(x_k) + H_k \bar{s}_k\right] \\ &= -\|F(x_k)\|^2 + F(x_k)^T \left[F(x_k) + H_k \bar{s}_k\right] \\ &\leq -\|F(x_k)\|^2 + \|F(x_k)\| \cdot \|F(x_k) + H_k \bar{s}_k\| \\ &\leq -\|F(x_k)\|^2 + \eta_{max} \|F(x_{\ell(k)})\|^2, \end{aligned} \tag{26}$$

thus we can write

$$\lim_{k\to+\infty} F(x_k)^T H_k \bar{s}_k \quad \leq \quad \lim_{k\to+\infty} -\|F(x_k)\|^2 + \eta_{max} \|F(x_{\ell(k)})\|^2.$$

Furthermore, considering the subsequence $\{x_k\}_{k\in K}$, it follows that

$$\lim_{k\to+\infty, k\in K} F(x_k)^T H_k \bar{s}_k \leq -(1-\eta_{max}) \|F(x_*)\|^2. \tag{27}$$

From (25) and (27) we deduce

$$-2\beta \|F(x_*)\|^2 \leq -(1-\eta_{max}) \|F(x_*)\|^2.$$

Since we set $(1-\eta_{max}) > 2\beta$, then we must have $\|F(x_*)\| = 0$.
This implies that $\lim_{k\to+\infty} \|F(x_{\ell(k)})\| = 0$ and, consequently from (7), we have

$$\lim_{k\to+\infty} \|F(x_k)\| = 0.$$

Thus, the convergence of the sequence is ensured by Theorem 3.1.
b) Writing the backtracking condition for the iterate $\ell(k)$, we obtain

$$\|F(x_{\ell(k)})\| \leq (1 - \alpha_{\ell(k)-1} \beta (1 - \bar{\eta}_{\ell(k)-1})) \|F(x_{\ell(\ell(k)-1)})\|. \tag{28}$$

When k diverges, we can write

$$L \leq L - L \cdot \lim_{k\to\infty} \alpha_{\ell(k)-1} \beta (1 - \bar{\eta}_{\ell(k)-1}), \tag{29}$$

where L is defined as in (18).
Since β is a constant and $1 - \bar{\eta}_j \geq 1 - \eta_{max} > 0$ for any j, the inequality (29) yields

$$L \cdot \lim_{k\to\infty} \alpha_{\ell(k)-1} \leq 0$$

that implies

$$L = 0$$

or

$$\lim_{k\to\infty} \alpha_{\ell(k)-1} = 0. \tag{30}$$

Suppose that $L \neq 0$, so that (30) holds. Defining $\hat{\ell}(k) = \ell(k+N+1)$ so that $\hat{\ell}(k) > k$, we show by induction that for any $j \geq 1$ we have

$$\lim_{k\to\infty} \alpha_{\hat{\ell}(k)-j} = 0 \tag{31}$$

and

$$\lim_{k\to\infty} \|F(x_{\hat{\ell}(k)-j})\| = L. \tag{32}$$

For $j = 1$, since $\{\alpha_{\hat{\ell}(k)-1}\}$ is a subsequence of $\{\alpha_{\ell(k)-1}\}$, (30) implies (31). Thanks to the assumption A4, we also obtain

$$\lim_{k\to\infty} \|x_{\hat{\ell}(k)} - x_{\hat{\ell}(k)-1}\| = 0. \tag{33}$$

By exploiting the Lipschitz property of F, from $|\|F(x)\| - \|F(y)\|| \leq \|F(x) - F(y)\|$ and (33) we obtain

$$\lim_{k\to\infty} \|F(x_{\hat{\ell}(k)-1})\| = L. \tag{34}$$

Assume now that (31) and (32) hold for a given j. We have

$$\|F(x_{\ell(k)-j})\| \leq (1 - \alpha_{\ell(k)-(j+1)}\beta(1 - \eta_{\ell(k)-(j+1)}))\|F(x_{\ell(\ell(k)-(j+1))})\|.$$

Using the same arguments employed above, since $L > 0$, we obtain

$$\lim_{k\to\infty} \alpha_{\hat{\ell}(k)-(j+1)} = 0$$

and so

$$\lim_{k\to\infty} \|x_{\hat{\ell}(k)-j} - x_{\hat{\ell}(k)-(j+1)}\| = 0,$$
$$\lim_{k\to\infty} \|F(x_{\hat{\ell}(k)-(j+1)})\| = L.$$

Thus, we conclude that (31) and (32) hold for any $j \geq 1$. Now, for any k, we can write

$$\|x_{k+1} - x_{\hat{\ell}(k)}\| \leq \sum_{j=1}^{\hat{\ell}(k)-k-1} \alpha_{\hat{\ell}(k)-j} \|\bar{s}_{\hat{\ell}(k)-j}\|$$

so that, since we have $\hat{\ell}(k) - k - 1 \leq N$, we have

$$\lim_{k\to\infty} \|x_{k+1} - x_{\hat{\ell}(k)}\| = 0. \tag{35}$$

Furthermore, we have

$$\|x_{\hat{\ell}(k)} - x_*\| \leq \|x_{\hat{\ell}(k)} - x_{k+1}\| + \|x_{k+1} - x_*\| \tag{36}$$

Since x_* is an accumulation point of $\{x_k\}$ and (35) holds, (36) implies that x_* is an accumulation point for the sequence $\{x_{\hat{\ell}(k)}\}$. From (33) we conclude that x_* is an accumulation point also for the sequence $\{x_{\hat{\ell}(k)-1}\}$, which contradicts the assumption made. Indeed, since $\{x_{\hat{\ell}(k)-1}\}$ converges to x_*, we should have that $\alpha_{\hat{\ell}(k)-1}$ is bounded away from zero, from the hypothesis b). Hence, we necessarily have $L = 0$, that implies

$$\lim_{k\to\infty} \|F(x_k)\| = 0.$$

Now Theorem 3.1 completes the proof.

$\square$

The previous theorem is proved under the assumptions A1–A4: the hypothesis A4 is analogous to the one employed in [1] in the smooth case, while A3 is the nonmonotone, and weaker, version of the assumption (A4) in [12]. This hypothesis is not required in the smooth case, thanks to the stronger properties of the function F and of its Jacobian $\nabla F(x)^T$ (see §3.2.10 in [14]).

4 An application to the Karush–Kuhn–Tucker systems

In this section we consider a particular semismooth system of equations derived from the optimality conditions of variational inequalities or nonlinear programming problems.

We consider the classical variational inequality problem VIP(C,V), which is to find $x^* \in C$, such that

$$< V(x^*), x - x^* > \geq 0, \forall x \in C \tag{37}$$

where C is a nonempty closed convex subset of $\mathbb{R}^n$, $< \cdot, \cdot >$ is the usual inner product in $\mathbb{R}^n$ and $V : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous function.

When V is the gradient mapping of the real-valued function $f : \mathbb{R}^n \to \mathbb{R}$, the problem VIP(C,V) becomes the stationary point problem of the following optimization problem

$$\begin{array}{ll} \min & f(x). \\ \text{s. t. } x \in C & \end{array} \tag{38}$$

We assume, as in [20], that the feasible set C can be represented as follows

$$C = \{x \in \mathbb{R}^n | h(x) = 0, g(x) \geq 0, \Pi_l x \geq l, \Pi_u x \leq u\}, \tag{39}$$

where $h : \mathbb{R}^n \longrightarrow \mathbb{R}^{neq}, g : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, $\Pi_l \in \mathbb{R}^{nl \times n}$ and $\Pi_u \in \mathbb{R}^{nu \times n}$; Π_l (or Π_u) denotes a matrix given by the rows of the identity matrix with indices equal to those of the entries of x which are bounded below (above).
Furthermore, nl and nu denote the number of entries of the vector x subject to lower and upper bounds respectively.
We consider the following conditions, representing the Karush-Kuhn-Tucker (KKT) optimality conditions of VIP(C,V) or of the nonlinear programming problem (38):

$$\begin{array}{l} L(x, \lambda, \mu, \kappa_l, \kappa_u) = 0 \\ h(x) = 0 \\ \mu^T g(x) = 0 \quad g(x) \geq 0 \quad \mu \geq 0 \\ \kappa_l^T (\Pi_l x - l) = 0 \quad \Pi_l x - l \geq 0 \quad \kappa_l \geq 0 \\ \kappa_u^T (u - \Pi_u x) = 0 \quad u - \Pi_u x \geq 0 \quad \kappa_u \geq 0 \end{array} \tag{40}$$

where $L(x, \lambda, \mu, \kappa_l, \kappa_u) = V(x) - \nabla h(x)\lambda - \nabla g(x)\mu - \Pi_l^T \kappa_l + \Pi_u^T \kappa_u$ is the Lagrangian function. Here $\nabla h(x)^T$ and $\nabla g(x)^T$ are the Jacobian matrices of $h(x)$ and $g(x)$ respectively.

In order to rewrite the KKT-conditions as a nonlinear system of equations, we make use of the Fischer's function, [9], $\varphi : \mathbb{R}^2 \to \mathbb{R}$ defined by

$$\varphi(a, b) := \sqrt{a^2 + b^2} - a - b.$$

The main property of this function is the following characterization of its zeros:

$$\varphi(a, b) = 0 \Leftrightarrow a \geq 0, b \geq 0, ab = 0.$$

Therefore, the KKT-conditions (40) can be equivalently written as the nonlinear system of equations

$$\begin{aligned} V(x) - \nabla h(x)\lambda - \nabla g(x)\mu - \Pi_l^T \kappa_l + \Pi_u^T \kappa_u &= 0 \\ h(x) &= 0 \\ \varphi_I(\mu, g(x)) &= 0 \\ \varphi_l(\kappa_l, \Pi_l x - l) &= 0 \\ \varphi_u(\kappa_u, u - \Pi_u x) &= 0 \end{aligned}$$

or, in more concise form,

$$\Phi(w) = 0 \tag{41}$$

where $w = (x^T, \lambda^T, \mu^T, \kappa_l^T, \kappa_u^T)^T$; $\varphi_I : \mathbb{R}^{2m} \to \mathbb{R}^m$ with $\varphi_I(\mu, g(x)) := (\varphi(\mu_1, g_1), \dots, \varphi(\mu_m, g_m))^T$; $\varphi_l : \mathbb{R}^{2nl} \to \mathbb{R}^{nl}$, with $\varphi_l(\kappa_l, \Pi_l x - l) \in \mathbb{R}^{nl}$; $\varphi_u : \mathbb{R}^{2nu} \to \mathbb{R}^{nu}$, with $\varphi_u(\kappa_u, u - \Pi_u x) \in \mathbb{R}^{nu}$.
Note that the functions $\varphi_I, \varphi_l, \varphi_u$ are not differentiable in the origin, so that the system (41) is a semismooth reformulation of the KKT-conditions (40).

The system (41) can be solved by the semismooth inexact Newton method [8], given by

$$w_{k+1} = w_k + \alpha_k \Delta w_k,$$

with a given starting point w_0, where α_k is a damping parameter and Δw_k is the solution of the following linear system

$$H_k \Delta w = -\Phi(w_k) + r_k \tag{42}$$

where $H_k \in \partial_B \Phi(w_k)$ and r_k is the residual vector and it satisfies the condition

$$\|r_k\| \leq \eta_k \|\Phi(w_k)\|.$$

As shown in [20], permuting the equations of the system (42) and changing the sign of the fourth equation, the system (42) can be written as follows:

$$\begin{bmatrix} R_{\kappa_l} & 0 & 0 & R_l\Pi_l & 0 \\ 0 & R_{\kappa_u} & 0 & R_u\Pi_u & 0 \\ 0 & 0 & R_\mu & R_g(\nabla g(x))^T & 0 \\ \Pi_l^T & -\Pi_u^T & \nabla g(x) & -\nabla V(x) + \nabla^2 g(x)\mu + \nabla^2 h(x)\lambda & \nabla h(x) \\ 0 & 0 & 0 & (\nabla h(x))^T & 0 \end{bmatrix} \cdot$$

$$\cdot \begin{bmatrix} \triangle\kappa_l \\ \triangle\kappa_u \\ \triangle\mu \\ \triangle x \\ \triangle\lambda \end{bmatrix} = - \begin{bmatrix} \varphi_l(\kappa_l, \Pi_l x - l) \\ \varphi_u(\kappa_u, u - \Pi_u x) \\ \varphi_I(\mu, g(x)) \\ \overline{\alpha} \\ h(x) \end{bmatrix} + P_r$$

where P_r is the permuting residual vector and $-\overline{\alpha} = V(x) - \nabla h(x)\lambda - \nabla g(x)\mu - (\Pi_l)^T\kappa_l + (\Pi_u)^T\kappa_u$;

$R_g = \mathrm{diag}(r_{g_1}, \dots, r_{g_m})$

$$(r_g)_i = \begin{cases} \left(\dfrac{g_i}{\sqrt{\mu_i^2 + g_i^2}} - 1 \right) & \text{if } (g_i(x), \mu_i) \neq 0 \\ -1 & \text{if } (g_i(x), \mu_i) = 0 \end{cases}$$

$R_\mu = \mathrm{diag}(r_{\mu_1}, \dots, r_{\mu_m})$

$$(r_\mu)_i = \begin{cases} \left(\dfrac{\mu_i}{\sqrt{\mu_i^2 + g_i^2}} - 1 \right) & \text{if } (g_i(x), \mu_i) \neq 0 \\ -1 & \text{if } (g_i(x), \mu_i) = 0 \end{cases}$$

$R_l = \mathrm{diag}(r_{l_1}, \dots, r_{l_{nl}})$

$$(r_l)_i = \begin{cases} \left(\dfrac{((\Pi_l x)_i - l_i)}{\sqrt{(\kappa_l)_i^2 + ((\Pi_l x)_i - l_i)^2}} - 1 \right) & \text{if } ((\Pi_l x)_i - l_i, (\kappa_l)_i) \neq 0 \\ -1 & \text{if } ((\Pi_l x)_i - l_i, (\kappa_l)_i) = 0 \end{cases}$$

$R_{\kappa_l} = \mathrm{diag}(r_{\kappa_{l_1}}, \dots, r_{\kappa_{l_{nl}}})$

$$(r_{\kappa_l})_i = \begin{cases} \left(\dfrac{(\kappa_l)_i}{\sqrt{(\kappa_l)_i^2 + ((\Pi_l x)_i - l_i)^2}} - 1 \right) & \text{if } ((\kappa_l)_i, (\Pi_l x)_i - l_i) \neq 0 \\ -1 & \text{if } ((\kappa_l)_i, (\Pi_l x)_i - l_i) = 0 \end{cases}$$

$R_u = \mathrm{diag}(r_{u_1}, \ldots, r_{u_{nu}})$

$$(r_u)_i = \begin{cases} -\left(\dfrac{(u_i - (\Pi_u x)_i)}{\sqrt{(\kappa_u)_i^2 + (u_i - (\Pi_u x)_i)^2}} - 1 \right) & \text{if } (u_i - (\Pi_u x)_i, (\kappa_u)_i) \neq 0 \\ -1 & \text{if } (u_i - (\Pi_u x)_i, (\kappa_u)_i) = 0 \end{cases}$$

$R_{\kappa_u} = \mathrm{diag}(r_{\kappa_{u_1}}, \ldots, r_{\kappa_{u_{nu}}})$

$$(r_{\kappa_u})_i = \begin{cases} \left(\dfrac{(\kappa_u)_i}{\sqrt{(\kappa_u)_i^2 + (u_i - (\Pi_u x)_i)^2}} - 1 \right) & \text{if } ((\kappa_u)_i, u_i - (\Pi_u x)_i) \neq 0 \\ -1 & \text{if } ((\kappa_u)_i, u_i - (\Pi_u x)_i) = 0. \end{cases}$$

Now we define the merit function $\Psi : \mathbb{R}^{m+n+neq+nl+nu} \rightarrow \mathbb{R}$ as

$$\Psi(w) = \frac{1}{2}\|\Phi(w)\|^2. \tag{43}$$

The differentiability of the function $\Psi(w)$ plays a crucial role in the globalized strategy of the semismooth inexact Newton method proposed in [8]. In the approach followed here, this property is not required, since the convergence theorem 3.2 can be proved without assuming this hypothesis, thanks to the backtracking rule (28) which is similar to the ones proposed in [7] and in [12]. Now we introduce the nonmonotone inexact Newton algorithm, as follows:

Algorithm 4.1

Step 1 Choose $w_0 = (x_0, \lambda_0, \mu_0, \kappa_{l_0}, \kappa_{u_0}) \in \mathbb{R}^{m+n+neq+nl+nu}, \theta \in (0,1), \beta \in (0, 1/2)$ and fix $\eta_{max} < 1$; $\lambda_0 = 0, \kappa_{l_0} = 0, \kappa_{u_0} = 0, \mu_0 = 0$.

Step 2 (Stopping criterion)
if $\|\Phi(w_k)\| \leq tol$ then stop
else

Step 3 (Search direction Δw)
Select an element $H_k \in \partial_B \Phi(w_k)$.
Find the direction $\Delta w_k \in \mathbb{R}^n$ and a parameter $\eta_k \in [0, \eta_{max}]$ such that

$$\|H_k \Delta w_k + \Phi(w_k)\| \leq \eta_k \|\Phi(w_{\ell(k)})\| \tag{44}$$

Step 4 (Linesearch)
Compute the minimum integer h, such that, if $\alpha_k = \theta^h$ the following condition holds

$$\|\Phi(w_k + \alpha_k \Delta w_k)\| \leq (1 - \beta\alpha_k(1 - \eta_k))\|\Phi(w_{\ell(k)})\| \tag{45}$$

Step 5 Compute $w_{k+1} = w_k + \alpha_k \Delta w_k$ go to **Step 2**.

It is straightforward to observe that Algorithm 4.1 is a special case of Algorithm 3.1. Furthermore, the merit function $\Psi(w)$ is differentiable and $\nabla\Psi(w) = H^T\Phi(w)$ where H belongs to $\partial_B\Phi(w)$ (see Proposition 4.2 in [8]). This implies that the hypothesis A3 holds [12].
Moreover we assume that H_k in (44) is nonsingular and that all the iterates w_k belong to a compact set. As a consequence, we have that the norm of the search direction Δw_k is bounded: indeed, for any k, from (44) we obtain

$$\|\Delta w_k\| \leq M(1 + \eta_{max})\|\Phi(w_0)\|$$

where $M = \max_{w_k} \|H_k^{-1}\|$.

5 Numerical results

In this section we report some numerical experiments, obtained by coding Algorithm 4.1 in FORTRAN 90 using double precision on HP zx6000 workstation with Itanium2 processor with 1.3 GHz and 2 Gb of RAM, running HP-UX operating system.
In particular, we set $\beta = 10^{-4}$, $\theta = 0.5$, $tol = 10^{-8}$.
Our aim is to compare the performances of the Algorithm 4.1 with different monotonicity degrees, by choosing different values for the parameter N.
We declare a failure of the algorithm when the tolerance tol can not be reached after 500 iterations or when, in order to satisfy the backtracking condition (45), more than 30 reductions of the damping parameter have been performed.
The forcing term η_k has been adaptively chosen as

$$\eta_k = \max\left(\frac{1}{1+k}, 10^{-8}\right).$$

The solution of the linear system (44) is computed by the LSQR method [15] with a suitable preconditioner proposed in [20]. The stopping criterion for the inner linear solver is the condition (44).

The test problems we considered are the nonlinear programming problems and the complementarity problems listed in Table 1, where we also report the number of variables n, the number of equality and inequality constraints, neq and m respectively, and the number of lower, and upper bounds, nl and nu respectively.

Tables 2 and 3 summarize the results obtained on this set of test problems. The tables report a comparison of the performances of the algorithm with different choices of the parameter N ($N = 1, 3, 5, 7$) in terms of number of external and inner iterations, in the rows with the "ext." and "inn." symbols respectively, and of number of backtracking reductions (the rows denoted by "back").
The case $N = 1$ is the usual monotone case.

Tables 2 and 3 show that the nonmonotone strategies can produce a sensible decrease not only of the number of backtracking reduction, but also of the number of inner iterations. Furthermore, in some cases, also the number of external iterations is reduced, when nonmonotone choices are employed. This fact could be explained by observing that different choices of the parameter N imply different values of the inner tolerance: since the direction Δw_k computed at the step 3 of Algorithm 4.1 depends on the inner tolerance, for different values of N, we obtain different search directions.
Figure 1 depicts the decreasing of the function $\Psi(w)$ defined in (43): the value $\Psi(w_k)$ has been reported in logarithmic scale on the y axis for each iteration of the Algorithm 4.1 applied to the MCP problem lincont. For $N = 3$, $N = 5$ and $N = 7$, a nonmonotone decrease can be observed, and the tolerance of 10^{-8} is reached after 32, 33 and 34 iterations respectively, while in the monotone case ($N = 1$), the same tolerance is satisfied after 46 iterations. A similar behaviour has been observed also for the most part of the MCP and NLP test problems, for example ehl-kost, ehl-def, optcntrl, marine, rosenbr.

On the other side, a too large value of the parameter N could, in some case, produce a degenerate behaviour of the algorithm, as we observed for example in the MCP problem duopoly.

The decrease of the number of iterations and of the number of backtracking steps corresponds to a decreasing of the execution time. For example, in the problem lincont, the execution time of the monotone algorithm is 3.62 seconds: setting $N = 3$ the time is reduced to 1.37 seconds, which is less than one half of the monotone algorithm time. The CPU time for $N = 5$ and $N = 7$ are 1.36 and 1.42 respectively.

A significant reduction has been obtained, for example, also for the test problem ehl-kost, for which we have obtained 0.43, 0.26, 0.29 and 0.34 seconds with $N = 1$, $N = 3$, $N = 5$ and $N = 7$ respectively, for the problem ehl-def (0.23, 0.15, 0.17, 0.21), dtoc6 (0.11, 3.1e-2, 2.3e-2, 2.3e-2), marine (1.68, 1.06, 1.06, 1.11), opt-cont3 (10.3, 18.7, 18.7, 18.7).

Since the execution time related to the other test problems is very small (less than one seconds), we report in the graphs of figures 2 and 3 the ratio between the execution time of the nonmonotone algorithms obtained with the different values of N and the execution time of the monotone algorithm. Thus, the value 1 on the y axis represents the time employed by the monotone algorithm.

In general, the nonmonotone choice improved the execution times, but we observed that, for large values of the parameter N, the number of inner iterations could decrease while the number of external iterations could rise. This leads to an increase of the execution time.

Thus, we can conclude that the nonmonotone strategies are effective in improving the performance of the Algorithm 4.1, but the nonmonotonicity parameter N has to be chosen very carefully.

Acknowledgements. The authors are grateful to the referees for their comments, which helped to improve the presentation of the paper.

Table 1: Test Problems

NLP Problem	Ref.	n	neq	m	nl	nu
harkerp2	[2]	100	0	0	100	0
himmelbk	[2]	24	14	0	24	0
optcdeg2	[2]	295	197	0	197	0
optcdeg3	[2]	295	197	0	197	99
optcntrl	[2]	28	19	1	20	10
aug2dc	[2]	220	96	0	18	0
minsurf	[6]	225	0	0	225	0
marine	[6]	175	152	0	15	0
steering	[6]	294	236	0	61	60
dtoc2	[2]	294	196	0	0	0
dtoc6	[2]	298	149	0	0	0
lukvle8	[11]	300	298	0	0	0
blend	*	24	14	0	24	0
branin	*	2	2	0	2	0
kowalik	*	4	0	0	4	4
osbornea	[2]	5	0	0	5	5
rosenbr	[2]	2	0	0	0	0
hs6	[2]	2	0	1	0	0
mitt105	[13, Ex.5.5] $\alpha = 0.01$, $N = 5$	65	45	0	65	65
mitt305	[13, Ex.4] $\alpha = 0.001$, $N = 5$	70	45	0	25	50
mitt405	[13, Ex.3] $\alpha = 0.001$, $N = 5$	50	25	0	25	50
MCP Problem	**Ref.**	n	neq	m	nl	nu
ehl-kost	[5]	101	0	0	100	0
ehl-def	[5]	101	0	0	100	0
bertsek	[5]	15	0	0	10	0
choi	[5]	13	0	0	0	0
josephy	[5]	4	0	0	4	0
bai-haung	[5]	4900	0	0	4900	0
bratu	[5]	5929	0	0	5625	5625
duopoly	[5]	69	0	0	63	0
ehl-k40	[5]	41	0	0	40	0
hydroc06	[5]	29	0	0	11	0
lincont	[5]	419	0	0	170	0
opt-cont1	[5]	1024	0	0	512	512
opt-cont2	[5]	4096	0	0	2048	2048
opt-cont3	[5]	16384	0	0	8192	8192

*http://scicomp.ewha.ac.kr/netlib/ampl/models/nlmodels/

Table 2: Nonmonotone results in the NLP problems

NLP Problem		N=1	N=3	N=5	N=7
harkerp2	ext.	105	104	104	104
	inn.	471	404	415	438
	back	37	20	14	8
himmelbk	ext.	22	23	27	25
	inn.	114	96	149	110
	back	22	60	83	58
optcdeg2	ext.	-	-	87	75
	inn.	-	-	294	278
	back	-	-	313	289
optcdeg3	ext.	-	100	73	68
	inn.	-	266	158	134
	back	-	315	133	82
optcntrl	ext.	31	25	23	23
	inn.	78	54	45	45
	back	135	70	54	54
aug2dc	ext.	7	7	7	7
	inn.	12	7	7	7
	back	0	0	0	0
minsurf	ext.	5	5	5	5
	inn.	10	8	8	5
	back	0	0	0	0
marine	ext.	94	68	67	72
	inn.	436	296	284	287
	back	473	327	287	309
steering	ext.	11	-	-	-
	inn.	30	-	-	-
	back	37	-	-	-
dtoc2	ext.	7	7	7	7
	inn.	11	8	8	8
	back	1	1	1	1
dtoc6	ext.	21	9	9	9
	inn.	21	9	9	9
	back	39	1	0	0
lukvle8	ext.	-	20	21	25
	inn.	-	418	365	455
	back	-	34	34	34
blend	ext.	23	20	19	19
	inn.	79	112	82	75
	back	225	98	72	72
branin	ext.	8	8	8	8
	inn.	8	8	8	8
	back	11	2	2	2
kowalik	ext.	37	11	22	20
	inn.	37	11	22	20
	back	149	1	14	6

NLP Problem		N=1	N=3	N=5	N=7
osborne1	ext.	121	16	16	16
	inn.	121	16	16	16
	back	443	0	0	0
rosenbr	ext.	180	14	9	9
	inn.	180	14	9	9
	back	1121	4	2	2
hs6	ext.	8	7	7	7
	inn.	8	7	7	7
	back	14	11	11	11
mitt105	ext.	11	9	8	8
	inn.	19	13	9	9
	back	13	4	0	0
mitt305	ext.	31	24	24	20
	inn.	71	47	46	37
	back	195	105	102	68
mitt405	ext.	32	26	19	19
	inn.	77	54	39	39
	back	227	144	81	81

$-$ the algorithm does not converge

References

[1] S. Bonettini (2005). *A Nonmonotone Inexact Newton Method*, Optim. Meth. Software, **20**, 4-5, 475–491 .

[2] I. Bongartz, A.R. Conn, N. Gould and Ph. L. Toint (1995). *CUTE: Constrained and Unconstrained Testing Environnment*, ACM Transactions on Mathematical Software, **21**, 123–160.

[3] F. H. Clarke (1983). *Optimization and Nonsmooth Analisys*, John Wiley & Songs, New York,

[4] R. S. Dembo, S. C. Eisenstat and T. Steihaug (1982). *Inexact Newton methods*, SIAM Journal on Numerical Analysis, **19**, 400–408.

Table 3: Nonmonotone results in the MCP problems

MCP Problem		N=1	N=3	N=5	N=7
ehl-kost	ext.	14	12	14	16
	inn.	104	50	50	48
	back	17	0	0	0
ehl-def	ext.	14	12	14	16
	inn.	103	50	50	48
	back	17	0	0	0
bertsek	ext.	6	6	6	6
	inn.	8	7	6	6
	back	0	0	0	0
choi	ext.	5	5	5	5
	inn.	5	5	5	5
	back	0	0	0	0
josephy	ext.	6	6	7	7
	inn.	8	7	7	7
	back	2	2	2	2
bai-haung	ext.	6	6	6	6
	inn.	13	9	9	9
	back	0	0	0	0
bratu	ext.	5	5	5	5
	inn.	10	6	6	6
	back	0	0	0	0
duopoly	ext.	44	38	48	$*$
	inn.	135	127	140	$*$
	back	225	158	81	$*$
ehl-k40	ext.	$*$	$*$	203	333
	inn.	$*$	$*$	1292	2056
	back	$*$	$*$	2252	3230
hydroc06	ext.	5	5	5	5
	inn.	8	6	6	6
	back	1	1	1	1
lincont	ext.	46	32	33	34
	inn.	385	165	144	170
	back	220	89	77	73
opt-cont1	ext.	10	10	10	10
	inn.	49	33	21	19
	back	0	0	0	0
opt-cont2	ext.	9	9	9	11
	inn.	101	52	42	42
	back	0	0	0	0
opt-cont3	ext.	8	8	8	8
	inn.	22	15	14	14
	back	0	0	0	0

$*$ maximum number of backtracking reductions reached

Figure 1: Decrease of the merit function

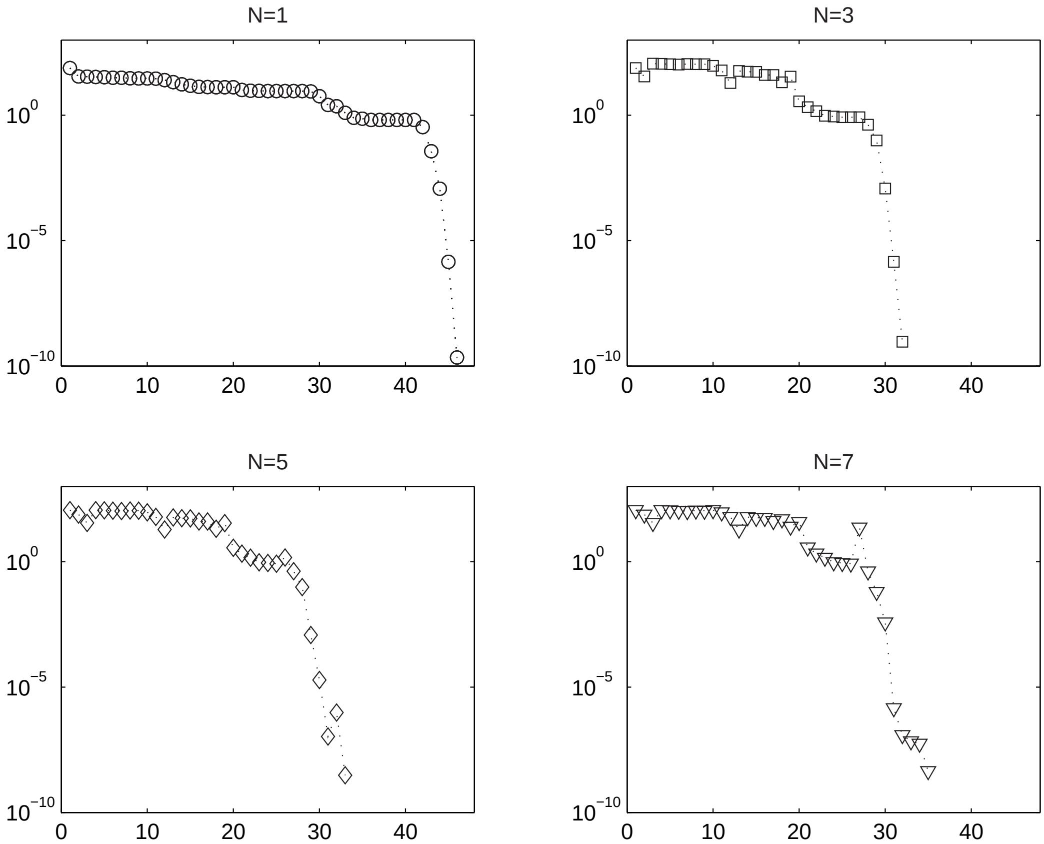


Figure 2: Time comparison for the MCP problems

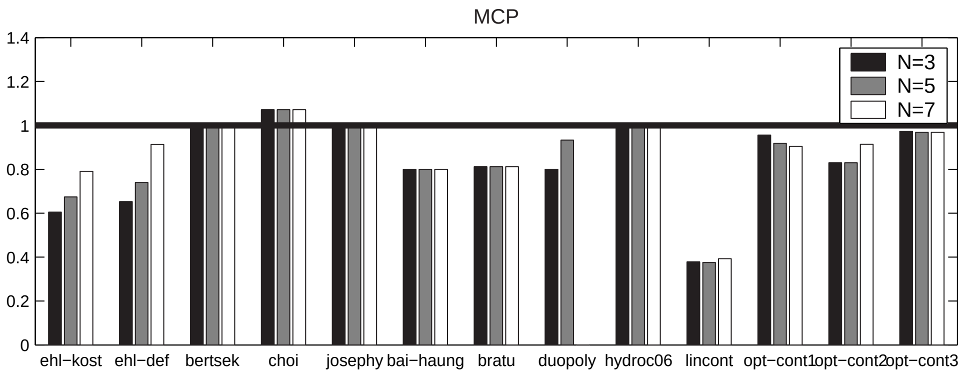
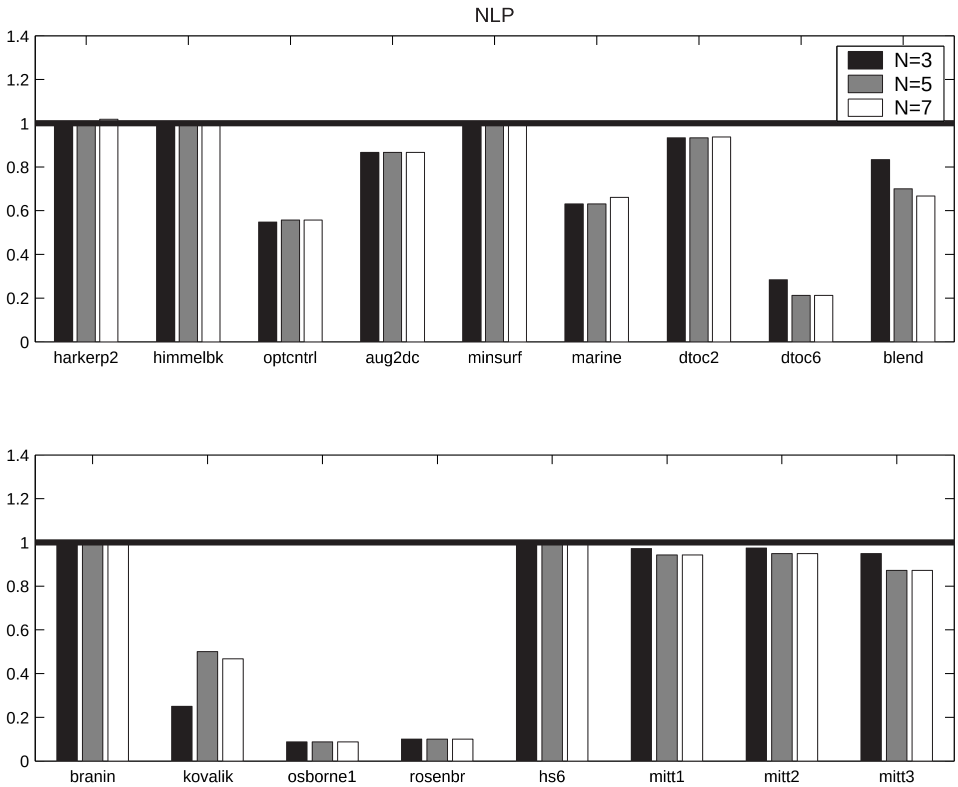


Figure 3: Time comparison for the NLP problems



[5] S. P. Dirkse and M. C. Ferris (1995). *A collection of nonlinear mixed complementarity problems*, Optimization Methods and Software, **5**, 123–156.

[6] E.D. Dolan, J. J. Moré and T. S. Munson (2004). *Benchmarking optimization software with COPS 3.0*, Technical Report ANL/MCS-TM-273, Argonne National Laboratory, Illinois, USA.

[7] S. C. Eisenstat and H. F. Walker (1994). *Globally convergent Inexact Newton methods*, SIAM Journal on Optimization, **4**, 393–422.

[8] F. Facchinei, A. Fischer and C. Kanzow (1996). *Inexact Newton methods for semismooth equations with applications to variational inequality problems*, in G. Di Pillo and F. Giannessi (eds.), Nonlinear Optimization and Applications, Plenum Press, New York 1996, 125-139.

[9] A. Fischer (1992). *A special Newton-type optimization method*, Optimization, **24**, 269–284.

[10] L. Grippo, F. Lampariello and S. Lucidi (1986). *A Nonmonotone line search technique for Newton's method*, SIAM Journal on Numerical Analysis **23**, 707–716.

[11] L. Lukšan and J. Vlček (1999). *Sparse and partially separable test problems for unconstrained and equality constrained optimization*, Technical Report 767, Institute of Computer Science, Academy of Science of the Czech Republic.

[12] J. M. Martinez and L. Qi (1995). *Inexact Newton methods for solving nonsmooth equations*, J. Comput. Appl. Math., **60**, 127–145.

[13] H. D. Mittelmann and H. Maurer (1999). *Optimization techiques for solving elliptic control problems with control and state constraints: Part 1. Boundary control*, Computational Optimization and Applications, **16**, 29–55.

[14] J. M. Ortega and W. C. Rheimboldt (1970). *Iterative solution of nonlinear equations in several variables*, Academic Press, New York.

[15] C. C. Paige and M. A. Saunders (1982). *LSQR: An Algorithm for Sparce Linear Equations and Sparse Least Squares*, ACM Transactions on Mathematical Software, **8**, 1, 43–71.

[16] J. S. Pang and L. Qi (1993). *Nonsmooth equations: Motivations and algorithms*, SIAM J. Optim., **3**, 443–465.

[17] L. Qi (1993). *A convergence analysis of some algorithms for solving nonsmooth equations*, Mathematical of Operator Research, **18**, 227–244.

[18] L. Qi, J. Sun (1993). *A nonsmooth version of Newton method*, Mathematical Programming, **58**, 353–367.

[19] W.C. Rheinboldt (1998). Methods for Solving Systems of Nonlinear Equations, Second Edition, SIAM, Philadelphia.

[20] V. Ruggiero and F. Tinti (2005). *A preconditioner for solving large-scale variational inequality problems by a semismooth inexact approach*, Tech. Rep. n. 69, Dipartimento di Matematica, Università di Modena e Reggio Emilia, Modena.