Journal of Computational and Applied Mathematics 233 (2010) 2312–2321

Contents lists available at ScienceDirect

Journal of Computational and Applied Mathematics

journal homepage: www.elsevier.com/locate/cam

A nonmonotone smoothing-type algorithm for solving a system of equalities and inequalities[☆]

Ying Zhang, Zheng-Hai Huang [*]

Department of Mathematics, School of Science, Tianjin University, Tianjin 300072, PR China

ARTICLE INFO

Article history:
Received 28 March 2009
Received in revised form 8 October 2009

MSC:
90C33
65K05
65H05

Keywords:
System of equalities and inequalities
Smoothing-type algorithm
Nonmonotone line search
Global convergence
Local superlinear convergence

ABSTRACT

In this paper, we investigate a smoothing-type algorithm with a nonmonotone line search for solving a system of equalities and inequalities. We prove that the nonmonotone algorithm is globally and locally superlinearly convergent under suitable assumptions. The preliminary numerical results are reported.

1. Introduction

Smoothing-type algorithms have been successfully applied to solve various optimization problems, including linear programs, complementarity and variational inequality problems, mathematical programming with equilibrium problems, optimization problems over symmetric cones, and so on. Recently, a smoothing-type algorithm was developed for solving the system of inequalities [1]. The preliminary numerical results given in [1] show that this class of algorithms is effective for solving the system of inequalities. Some other iteration methods for solving the system of inequalities can be found in [2–5].

Most existing smoothing-type algorithms were designed on the basis of a monotone line search. In order to achieve better computational results, however, the nonmonotone line search technique was adopted in the numerical computations of smoothing-type algorithms (see, for example, [6,7]). Recently, the theoretical analysis of some nonmonotone smoothing-type algorithms was given in [8,9]. It is well known that the nonmonotone line search scheme can improve the likelihood of finding a global optimal solution and convergence speed in cases where the function involved is highly nonconvex and has a valley in a small neighborhood of some point. Various nonmonotone line search schemes have been proposed in the many iteration methods (see, for example, [10–12]).

In this paper, we consider the following system of equalities and inequalities:

$$\begin{cases} f_{\mathcal{I}}(x) \leq 0, \\ f_{\mathcal{E}}(x) = 0, \end{cases} \tag{1.1}$$

☆ This work was partially supported by the National Natural Science Foundation of China (Grant No. 10871144) and the Natural Science Foundation of Tianjin (Grant No. 07JCYBJC05200).

* Corresponding author. Tel.: +86 22 27402850; fax: +86 22 27403425.
E-mail addresses: zhangyhit@yahoo.com.cn (Y. Zhang), huangzhenghai@tju.edu.cn (Z.-H. Huang).

0377-0427/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2009.10.016

where $\mathcal{I} = \{1, \dots, m\}$ and $\mathcal{E} = \{m+1, \dots, n\}$. Define $f(x) := [f_1(x), \dots, f_n(x)]^{\mathrm{T}}$ with $f_i : \Re^n \to \Re$ for any $i \in \{1, \dots, n\}$. Throughout this paper, we assume that f is continuously differentiable. We will extend the smoothing-type algorithm given in [1] to solve (1.1). There are two main differences between [1] and this paper. One is that the problem that we are concerned with is more general than the one discussed in [1] since $\mathcal{E} = \emptyset$ in [1]. Another is that, instead of the monotone linear search used in [1], we use a nonmonotone line search in the algorithm of this paper. Under suitable assumptions, we show that the nonmonotone smoothing-type algorithm is globally and locally superlinearly convergent. We also report some preliminary numerical results, which demonstrate that the algorithm is effective for solving (1.1).

The rest of this paper is organized as follows. In Section 2, we reformulate (1.1) as a system of smooth equations. In Section 3, a smoothing-type algorithm is presented for solving (1.1); and some basic properties of the algorithm are discussed. In Sections 4 and 5, we investigate the global and local convergence of the algorithm, respectively. Section 6 gives some further discussions on applications of the algorithm and some preliminary numerical results. The final remarks are given in Section 7.

Throughout this paper, we use the following notation. The nonnegative (resp., positive) orthant in $\Re^n$ will be denoted by $\Re^n_+$ (resp., $\Re^n_{++}$). I_n denotes the $n \times n$ identity matrix. For $x \in \Re^n$, $\|x\|$ denotes the 2-norm of x. For any vectors $u, v \in \Re^n$, we write $(u^{\mathrm{T}}, v^{\mathrm{T}})^{\mathrm{T}}$ as (u, v) for simplicity. We use $\mathcal{J}$ to denote the set of all nonnegative integers, i.e., $\mathcal{J} := \{0, 1, 2, \dots\}$. For any $(\mu, x, s), (\mu_k, x^k, s^k) \in \Re_+ \times \Re^n \times \Re^m$, we always use the following notation throughout this paper unless stated otherwise: $z := (\mu, x, s)$ and $z^k := (\mu_k, x^k, s^k)$. For any $\xi, \rho \in \Re_+$, $\xi = O(\rho)$ ($\xi = o(\rho)$) means $\limsup_{\rho \to 0} \frac{\xi}{\rho} < +\infty$ ($\limsup_{\rho \to 0} \frac{\xi}{\rho} = 0$).

2. Smooth reformulation of (1.1)

In this section, we reformulate (1.1) as a system of smoothing equations in a similar way to that described in [1].

For any $x \in \Re^l$, we define

$$x_+ := (\max\{0, x_1\}, \dots, \max\{0, x_l\})^{\mathrm{T}}.$$

Then, (1.1) is equivalent to the following system of equations:

$$\begin{cases} f_{\mathcal{I}}(x)_+ = 0, \\ f_{\mathcal{E}}(x) = 0. \end{cases} \tag{2.1}$$

Since the function in (2.1) is nonsmooth, the classical Newton methods cannot be directly applied to solve (2.1). In order to make (2.1) solvable by the classical Newton-type methods, we will use the similar smoothing technique investigated in [1].

The following function was discussed in [1]:

$$\phi(\mu, a) := \begin{cases} a & \text{if } a \geq \mu, \\ \dfrac{(\mu + a)^2}{4\mu} & \text{if } -\mu < a < \mu, \\ 0 & \text{if } a \leq -\mu; \end{cases} \tag{2.2}$$

and the following results were given in [1, Proposition 1.1]:

Proposition 2.1. *For any $(\mu, a) \in \Re^2$, we have the following results.*

(i) $\phi(\cdot, \cdot)$ *is continuously differentiable at any* $(\mu, a) \in \Re^2$ *with* $\mu > 0$*;*
(ii) $\phi(0, a) = a_+$*;*
(iii) $\frac{\partial \phi(\mu, a)}{\partial a} \geq 0$ *at any* $(\mu, a) \in \Re^2$ *with* $\mu \geq 0$*.*

Define

$$F(z) := \begin{bmatrix} f_{\mathcal{I}}(x) - s \\ f_{\mathcal{E}}(x) \\ \Phi(\mu, s) \end{bmatrix} \quad \text{with } \Phi(\mu, s) := \begin{pmatrix} \phi(\mu, s_1) \\ \vdots \\ \phi(\mu, s_m) \end{pmatrix}. \tag{2.3}$$

Then, by Proposition 2.1 (ii) we have

$$F(z) = 0 \text{ and } \mu = 0 \quad \Longleftrightarrow \quad s = f_{\mathcal{I}}(x),\ s_+ = 0,\ f_{\mathcal{E}}(x) = 0.$$

This, together with Proposition 2.1(i), indicates that one can solve (1.1) by applying Newton-type methods to solve $F(z) = 0$ and make $\mu \downarrow 0$. Furthermore, we define a function $H : \Re^{1+n+m} \to \Re^{1+n+m}$ by

$$H(z) := \begin{pmatrix} \mu \\ f_{\mathcal{I}}(x) - s + \mu x_{\mathcal{I}} \\ f_{\mathcal{E}}(x) + \mu x_{\mathcal{E}} \\ \Phi(\mu, s) + \mu s \end{pmatrix}, \tag{2.4}$$

where $x_{\mathcal{I}} = (x_1, x_2, \ldots, x_m)^{\mathrm{T}}$, $x_{\mathcal{E}} = (x_{m+1}, x_{m+2}, \ldots, x_n)^{\mathrm{T}}$, $s \in \Re^m$, $x := (x_{\mathcal{I}}, x_{\mathcal{E}}) \in \Re^n$, and functions ϕ and Φ are defined by (2.2) and (2.3), respectively. Thereby, it is obvious that if $H(z) = 0$, then $\mu = 0$ and x solves (1.1). It is not difficult to see that, for any $z \in \Re_{++} \times \Re^n \times \Re^m$, the function H is continuously differentiable. Let H' denote the Jacobian of the function H; then for any $z \in \Re_{++} \times \Re^n \times \Re^m$,

$$H'(z) = \begin{bmatrix} 1 & 0_n & 0_m \\ x_{\mathcal{I}} & f'_{\mathcal{I}}(x) + \mu U & -I_m \\ x_{\mathcal{E}} & f'_{\mathcal{E}}(x) + \mu V & 0_{(n-m)\times m} \\ s + \Phi'_\mu(\mu, s) & 0_{m\times n} & \Phi'_s(\mu, s) + \mu I_m \end{bmatrix}, \tag{2.5}$$

where $U := [I_m \ 0_{m\times(n-m)}]$ and $V := [0_{(n-m)\times m} \ I_{n-m}]$. Here, we use 0_l to denote the l-dimensional zero vector and $0_{l\times q}$ to denote the $l \times q$ zero matrix for any positive integers l and q. Thus, we might apply some Newton-type methods to solve the system of smooth equations $H(z) = 0$ at each iteration and make $\mu > 0$ and $H(z) \to 0$ so that a solution of (1.1) can be found.

3. A smoothing-type algorithm

In this section, we propose a smoothing-type algorithm with a nonmonotone line search. Some basic properties are given. In particular, we show that the algorithm is well defined. We will use the following function: $\Psi(z) := \|H(z)\|^2$.

Algorithm 3.1 (*A Nonmonotone Smoothing-Type Algorithm*).

Step 0 Choose $\delta \in (0, 1)$, $\sigma \in (0, 1/2)$, $\beta > 0$. Take $\tau \in (0, 1)$ such that $\tau\beta < 1$. Let $\mu_0 = \beta$ and $(x^0, s^0) \in \Re^{n+m}$ be an arbitrary vector. Set $z^0 := (\mu_0, x^0, s^0)$. Take $\mathrm{e}^0 := (1, 0, \ldots, 0) \in \Re^{1+n+m}$, $R_0 := \|H(z^0)\|^2 = \Psi(z^0)$ and $Q_0 := 1$. Choose $\eta_{\min}$ and $\eta_{\max}$ such that $0 \le \eta_{\min} \le \eta_{\max} < 1$. Set $\theta(z^0) := \tau \min\{1, \Psi(z^0)\}$ and $k := 0$.

Step 1 If $\|H(z^k)\| = 0$, stop.

Step 2 Compute $\Delta z^k := (\Delta\mu_k, \Delta x^k, \Delta s^k) \in \Re \times \Re^n \times \Re^m$ by using

$$H'(z^k)\Delta z^k = -H(z^k) + \beta\theta(z^k)\mathrm{e}^0. \tag{3.1}$$

Step 3 Let α_k be the maximum of the values $1, \delta, \delta^2, \ldots$ such that

$$\Psi(z^k + \alpha_k \Delta z^k) \le [1 - 2\sigma(1 - \tau\beta)\alpha_k]R_k. \tag{3.2}$$

Step 4 Set $z^{k+1} := z^k + \alpha_k \Delta z^k$. If $\|H(z^{k+1})\| = 0$, stop.

Step 5 Choose $\eta_k \in [\eta_{\min}, \eta_{\max}]$. Set

$$\begin{aligned} &Q_{k+1} := \eta_k Q_k + 1, \\ &\theta(z^{k+1}) := \min\{\tau, \tau\Psi(z^k), \theta(z^k)\}, \\ &R_{k+1} := (\eta_k Q_k R_k + \Psi(z^{k+1}))/Q_{k+1}, \end{aligned} \tag{3.3}$$

and $k := k + 1$. Go to Step 2.

In Algorithm 3.1, a nonmonotone line search technique, introduced in [11], is adopted. It is easy to see that R_{k+1} is a convex combination of R_k and $\Psi(z^{k+1})$. Since $R_0 = \Psi(z^0)$, it follows that R_k is a convex combination of the function values $\Psi(z^0), \Psi(z^1), \ldots, \Psi(z^k)$. The choice of η_k controls the degree of nonmonotonicity. If $\eta_k = 0$ for every k, then the line search is the usual monotone Armijo line search.

Define $f'(x) := [f'_{\mathcal{I}}(x)^{\mathrm{T}}, f'_{\mathcal{E}}(x)^{\mathrm{T}}]^{\mathrm{T}}$. We will use the following assumption.

Assumption 3.1. $f'(x) + \mu I_n$ is invertible for any $x \in \Re^n$ and $\mu \in \Re_{++}$.

Some basic results involving Algorithm 3.1 are included in the following remark.

Remark 3.1. Let the sequence $\{R_k\}$ and $\{z^k\}$ be generated by Algorithm 3.1.

(i) We have that the sequence $\{R_k\}$ is monotonically decreasing.

In fact, by using (3.2) and the definition of R_k in (3.3), it follows that for any $k \in \mathcal{J}$,

$$R_{k+1} \le \frac{\eta_k Q_k R_k + R_k - 2\sigma(1-\tau\beta)\alpha_k R_k}{Q_{k+1}} = R_k - \frac{2\sigma(1-\tau\beta)\alpha_k R_k}{Q_{k+1}} \le R_k, \tag{3.4}$$

which implies that $\{R_k\}$ is monotonically decreasing.

(ii) We have that $\Psi(z^k) \leq R_k$ for all $k \in \mathcal{J}$.

In fact, this can be obtained by an inductive method. Firstly, it is evident from the choice of $\mathfrak{R}_0$ that the result holds when $k = 0$. Secondly, if we assume that the result holds when $k = l$, then we only need to show that the result holds when $k = l + 1$. Note that

$$\begin{aligned} \Psi(z^{l+1}) &= Q_{l+1}R_{l+1} - \eta_l Q_l R_l \leq Q_{l+1}R_{l+1} - \eta_l Q_l R_{l+1} \\ &= (Q_{l+1} - \eta_l Q_l)R_{l+1} = R_{l+1}, \end{aligned}$$

where the first equality follows from the definition of R_{l+1} in (3.3); the first inequality from the above result (i); and the last equality from the definition of Q_{l+1} in (3.3).

(iii) We have that the sequence $\{\theta(z^k)\}$ is monotonically decreasing.

In fact, this result holds directly from the above results (i) and (ii).

(iv) We have that $\beta\theta(z^k) \leq \mu_k$ for all $k \in \mathcal{J}$.

In fact, from the choice of the starting point it follows that $\beta\theta(z_0) \leq \mu_0$. Next, we assume that $\beta\theta(z_l) \leq \mu_l$ for some index $l \in \mathcal{J}$. Then

$$\begin{aligned} \mu_{l+1} - \beta\theta(z^{l+1}) &= \mu_l + \alpha_l \Delta\mu_l - \beta\theta(z^{l+1}) \\ &= (1 - \alpha_l)\mu_l + \beta\alpha_l\theta(z^l) - \beta\theta(z^{l+1}) \\ &\geq \beta(\theta(z^l) - \theta(z^{l+1})) \geq 0, \end{aligned}$$

where the second equality follows from the first equation of (3.1) and $\mu_{l+1} = \mu_l + \alpha_l \Delta\mu_l$; the first inequality holds from $\beta\theta(z^l) \leq \mu_l$; and the second inequality from the above result (iii). Thus, by using the inductive method we obtain the desired result.

(v) We have that $\mu_k > 0$ for all $k \in \mathcal{J}$ and the sequence $\{\mu_k\}$ is monotonically decreasing.

In fact, from the first equation of (3.1) it follows that

$$\mu_{k+1} = \mu_k + \alpha_k \Delta\mu_k = (1 - \alpha_k)\mu_k + \beta\alpha_k\theta(z^k) > 0, \tag{3.5}$$

which indicates that $\mu_k > 0$ for all $k \in \mathcal{J}$. Combining (3.5) and the above result (iv), we have that for any $k \in \mathcal{J}$,

$$\mu_{k+1} = (1 - \alpha_k)\mu_k + \beta\alpha_k\theta(z^k) \leq (1 - \alpha_k)\mu_k + \alpha_k\mu_k = \mu_k,$$

which implies that the sequence $\{\mu_k\}$ is monotonically decreasing.

Theorem 3.1. *Suppose that f is a continuously differentiable function and Assumption 3.1 is satisfied. Then Algorithm 3.1 is well defined.*

Proof. Firstly, we show that the line search (3.2) is well defined. Let $L^k(\alpha) := \Psi(z^k + \alpha\Delta z^k) - \Psi(z^k) - \alpha\Psi'(z^k)\Delta z^k$; then by (3.1),

$$\begin{aligned} \Psi(z^k + \alpha\Delta z^k) &= L^k(\alpha) + \Psi(z^k) + \alpha\Psi'(z^k)\Delta z^k \\ &= L^k(\alpha) + \Psi(z^k) + 2\alpha H(z^k)^{\mathrm{T}}(-H(z^k) + \beta\theta(z^k))\mathrm{e}^0 \\ &\leq L^k(\alpha) + (1 - 2\alpha)\Psi(z^k) + 2\alpha\beta\theta(z^k)\|H(z^k)\|. \end{aligned}$$

On one hand, if $\Psi(z^k) \leq 1$, then $\|H(z^k)\| \leq 1$. So we can obtain that $\theta(z^k)\|H(z^k)\| \leq \tau\Psi(z^k)\|H(z^k)\| \leq \tau\Psi(z^k)$. On the other hand, if $\Psi(z^k) > 1$, then $\Psi(z^k) = \|H(z^k)\|^2 \geq H(z^k)$. So we can obtain that $\theta(z^k)\|H(z^k)\| \leq \tau\|H(z^k)\| \leq \tau\Psi(z^k)$. Thus,

$$\begin{aligned} \Psi(z^k + \alpha\Delta z^k) &\leq L^k(\alpha) + (1 - 2\alpha)\Psi(z^k) + 2\alpha\beta\tau\Psi(z^k) \\ &= L^k(\alpha) + [1 - 2(1 - \tau\beta)\alpha]\Psi(z^k) \\ &\leq L^k(\alpha) + [1 - 2(1 - \tau\beta)\alpha]R_k. \end{aligned} \tag{3.6}$$

Since the function H is continuously differentiable for any $z \in \mathfrak{R} \times \mathfrak{R}^n \times \mathfrak{R}^m$ with $\mu > 0$, it follows from (3.6) and (v) that $L^k(\alpha) = o(\alpha)$ for all $k \in \mathcal{J}$. Thus, the desired result holds by $\Psi(z^k) \leq R_k$ for all $k \in \mathcal{J}$.

Secondly, we show that Step 2 is well defined. For any square matrix A, we use $\det(A)$ to denote the determinant of A. It is easy to see from (2.5) that $\det(H'(z)) = \det(f'(x) + \mu I_n) \cdot \det(\Phi'_s(\mu, s) + \mu I_m)$ for any $z \in \mathfrak{R}_{++} \times \mathfrak{R}^n \times \mathfrak{R}^m$. Furthermore, we know from Proposition 2.1(iii) that $\Phi'_s(\mu, s)$ is positive semi-definite. Thus, by Assumption 3.1 we obtain that $H'(z)$ is nonsingular for any $z \in \mathfrak{R}^{1+n+m}$ with $\mu > 0$. This, together with the result (v), implies that the system of equations (3.1) is solvable, i.e. Step 2 is well defined.

Therefore, Algorithm 3.1 is well defined. □

4. Global convergence of Algorithm 3.1

The following assumption was introduced in [1].

Assumption 4.1. For an arbitrary sequence $\{(\mu_k, x^k)\}$ with $\lim_{k\to\infty} \| x^k \| = +\infty$ and the sequence $\{\mu_k\} \subset \Re_+$ bounded, then either:

(i) there is at least an index i_0 such that $\limsup_{k\to\infty}\{f_{i_0}(x^k) + \mu_k x^k_{i_0}\} = +\infty$; or

(ii) there is at least an index i_0 such that $\limsup_{k\to\infty}\{\mu_k(f_{i_0}(x^k) + \mu_k x^k_{i_0})\} = -\infty$.

It can be seen from [1] that many functions satisfy Assumption 4.1.

The global convergence of Algorithm 3.1 is stated as follows.

Theorem 4.1. *Suppose that f is a continuously differentiable function and Assumptions 3.1 and 4.1 are satisfied. Then the infinite sequence $\{z^k\}$ generated by Algorithm 3.1 is bounded; and any accumulation point of $\{x^k\}$ is a solution of (1.1).*

Proof. We divide the proof into the following two parts.

Part 1. We show that the sequence $\{z^k\}$ is bounded. By Remark 3.1(v) we know that the sequence $\{\mu_k\}$ is bounded, and hence, we only need to show that $\{(x^k, s^k)\}$ is bounded. In the following, by assuming that $\{x^k\}$ is unbounded, we will derive a contradiction. By (2.4) and the definition of Ψ it follows that

$$\Psi(z^k) = \mu_k^2 + \| f_{\mathcal{I}}(x^k) - s^k + \mu_k x^k_{\mathcal{I}} \|^2 + \| f_{\mathcal{E}}(x^k) + \mu_k x^k_{\mathcal{E}} \|^2 + \| \Phi(\mu_k, s^k) + \mu_k s^k \|^2 . \tag{4.1}$$

Since the sequence $\{R_k\}$ is monotonically decreasing and $R_k > 0$, it follows that the sequence $\{R_k\}$ is bounded. Then, by Remark 3.1(ii) we get that $\{\Psi(z^k)\}$ is bounded. Thus, from (4.1) we obtain that

$$\{f_{\mathcal{I}}(x^k) - s^k + \mu_k x^k_{\mathcal{I}}\}, \quad \{f_{\mathcal{E}}(x^k) + \mu_k x^k_{\mathcal{E}}\}, \quad \{\Phi(\mu_k, s^k) + \mu_k s^k\} \quad \text{are bounded.} \tag{4.2}$$

For any $k \in \mathcal{J}$, let $h(z^k) := s^k - f_{\mathcal{I}}(x^k) - \mu_k x^k_{\mathcal{I}}$; then $\{h(z^k)\}$ is bounded and

$$s^k = h(z^k) + f_{\mathcal{I}}(x^k) + \mu_k x^k_{\mathcal{I}}. \tag{4.3}$$

Since $\{f_{\mathcal{E}}(x^k) + \mu_k x^k_{\mathcal{E}}\}$ is bounded by (4.2), it follows from Assumption 4.1 that either:

(I) there is at least an index $i_0 \in \mathcal{I}$ such that $\limsup_{k\to\infty}\{f_{i_0}(x^k) + \mu_k x^k_{i_0}\} = +\infty$; or

(II) there is at least an index $i_0 \in \mathcal{I}$ such that $\limsup_{k\to\infty}\{\mu_k(f_{i_0}(x^k) + \mu_k x^k_{i_0})\} = -\infty$.

In the following, we consider these two cases, separately.

- If the above result (I) holds, then by (4.3) we have

  $$\limsup_{k\to\infty} s^k_{i_0} = \limsup_{k\to\infty}\{h_{i_0}(z^k) + f_{i_0}(x^k) + \mu_k x^k_{i_0}\} = +\infty.$$

  Furthermore, by using definitions of ϕ and Φ we have

  $$\limsup_{k\to\infty}\{\Phi_{i_0}(\mu_k, s^k) + \mu_k s^k_{i_0}\} = \limsup_{k\to\infty}\{s^k_{i_0} + \mu_k s^k_{i_0}\} = +\infty,$$

  which indicates that $\{\Phi(z^k) + \mu_k s^k\}$ is unbounded. This contradicts (4.2).
- If the above result (II) holds, then $\limsup_{k\to\infty}\{f_{i_0}(x^k) + \mu_k x^k_{i_0}\} = -\infty$ since $\{\mu_k\}$ is a nonnegative and bounded sequence. Thus, by (4.3) we have $\limsup_{k\to\infty} s^k_{i_0} = -\infty$. Furthermore,

  $$\limsup_{k\to\infty}\{\Phi_{i_0}(\mu_k, s^k) + \mu_k s^k_{i_0}\} = \limsup_{k\to\infty}\{\mu_k s^k_{i_0}\} = -\infty,$$

  which implies that $\{\Phi(z^k) + \mu_k s^k\}$ is unbounded. This contradicts (4.2).

  Therefore, the sequence $\{x^k\}$ is bounded.

Since sequences $\{\mu_k\}$, $\{x^k\}$, and $\{h(z^k)\}$ are bounded and the function f is continuous, by (4.3) we further obtain that the sequence $\{s^k\}$ is bounded. Therefore, the sequence $\{z^k\}$ is bounded.

Part 2. We prove that any accumulation point of $\{x^k\}$ generated by Algorithm 3.1 is a solution of (1.1). By Remark 3.1(i), we know that the sequence $\{R_k\}$ is nonnegative and monotone decreasing, and hence, it is convergent. From Remark 3.1(i) and (ii) we have

$$0 \le \|H(z^k)\|^2 = \Psi(z^k) \le R_k \le R_{k-1} \le R_0, \tag{4.4}$$

so the sequence $\{R_k\}$ converges and both sequences $\{\Psi(z^k)\}$ and $\{\|H(z^k)\|\}$ are bounded. In addition, by the first result of this theorem, we obtain that the sequence $\{z^k\}$ is bounded, and hence, it has at least a subsequence which is convergent. We denote this subsequence by $\{z^k\}$ where $k \in \bar{\mathcal{J}} \subseteq \mathcal{J}$. Thus, there exists a point $z^* = (\mu_*, x^*, s^*) \in \Re_{++} \times \Re^n \times \Re^m$ such that $\lim_{\bar{\mathcal{J}}\ni k\to\infty} z^k = z^*$, and by continuity of the function H, we get $\lim_{\bar{\mathcal{J}}\ni k\to\infty} \|H(z^k)\| = \|H(z^*)\|$. Define $R^* := \lim_{\bar{\mathcal{J}}\ni k\to\infty} R_k$. If $R^* = 0$, then $\|H(z^*)\| = 0$, and hence, x^* is a solution of (1.1). In the following, we assume that $R^* > 0$ and $\|H(z^*)\| > 0$, and then derive a contradiction.

- Suppose that $\alpha_k \geq \alpha_* > 0$ for all $k \in \bar{\mathcal{J}}$, where α_* is a fixed constant. On one hand, from (3.4) it follows that $R_{k+1} \leq R_k - \frac{2\sigma(1-\tau\beta)\alpha_*}{Q_{k+1}} R_k$ for any $k \in \bar{\mathcal{J}}$. Since $\{R_k\}$ is bounded, we further obtain that

$$\sum_{k=0}^{\infty} \frac{2\sigma(1-\tau\beta)\alpha_*}{Q_{k+1}} R_k < \infty. \tag{4.5}$$

On the other hand, by the definition of Q_k and the fact that $\eta_{\max} \in [0, 1)$ given in the algorithm, we have

$$Q_{k+1} = 1 + \sum_{i=0}^{k} \prod_{j=0}^{i} \eta_{k-j} \leq 1 + \sum_{i=0}^{k} \eta_{\max}^{i+1} \leq \sum_{i=0}^{\infty} \eta_{\max}^{i} = \frac{1}{1-\eta_{\max}}$$

for any $k \in \bar{\mathcal{J}}$. This, together with (4.5), implies that $\lim_{\bar{\mathcal{J}} \ni k \to \infty} R_k = 0$, which contradicts $R^* > 0$.
- Suppose that $\lim_{\bar{\mathcal{J}} \ni k \to \infty} \alpha_k = 0$. Then, the step size $\hat{\alpha}_k := \alpha_k/\delta$ does not satisfy the line search criterion (3.2) for any sufficiently large $k \in \bar{\mathcal{J}}$, i.e.,

$$\|H(z^k + \hat{\alpha}_k \Delta z^k)\|^2 = \Psi(z^k + \hat{\alpha}_k \Delta z^k) > [1 - 2\sigma(1-\tau\beta)\hat{\alpha}_k] R_k \tag{4.6}$$

holds for any sufficiently large $k \in \bar{\mathcal{J}}$. It is easy to see that the following results hold.

(a) $\mu_* > 0$. This can be easily obtained by Remark 3.1(v).

(b) $\{\Delta z^k\}_{k \in \bar{\mathcal{J}}}$ is convergent. In fact, since $\mu_* > 0$, it follows that $H'(z^k)$ is an invertible continuously linear operator for any $k \in \bar{\mathcal{J}}$. Thus, by using (3.1) we may obtain the desired result. Define $\Delta z^* := \lim_{\bar{\mathcal{J}} \ni k \to \infty} \Delta z^k$.

(c) $R^* = \Psi(z^*)$. In fact, from (4.4) it follows that $R^* \geq \Psi(z^*)$; and from (4.6) it follows that $R^* \leq \Psi(z^*)$.

By (4.6), we have that $\Psi(z^k + \hat{\alpha}_k \Delta z^k) > [1 - 2\sigma(1-\tau\beta)\hat{\alpha}_k] R_k$. Since $R_k \geq \Psi(z^k)$, the above inequality becomes that

$$\{\Psi(z^k + \hat{\alpha}_k \Delta z^k) - \Psi(z^k)\}/\hat{\alpha}_k > -2\sigma(1-\tau\beta) R_k. \tag{4.7}$$

By the above result (a) we know that $\|H(\cdot)\|$ is continuously differentiable at z^*, and so is $\Psi(\cdot)$. Hence, by taking the limit for (4.7) and using the above result (b), we have

$$2H(z^*)^{\mathrm{T}} H'(z^*) \Delta z^* \geq -2\sigma(1-\tau\beta) R^*. \tag{4.8}$$

In addition, by (3.1) and the above results (b) and (c), we have

$$\begin{aligned} 2H(z^*)^{\mathrm{T}} H'(z^*) \Delta z^* &= 2H(z^*)^{\mathrm{T}}(-H(z^*) + \beta\theta(z^*) e^0) \\ &\leq -2\Psi(z^*) + 2\beta\theta(z^*)\|H(z^*)\| \\ &= -2R^* + 2\beta\theta(z^*)\|H(z^*)\|. \end{aligned}$$

By the definition of $\theta(\cdot)$ in (3.3), it is obvious that $\theta(z^*) \leq \tau$ if $\|H(z^*)\| \geq 1$. Hence we can obtain that $\theta(z^*)\|H(z^*)\| \leq \tau\Psi(z^*) = \tau R^*$. In addition, $\theta(z^*) \leq \Psi(z^*) \leq \tau R^*$ if $\|H(z^*)\| < 1$, and hence, $\theta(z^*)\|H(z^*)\| \leq \|H(z^*)\|\tau R^* \leq \tau R^*$. Thus, we get

$$2H(z^*)^{\mathrm{T}} H'(z^*) \Delta z^* \leq -2R^* + 2\beta\tau R^* = -2(1-\tau\beta) R^*. \tag{4.9}$$

Furthermore, from (4.8), (4.9), and $C^* > 0$, it is easy to see that $1 - \tau\beta \leq \sigma(1-\tau\beta)$, which contradicts the fact that $\sigma \in (0, \frac{1}{2})$ and $\tau\beta < 1$.

This proof is complete. □

5. Local superlinear convergence of Algorithm 3.1

In this section, we analyse the rate of convergence for Algorithm 3.1. A locally Lipschitz function $F : \Re^n \to \Re^m$, which has the generalized Jacobian $\partial F(x)$ in the sense of Clarke [13], is said to be semismooth (or strongly semismooth) at $x \in \Re^n$ if F is directionally differentiable at x and $F(x+h) - F(x) - Vh = o(\|h\|)(or = O(\|h\|^2))$ holds for any $V \in \partial F(x+h)$.

It is well known that convex functions, smooth functions, and piecewise linear functions are examples of semismooth functions; and the composition of (strongly) semismooth functions is still a (strongly) semismooth function [14]. It is easy to show that the function ϕ defined by (2.2) is strongly semismooth on $\Re^2$. Thus, on noticing that f is continuously differentiable, we obtain that the function H defined by (2.4) is semismooth (or strongly semismooth if f' is Lipschitz continuous on $\Re^n$).

Now, we show the local superlinear convergence of Algorithm 3.1.

Theorem 5.1. *Suppose that the conditions given in Theorem 4.1 are satisfied and $z^* = (\mu_*, x^*, s^*)$ is an accumulation point of $\{z^k\}$ generated by Algorithm 3.1. If all $V \in \partial H(z^*)$ are nonsingular, then:*

(i) *$\alpha_k \equiv 1$ for all z^k sufficiently close to z^*;*

(ii) *the whole sequence $\{z^k\}$ converges to z^*;*

(iii) $\|z^{k+1} - z^*\| = o(\|z^k - z^*\|)$ *(or* $\|z^{k+1} - z^*\| = O(\|z^k - z^*\|^2)$ *if* f' *is Lipschitz continuous on* $\Re^n$*); and*
(iv) $\mu_{k+1} = o(\mu_k)$ *(or* $\mu_{k+1} = O(\mu_k^2)$ *if* f' *is Lipschitz continuous on* $\Re^n$*).*

Proof. It holds by the proof of Theorem 4.1 that $H(z^*) = 0$ and z^* is a solution of $H(z) = 0$. Note that all $V \in \partial H(z^*)$ are singular, in [15, Proposition 3.1] we have that $\|H'(z^k)^{-1}\| = O(1)$ holds for all z^k sufficiently close to z^*. Since the function H is semismooth (or strongly semismooth if f' is Lipschitz continuous on $\Re^n$), it follows that for all z^k sufficiently close to z^*,

$$\|H(z^k) - H(z^*) - H'(z^k)(z^k - z^*)\| = o(\|z^k - z^*\|) \quad (\text{or } = O(\|z^k - z^*\|^2)).$$

Notice that the function H is locally Lipschitz continuous near z^*. Therefore, for all z^k sufficiently close to z^*, $\|H(z^k)\| = O(\|z^k - z^*\|)$, which implies that $\Psi(z^k) = o(\|z^k - z^*\|)$. Thus, for all z^k sufficiently close to z^*,

$$\begin{aligned}\|z^k + \Delta z^k - z^*\| &= \|z^k + H'(z^k)^{-1}(-H(z^k) + \beta\theta(z^k)e^0) - z^*\| \\ &\le \|H'(z^k)^{-1}\|(\|H(z^k) - H(z^*) - H'(z^k)(z^k - z^*)\| + \beta\theta(z^k)) \\ &\le \|H'(z^k)^{-1}\|(\|H(z^k) - H(z^*) - H'(z^k)(z^k - z^*)\| + \beta\tau\Psi(z^k)) \\ &= o(\|z^k - z^*\|) \quad (\text{or } = O(\|z^k - z^*\|^2)).\end{aligned} \tag{5.1}$$

By following the proof of [16, Theorem 3.1], we have that $\|z^k - z^*\| = O(\|H(z^k) - H(z^*)\|)$ holds for all z^k sufficiently close to z^*. Hence, for all z^k sufficiently close to z^*,

$$\begin{aligned}\|H(z^k + \Delta z^k)\| &= O(\|z^k + \Delta z^k - z^*\|) \\ &= o(\|z^k - z^*\|) \quad (\text{or } = O(\|z^k - z^*\|^2)) \\ &= o(\|H(z^k) - H(z^*)\|) \quad (\text{or } = O(\|H(z^k) - H(z^*)\|^2)) \\ &= o(\|H(z^k)\|) \quad (\text{or } = O(\|H(z^k)\|^2)).\end{aligned} \tag{5.2}$$

By the proof of Theorem 4.1 it follows that $\lim_{k\to\infty} \|H(z^k)\| = 0$. Hence, (5.2) implies that $\alpha_k = 1$ holds for all z^k which is sufficiently close to z^*. This proves the result (i). Therefore, for all z^k sufficiently close to z^*, we have $z^{k+1} = z^k + \Delta z^k$, which, together with (5.1), indicates that the results (ii) and (iii) hold.

In addition, since $\mu_{k+1} = \mu_k + \Delta\mu_k = \beta\theta(z^k) \le \beta\tau\Psi(z^k)$ for all sufficiently large k, by using (5.2) we have that

$$\begin{aligned}\mu_{k+1} &= O(\Psi(z^k)) = o(\Psi(z^{k-1})) \quad (\text{or } = O(\Psi(z^{k-1})^2)) \\ &= o(\mu_k) \quad (\text{or } = O(\mu_k^2)),\end{aligned}$$

i.e., the result (iv) holds. This completes the proof. □

6. Further discussions and numerical results

Consider the following system of equalities and inequalities:

$$\begin{cases} f_i(x) \le 0, & i \in \{1, \dots, m\}, \\ f_i(x) = 0, & i \in \{m+1, \dots, p\}, \end{cases} \tag{6.1}$$

where $x \in \Re^n$ and $f(x) := (f_1(x), \dots, f_p(x))^T$ with every $f_i : \Re^n \to \Re$ being a continuously differentiable function.

Now, we assume that $p \ne n$. In this case, Algorithm 3.1 cannot be directly applied to solve (6.1). In a similar way to those given in [1, Section 4], however, (6.1) can be transformed as a new problem and we may solve the new problem using Algorithm 3.1, so a solution of the original problem can be found.

- Suppose that $p < n$. In this case, we assume that $n = p + 1$ without loss of generality. Then, we may add a trivial inequality, such as $\sum_{i=1}^n x_i^2 \le M$ where M is a sufficiently large number, into (6.1) so that Algorithm 3.1 can be applied to solve the new problem. Since the added inequality holds trivially, we may obtain a solution of (6.1).
- Suppose that $p > n$. In this case, we assume that $p = n + 1$ without loss of generality. If $m \ge 1$, we may add a variable x_{n+1} into the inequalities, for example $f_i(x) \le 0$ becomes $f_i(x) + x_{n+1}^2 \le 0$; and if $m = 0$, then we may add a trivial inequality, such as $\sum_{i=1}^{n+2} x_i^2 \le M$ where M is a sufficiently large number, into (6.1). Note that we have added two new variables x_{n+1} and x_{n+2} in the new inequality. In the transformed problem, the number of variables is equal to the sum of the numbers of equalities and inequalities, and hence, we may solve the transformed problem, so a solution of the original problem can be found.

In the following, we implement Algorithm 3.1 for solving systems of equalities and inequalities in Matlab in order to see the behavior of Algorithm 3.1. In our implementation, the function H defined by (2.4) is replaced by

$$H(z) := \begin{pmatrix} \mu \\ f_{\mathcal{I}}(x) - s + c\mu x_{\mathcal{I}} \\ f_{\mathcal{E}}(x) + c\mu x_{\mathcal{E}} \\ \Phi(\mu, s) + c\mu s \end{pmatrix}, \tag{6.2}$$

where c is a given constant. It is easy to see that such a change does not destroy any theoretical results obtained in this paper. In order to obtain an interior solution of (1.1) (i.e., a solution x^* of (1.1) with $f_{\mathcal{I}}(x^*) < 0$), we solve the following system of equalities and inequalities:

$$\begin{cases} f_{\mathcal{I}}(x) + \varepsilon e \leq 0, \\ f_{\mathcal{E}}(x) = 0, \end{cases}$$

where ε is a sufficiently small number and e is a vector of all ones. Throughout our computational experiments, the parameters used in the algorithm are chosen as

$$\delta := 0.3, \qquad \sigma := 0.0001, \qquad \varepsilon := 0.00001, \qquad \beta := 1.0, \qquad \mu = 1.0, \qquad Q_0 = 1.0,$$

and the parameters c, τ, η_0 and the starting point x^0 are chosen according to those listed in Table 1. Set $s^0 := f_{\mathcal{I}}(x^0)$, $z^0 := (\mu_0, x^0, s^0)$, and $\eta_k = \eta_0$ for all $k \in \mathcal{J}$. We use $\|H(z^k)\| \leq 10^{-3}$ as the stopping rule.

We consider the following seven examples.

Example 6.1. Consider (1.1), where $f := (f_1, f_2)^T$ with $x \in \Re^2$ and

$$f_1(x) := x_1^2 + x_2^2 - 1 \leq 0; \qquad f_2(x) := -x_1^2 - x_2^2 + (0.999)^2 \leq 0.$$

Example 6.2. Consider (1.1), where $f := (f_1, f_2, f_3, f_4, f_5, f_6)^T$ with $x \in \Re^2$ and

$$\begin{aligned} &f_1(x) := \sin(x_1) \leq 0; && f_2(x) := -\cos(x_2) \leq 0; && f_3(x) := x_1 - 3\pi \leq 0; \\ &f_4(x) := x_2 - \pi/2 - 2 \leq 0; && f_5(x) := -x_1 - \pi \leq 0; && f_6(x) := -x_2 - \pi/2 \leq 0. \end{aligned}$$

Example 6.3. Consider (1.1), where $f := (f_1, f_2)^T$ with $x \in \Re^2$ and

$$f_1(x) := \sin(x_1) \leq 0; \qquad f_2(x) := -\cos(x_2) \leq 0.$$

Example 6.4. Consider (1.1), where $f := (f_1, f_2, f_3, f_4, f_5)^T$ with $x \in \Re^5$ and

$$\begin{aligned} &f_1(x) := x_1 + x_3 - 1.6 \leq 0; && f_2(x) := 1.333x_2 + x_4 - 3 \leq 0; \\ &f_3(x) := -x_3 - x_4 + x_5 \leq 0; && f_4(x) := x_1^2 + x_3^2 - 1.25 = 0; \\ &f_5(x) := x_2^{1.5} + 1.5x_4 - 3 = 0. \end{aligned}$$

Example 6.5. Consider (1.1), where $f := (f_1, f_2, f_3)^T$ with $x \in \Re^3$ and

$$\begin{aligned} &f_1(x) := x_1 + x_2 \mathrm{e}^{0.8x_3} + \mathrm{e}^{1.6} \leq 0; && f_2(x) := x_1^2 + x_2^2 + x_3^2 - 5.2675 = 0; \\ &f_3(x) := x_1 + x_2 + x_3 - 0.2605 = 0. \end{aligned}$$

Example 6.6. Consider (1.1), where $f := (f_1, f_2, f_3)^T$ with $x \in \Re^2$ and

$$\begin{aligned} &f_1(x) := 0.8 - \mathrm{e}^{x_1 + x_2} \leq 0; && f_2(x) := 1.21\mathrm{e}^{x_1} + \mathrm{e}^{x_2} - 2.2 = 0; \\ &f_3(x) := x_1^2 + x_2^2 + x_2 - 0.1135 = 0. \end{aligned}$$

Example 6.7. Consider (1.1), where $f := (f_1, f_2)^T$ with $x \in \Re^2$ and

$$f_1(x) := x_1 - 0.7\sin(x_1) - 0.2\cos(x_2) = 0; \quad f_2(x) := x_2 - 0.7\cos(x_1) + 0.2\sin(x_2) = 0.$$

The first three examples only contain inequalities, which were tested in [1]. The other examples contain equalities and inequalities. Instead of these seven examples, we use Algorithm 3.1 to solve the following problems.

Example 1′. Consider (1.1), where $f := (f_1, f_2)^T$ with $x \in \Re^2$ and

$$f_1(x) := x_1^2 + x_2^2 - 1 + \varepsilon \leq 0; \qquad f_2(x) := -x_1^2 - x_2^2 + (0.999)^2 + \varepsilon \leq 0.$$

Example 2′. Consider (1.1), where $f := (f_1, f_2, f_3, f_4, f_5, f_6)^T$ with $x \in \Re^6$ and

$$\begin{aligned} &f_1(x) := \sin(x_1) + \varepsilon \leq 0; && f_2(x) := -\cos(x_2) + \varepsilon \leq 0; \\ &f_3(x) := x_1 - 3\pi + x_3^2 + \varepsilon \leq 0; && f_4(x) := x_2 - \pi/2 - 2 + x_4^2 + \varepsilon \leq 0; \\ &f_5(x) := -x_1 - \pi + x_5^2 + \varepsilon \leq 0; && f_6(x) := -x_2 - \pi/2 + x_6^2 + \varepsilon \leq 0. \end{aligned}$$

Example 3′. Consider (1.1), where $f := (f_1, f_2)^T$ with $x \in \Re^2$ and

$$f_1(x) := \sin(x_1) + \varepsilon \leq 0; \qquad f_2(x) := -\cos(x_2) + \varepsilon \leq 0.$$

Table 1
The numerical results of Examples 1′–7′.

Exam	ST	C	Tau	Eta0	NI	NF	SOL
1′	$(0, 5)^T$	100	0.006	0.01	8	9	$(-0.6188, 0.7853)^T$
2′	$(0, 0, 0, 0, 0, 0)^T$	0.5	0.2	0.01	6	8	$(-0.0097, 1.4281, 2.8461, 1.2795, 1.6392, 1.6662)^T$
3′	$(0, 0)^T$	0.5	0.2	0.01	3	3	$(-0.0152, 0.7207)^T$
4′	$(0.5, 2, 1, 0, 0)^T$	5	0.02	0.8	4	4	$(0.5557, 1.3243, 0.9703, 0.9840, 1.1561)^T$
5′	$(-1, -1, 1)^T$	0.5	0.2	0.8	5	5	$(-0.8301, -0.8662, 1.9566)^T$
6′	$(0, 0, 0)^T$	0.5	0.02	0.8	4	4	$(-0.0953, 0.0953, 0.3259)^T$
7′	$(0, 1, 0)^T$	0.5	0.006	0.8	9	14	$(0.5268, 0.5084, -99.9973)^T$

Example 4′. Consider (1.1), where $f := (f_1, f_2, f_3, f_4, f_5)^T$ with $x \in \Re^5$ and

$$\begin{aligned} &f_1(x) := x_1 + x_3 - 1.6 + \varepsilon \le 0; \quad f_2(x) := 1.333x_2 + x_4 - 3 + \varepsilon \le 0;\\ &f_3(x) := -x_3 - x_4 + x_5 + \varepsilon \le 0; \quad f_4(x) := x_1^2 + x_3^2 - 1.25 = 0;\\ &f_5(x) := x_2^{1.5} + 1.5x_4 - 3 = 0. \end{aligned}$$

Example 5′. Consider (1.1), where $f := (f_1, f_2, f_3)^T$ with $x \in \Re^3$ and

$$\begin{aligned} &f_1(x) := x_1 + x_2 e^{0.8x_3} + e^{1.6} + \varepsilon \le 0; \quad f_2(x) := x_1^2 + x_2^2 + x_3^2 - 5.2675 = 0;\\ &f_3(x) := x_1 + x_2 + x_3 - 0.2605 = 0. \end{aligned}$$

Example 6′. Consider (1.1), where $f := (f_1, f_2, f_3)^T$ with $x \in \Re^3$ and

$$\begin{aligned} &f_1(x) := 0.8 - e^{x_1 + x_2} + x_3^2 + \varepsilon \le 0; \quad f_2(x) := 1.21e^{x_1} + e^{x_2} - 2.2 = 0;\\ &f_3(x) := x_1^2 + x_2^2 + x_2 - 0.1135 = 0. \end{aligned}$$

Example 7′. Consider (1.1), where $f := (f_1, f_2, f_3)^T$ with $x \in \Re^3$ and

$$\begin{aligned} &f_1(x) := x_1^2 + x_2^2 + x_3^2 - 10000 + \varepsilon \le 0; \quad f_2(x) := x_1 - 0.7\sin(x_1) - 0.2\cos(x_2) = 0;\\ &f_3(x) := x_2 - 0.7\cos(x_1) + 0.2\sin(x_2) = 0. \end{aligned}$$

The numerical results are listed in Table 1, where Exam denotes the tested examples; ST denotes the starting point x^0; C denotes the value of the parameter c given in (6.2); Tau denotes the value of the parameter τ given in Algorithm 3.1; Eta0 denotes the value of the parameter η_0 given in Algorithm 3.1; NI denotes the total number of iterations; NF denotes the number of function evaluations for the function $H(z^k)$; and SOL denotes the solution obtained by Algorithm 3.1.

From Table 1 we obtain:

- a solution of Example 6.1, $(-0.6188, 0.7853)^T$, in eight iterations;
- a solution of Example 6.2, $(-0.0294, 1.5416)^T$, in six iterations;
- a solution of Example 6.3, $(-0.0152, 0.7207)^T$, in three iterations;
- a solution of Example 6.4, $(0.5557, 1.3243, 0.9703, 0.9840, 1.1561)^T$, in four iterations;
- a solution of Example 6.5, $(-0.8301, -0.8662, 1.9566)^T$, in five iterations;
- a solution of Example 6.6, $(-0.0953, 0.0953)^T$, in four iterations;
- a solution of Example 6.7, $(0.5268, 0.5084)^T$, in nine iterations.

From Table 1, it is easy to see that all problems that we tested have been solved with a small number of iterations and a small number of function evaluations.

- We have also tested these problems by using Algorithm 3.1 with the stopping rule $\|H(z^k)\| \le 10^{-6}$. The numerical results are listed in Table 2, where F denotes that the algorithm fails. From Table 2, it is easy to see that most numerical results are similar to those given in Table 1, but the algorithm fails to find a solution to Example 1′.
- We have also tested these problems by using the algorithm with the monotone line search (i.e., take $\eta_k = 0$ for all k in Algorithm 3.1), and the numerical results are listed in Table 3. It is easy to see from Tables 2 and 3 (or Table 1 and the numerical results in [1]) that the algorithm with the nonmonotone line search has some advantages over the one with the monotone line search.

We have also tested some other problems, and the computation effect is similar.

Table 2
The numerical results of Examples 1′–7′ using Algorithm 3.1 with accuracy 10^{-6}.

Exam	ST	C	Tau	Eta0	NI	NF	SOL
1′	$(0, 5)^T$	100	0.006	0.01	F	F	F
2′	$(0, 0, 0, 0, 0, 0)^T$	0.5	0.2	0.01	7	9	$(-0.0093, 1.4289, 2.8462, 1.2794, 1.6395, 1.6666)^T$
3′	$(0, 0)^T$	0.5	0.2	0.01	4	4	$(-0.0152, 0.7206)^T$
4′	$(0.5, 2, 1, 0, 0)^T$	5	0.02	0.8	5	5	$(0.5563, 1.3259, 0.9698, 0.9822, 1.1545)^T$
5′	$(-1, -1, 1)^T$	0.5	0.2	0.8	6	6	$(-0.8299, -0.8663, 1.9566)^T$
6′	$(0, 0, 0)^T$	0.5	0.02	0.8	4	4	$(-0.0953, 0.0953, 0.3259)^T$
7′	$(0, 1, 0)^T$	0.5	0.006	0.8	10	15	$(0.5265, 0.5079, -99.9973)^T$

Table 3
The numerical results of Examples 1′–7′ using Algorithm 3.1 with the monotone line search and accuracy 10^{-6}.

Exam	ST	C	Tau	Eta0	NI	NF	SOL
1′	$(0, 5)^T$	100	0.006	0	F	F	F
2′	$(0, 0, 0, 0, 0, 0)^T$	0.5	0.2	0	7	9	$(-0.0093, 1.4289, 2.8462, 1.2794, 1.6395, 1.6666)^T$
3′	$(0, 0)^T$	0.5	0.2	0	4	4	$(-0.0152, 0.7206)^T$
4′	$(0.5, 2, 1, 0, 0)^T$	5	0.02	0	5	5	$(0.5563, 1.3259, 0.9698, 0.9822, 1.1545)^T$
5′	$(-1, -1, 1)^T$	0.5	0.2	0	6	6	$(-0.8299, -0.8663, 1.9566)^T$
6′	$(0, 0, 0)^T$	0.5	0.02	0	4	4	$(-0.0953, 0.0953, 0.3259)^T$
7′	$(0, 1, 0)^T$	0.5	0.006	0	50	207	$(0.5265, 0.5079, -99.9973)^T$

7. Some final remarks

In this paper, we investigated a smoothing-type algorithm with a nonmonotone line search for solving the system of equalities and inequalities. Under suitable assumptions, we proved that the algorithm is globally and locally quadratically convergent. The preliminary numerical results demonstrate that the nonmonotone smoothing-type algorithm discussed in this paper is effective for solving this class of problems.

References

[1] Z.H. Huang, Y. Zhang, W. Wu, A smoothing-type algorithm for solving system of inequalities, J. Comput. Appl. Math. 220 (2008) 355–363.
[2] J.W. Daniel, Newton's method for nonlinear inequalities, Numer. Math. 21 (1973) 381–387.
[3] D.Q. Mayne, E. Polak, A.J. Heunis, Solving nonlinear inequalities in a finite number of iterations, J. Optim. Theory Appl. 33 (1981) 207–221.
[4] M. Sahba, On the solution of nonlinear inequalities in a finite number of iterations, Numer. Math. 46 (1985) 229–236.
[5] H.X. Yin, Z.H. Huang, L. Qi, The convergence of a Levenberg–Marquardt method for the l_2-norm solution of nonlinear inequalities, Numer. Funct. Anal. Optim. 29 (2008) 687–716.
[6] L. Qi, D. Sun, G. Zhou, A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequality problems, Math. Program. 87 (2000) 1–35.
[7] Z.H. Huang, J. Han, Z. Chen, A predictor–corrector smoothing Newton algorithm, based on a new smoothing function, for solving the nonlinear complementarity problem with a P_0 function, J. Optim. Theory Appl. 117 (2003) 39–68.
[8] Z.H. Huang, S.L. Hu, J. Han, Global convergence of a smoothing algorithm for symmetric cone complementarity problems with a nonmonotone line search, Sci. China Ser. A 52 (2009) 833–848.
[9] S.L. Hu, Z.H. Huang, P. Wang, A non-monotone smoothing Newton algorithm for solving nonlinear complementarity problems, Optim. Methods Softw. 24 (2009) 447–460.
[10] L. Grippo, F. Lampariello, S. Lucidi, A nonmonotone line search technique for Newton's method, SIAM J. Numer. Anal. 23 (1986) 707–716.
[11] H.C. Zhang, W.W. Hager, A nonmonotone line search technique and its application to unconstrained optimization, SIAM J. Optim. 14 (2004) 1043–1056.
[12] S.L. Hu, Z.H. Huang, A non-monotone line search algorithm for unconstrained optimization, J. Sci. Comput. (2009), in press (doi:10.1007/s10915-009-9314-0).
[13] F.H. Clark, Optimization and Nonsmooth Analysis, Wiley, New York, 1983.
[14] A. Fischer, Solution of monotone complementarity problems with Lipschitzian functions, Math. Program. 76 (1997) 513–532.
[15] L. Qi, J. Sun, A nonsmooth version of Newton's method, Math. Program. 58 (1993) 353–367.
[16] L. Qi, Convergence analysis of some algorithm for nonsmooth equations, Math. Oper. Res. 18 (1993) 227–244.