NUMERICAL MATHEMATICS: Theory, Methods and Applications
Numer. Math. Theor. Meth. Appl., Vol. **2**, No. 2, pp. 224-236 (2009)

Smoothing Newton-Like Method for the Solution of Nonlinear Systems of Equalities and Inequalities

Liu Yang[1], Yanping Chen[2,*] and Xiaojiao Tong[3]

[1] *School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411105, Hunan, China.*
[2] *School of Mathematical Sciences, South China Normal University, Guangzhou 510631, China.*
[3] *School of Mathematics and Computational Science, Changsha University of Science and Technology, Changsha 410076, Hunan, China.*

Received 30 January 2008; Accepted (in revised version) 27 October 2008

Abstract. In this paper, we present a smoothing Newton-like method for solving nonlinear systems of equalities and inequalities. By using the so-called max function, we transfer the inequalities into a system of semismooth equalities. Then a smoothing Newton-like method is proposed for solving the reformulated system, which only needs to solve one system of linear equations and to perform one line search at each iteration. The global and local quadratic convergence are studied under appropriate assumptions. Numerical examples show that the new approach is effective.

AMS subject classifications: 65K05, 90C30

Key words: Nonlinear systems of equalities and inequalities, semismooth function, smoothing Newton method, global convergence, local quadratic convergence.

1. Introduction

In this paper, we present a smoothing Newton-like method for the numerical solution of nonlinear systems of equalities and inequalities defined by

$$\begin{cases} c_i(x)=0, & i\in E=\{1,2,\cdots,m_e\}, \\ c_i(x)\le 0, & i\in I=\{m_e+1,\cdots,m\}, \end{cases} \tag{1.1}$$

where $c_i(x):\mathbb{R}^n\to\mathbb{R}, i=1,\cdots,m$ are continuously differentiable. Throughout this paper, we assume that the solution set of (1.1) is nonempty.

Systems of nonlinear equalities and inequalities appear in a wide variety of problems in applied mathematics. These systems play a central role in the model formulation design

*Corresponding author. *Email addresses:* `yangl410@xtu.edu.cn` (L. Yang), `yanpingchen@scnu.edu.cn` (Y. P. Chen), `tongxj@csust.edu.cn` (X. J. Tong)

and analysis of numerical techniques employed in solving problems arising in optimization, power system, nonlinear complementarity and variational inequalities, etc. Many researchers considered the problem, especially for the numerical methods, see, e.g., [1–8]. Daniel in [4] considered Newton's methods for solving the problem ; Polyak in [5] discussed a gradient method; Burke and Han in [6] presented a Gauss-Newton approach to solve generalized inequalities; Dennis et al. firstly presented trust-region methods for solving (1.1) and proved the global convergence under certain conditions in [7]; Tong-Zhou in [8] also studied another trust-region method for the problem and proved the global convergence under general conditions.

In the last decade, a class of popular numerical methods, namely, the so-called semismooth Newton methods, has been studied extensively for solving semismooth equations, see, e.g., [9–11, 16] and references therein. The typical characteristic of the semismooth Newton is twofold: it extends the classical Newton method for nonsmooth equations; it enjoys the same convergent property such as the locally superlinear convergence. The semismooth Newton method was firstly presented by Qi-Sun in [16]. Then it is studied extensively and used for solving many mathematical problems, such as large scale nonlinear complementarity, variational inequalities and the KKT system of optimization problems, etc (see [10]). Following the semismooth Newton methods, another related numerical method, called the smoothing Newton method, is also presented to help the calculation of generalized derivative of nonsmooth functions, see, e.g., [12–15, 18]. The key idea of smoothing Newton method is to approximate the nonsmooth function $F(x)$ by a smooth function $f(x,\varepsilon)$, where ε is called smoothing parameter. Then the generalized derivative $\partial F(x)$ is approximated by $f'(x,\varepsilon)$ with respect to the variable x. The main advantage of the smoothing Newton methods is that it still retains the nice convergence property. There are two kinds of smoothing methods. One is to handle ε as a parameter, which is updated step by step in iterations (see [12]). The other one is to handle ε as a variable. Then an extended system of equations is set and solved by Newton-type methods. Both smoothing Newton methods are proved to have nice convergence property.

We note that the methods proposed in [7, 8] are based on the least-squares approach, which implies that the system (1.1) is solved by optimization methods. As we all know, compared with the optimization-based methods, solving a system of equations is much easier and it has less calculating cost. Furthermore, the solution of the optimization-based method is the stationary point of the optimization problem, and may not be the solution of (1.1). Therefore, our aim in this paper is to set a system of equations for solving (1.1). To this end, we only consider the case that $m \le n$. We transfer (1.1) into a system of semismooth equations firstly, then a smoothing algorithm is presented for solving the equivalent system. Now we introduce the transformation of (1.1). Denote the maximal function by $\max\{0, c_i(x)\}$. The inequalities

$$c_i(x) \le 0, \quad i \in I$$

can be changed equivalently into the following equations:

$$\max\{0, c_i(x)\} = 0, \quad i \in I.$$

Therefore problem (1.1) is equivalent to the following problem:

$$F(x) = \begin{cases} c_i(x) = 0, & i \in E, \\ \max\{0, c_i(x)\} = 0, & i \in I. \end{cases} \tag{1.2}$$

Note that the system (1.2) is a system of semismooth equations due to the semismooth property of the maximal function.

The remainder of this paper is organized as follows. In Section 2, we construct a smoothing approximation function and study some related properties. In Section 3, we present a smoothing Newton algorithm and state some preliminary results. In Section 4, we study the global and local quadratic convergence of the proposed algorithm. Numerical results and final conclusions are given in Section 5.

The following notations will be used throughout this paper. $\|.\|$ denotes the 2-norm; $\mathbb{R}_{++}$ and $\mathbb{R}_+$ represent the sets $\{x \in \mathbb{R}, x > 0\}$ and $\{x \in \mathbb{R}, x \geq 0\}$ respectively.

2. The smoothing function and the related properties

In this section, we construct the smoothing approximation function $p(u, x)$ for

$$\max\{0, c_i(x)\}, \quad i \in I$$

by Chen-Mangasarian's smoothing approach in [14], where

$$p_i(u, x) = q(u_i, c_i(x)), \quad i \in I, \tag{2.1a}$$

$$q(\mu, \omega) = \begin{cases} \phi(|\mu|, \omega), & \mu \neq 0, \\ \max\{0,\ \omega\}, & \mu = 0, \end{cases} \tag{2.1b}$$

$$\phi(\mu, \omega) = \int_{-\infty}^{\infty} \max\{0,\ \omega - \mu s\} \rho(s) ds, \quad \mu \in \mathbb{R}_{++}. \tag{2.1c}$$

Let

$$supp(\rho) = \{s \in \mathbb{R} : \rho(s) > 0\}.$$

We select the kernel function

$$\rho(s) = \frac{e^{-s}}{(1 + e^{-s})^2},$$

with

$$\int_{-\infty}^{\infty} \rho(s) ds = 1, \quad k := \int_{-\infty}^{\infty} |s| \rho(s) ds = 2 \ln 2 < \infty.$$

Then the corresponding smoothing function is

$$\phi(\mu, \omega) = \mu \ln(1 + e^{\omega/\mu}), \quad (\mu, \omega) \in \mathbb{R}_{++} \times \mathbb{R}. \tag{2.2}$$

Let $\phi_0(\omega) := \max\{0, w\}$. We can get the following result easily.

Lemma 2.1. *For any $(\mu,\omega)\in\mathbb{R}_{++}\times\mathbb{R}$, we have*

$$\lim_{\mu\to 0}\phi(\mu,\omega)=\phi_0(\omega).$$

For the sake of convenience, for any given $\mu\in\mathbb{R}_{++}$, let $\phi_\mu(\omega):\mathbb{R}\to\mathbb{R}$ be defined by

$$\phi_\mu(\omega):=\phi(\mu,\omega),\quad \omega\in\mathbb{R}. \tag{2.3}$$

Lemma 2.2. *For any given $\mu>0$, the mapping $\phi_\mu(\omega)$ is continuously differentiable with*

$$\phi'_\mu(w)=\frac{e^{\omega/\mu}}{1+e^{\omega/\mu}}\in(0,1). \tag{2.4}$$

Proof. For any given $\mu>0$, we can get (2.4) by directly calculating the derivative of $\phi_\mu(\omega)$ with respect to ω. □

Lemma 2.3. *The mapping $q(\mu,\omega)$ is Lipschitz continuous on $\mathbb{R}^2$ with Lipschitz constant $L:=4\ln 2$.*

Proof. Suppose that (μ_1,ω_1) and (μ_2,ω_2) are two arbitrary points of $\mathbb{R}^2$. Then we have

$$\begin{aligned}
&|q(\mu_1,\omega_1)-q(\mu_2,\omega_2)|\\
&=\left|\int_{-\infty}^{\infty}\max\{0,\omega_1-|\mu_1|s\}\rho(s)ds-\int_{-\infty}^{\infty}\max\{0,\omega_2-|\mu_2|s\}\rho(s)ds\right|\\
&\le\int_{-\infty}^{\infty}|\max\{0,\omega_1-|\mu_1|s\}-\max\{0,\omega_2-|\mu_2|s\}|\rho(s)ds\\
&\le\int_{-\infty}^{\infty}|(\omega_1-|\mu_1|s)-(\omega_2-|\mu_2|s)|\rho(s)ds\\
&\le\int_{-\infty}^{\infty}|\omega_1-\omega_2|\rho(s)ds+\int_{-\infty}^{\infty}|\mu_1-\mu_2||s|\rho(s)ds\\
&=|\omega_1-\omega_2|+k|\mu_1-\mu_2|\\
&\le 2\max\{1,k\}\|(\mu_1,\omega_1)-(\mu_2,\omega_2)\|\\
&=4\ln 2\|(\mu_1,\omega_1)-(\mu_2,\omega_2)\|,
\end{aligned}$$

which completes the proof of the lemma. □

In the following, we set the smoothing system of equations of (1.2). Let $z=(u,x)\in\mathbb{R}\times\mathbb{R}^n$, define a mapping $H:\mathbb{R}^{n+1}\to\mathbb{R}^{m+1}$ by

$$H(z):=\begin{pmatrix} u\\ c_E(x)\\ p(u,x)\end{pmatrix}, \tag{2.5}$$

where $c_E(x)$ represents all $c_i(x), i \in E$, $p(u,x) = (p_{m_e+1}(u,x), \cdots, p_m(u,x))^T$ is the smoothing approximation function of $\max\{0, c_i(x)\}, i \in I$.

From Lemma 2.1, we know that (1.2) is equivalent to the following equation

$$H(z) = 0.$$

Then we have the following result for the smoothing system (2.5).

Theorem 2.1. *Let the smoothing function be* (2.2)*. Then $H(z)$ is continuously differentiable at any $z = (u,x) \in \mathbb{R}_{++} \times \mathbb{R}^n$, and*

$$H'(z) = \begin{pmatrix} 1 & \mathbf{0}_{1\times n} \\ \mathbf{0}_{m_e \times 1} & \nabla c_E(x) \\ p'_u(u,x) & p'_x(u,x) \end{pmatrix}, \tag{2.6}$$

where $\nabla c_E(x)$ is the Jacobi matrix of $c_i(x) = 0,\ i \in E$,

$$p'_u = (u,x) = \left(\frac{\partial p_{m_e+1}(u,x)}{\partial u}, \cdots, \frac{\partial p_m(u,x)}{\partial u} \right)^T, \tag{2.7}$$

$$p'_x(u,x) = \begin{pmatrix} \frac{\partial p_{m_e+1}(u,x)}{\partial x_1} & \cdots & \frac{\partial p_{m_e+1}(u,x)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial p_m(u,x)}{\partial x_1} & \cdots & \frac{\partial p_m(u,x)}{\partial x_n} \end{pmatrix}, \tag{2.8}$$

where

$$\frac{\partial p_i(u,x)}{\partial x_j} = \frac{e^{c_i(x)/u}}{1 + e^{c_i(x)/u}} \cdot \left(\frac{\partial c_i(x)}{\partial x_j} \right), \quad i \in I, \quad j = 1, \cdots, n.$$

3. Smoothing Newton-like methods

We now give our smoothing Newton-like algorithm for solving (1.1). Choose $\bar{u} \in \mathbb{R}_{++}, \gamma \in (0,1)$, such that $\gamma \bar{u} < 1$, Let $\bar{z} := (\bar{u}, \mathbf{0}) \in \mathbb{R}_{++} \times \mathbb{R}^n$.

Define the merit function $\psi : \mathbb{R}^{n+1} \to \mathbb{R}_+$ by

$$\psi(z) := \|H(z)\|^2. \tag{3.1}$$

Define $\beta : \mathbb{R}^{n+1} \to \mathbb{R}_+$ by

$$\beta(z) = \gamma \min\{1, \psi(z)\}. \tag{3.2}$$

Let $\Omega := \{z = (u,x) \in \mathbb{R} \times \mathbb{R}^n | u \geq \beta(z)\bar{u}\}$. Thus it ensures $z^0 = (\bar{u}, x) \in \Omega$ for any $x \in \mathbb{R}^n$.

Algorithm 3.1. (Smoothing Newton-like method)

Step 0: Choose constants $\delta \in (0,1), \sigma \in (0,0.5)$, let $z^0 = (u^0, x^0), u^0 := \bar{u}, x^0 \in \mathbb{R}^n$ be an arbitrary point and $k := 0$;

Step 1: If $\|H(z^k)\| = 0$ then stop. Otherwise, let $\beta_k := \beta(z^k)$.

Step 2: Compute $\Delta(z^k) := (\Delta u^k, \Delta x^k) \in \mathbb{R} \times \mathbb{R}^n$ by

$$H(z^k) + H'(z^k)\Delta z^k = \beta_k \bar{z}. \tag{3.3}$$

Step 3: Let λ_k be the largest value in the set $\{\delta^i / i = 0, 1, \cdots\}$ such that the inequality is satisfied

$$\psi(z^k + \delta^i \Delta z^k) \leq [1 - 2\sigma(1 - \gamma\bar{u})\delta^i]\psi(z^k). \tag{3.4}$$

Define $z^{k+1} := z^k + \lambda_k \Delta z^k$.

Step 4: Replace k by $k+1$ and go to Step 1.

Remark 3.1. (i) Algorithm 3.1 is a modified version of the smoothing Newton method developed in [13]. When $m = n$, Algorithm 3.1 is well-defined if the Jacobi matrix of $H'(z)$ is nonsingular. When $m < n$, $H(z)$ is a underdetermined system which we can use the generalized inverse proposed in [22] to solve it, i.e.,

$$\Delta z^k = H'(z^k)^+(-H(z^k) + \beta_k \bar{z}), \tag{3.5}$$

where $H'(z^k)^+$ is the Moore-Penrose inverse.

(ii) Suppose that the sequence $\{z^k = (u^k, x^k)\} \in \mathbb{R}^{n+1}$ is generated by Algorithm 3.1. From the first equation of (3.3) it follows that

$$u^{k+1} = u^k + \lambda_k \Delta u^k = (1 - \lambda_k)u^k + \lambda_k \bar{u}\beta(z^k) > 0,$$

which indicates that $u^k > 0$ for all $k \geq 0$. Consequently, $H(z)$ is continuously differentiable at any z^k. From (i), we can know that Eq. (3.3) is solvable for all $k \geq 0$ and if $H'(z^k)$ is of full rank $m+1$. Then the solution is unique. On the other hand, just like the proof of Lemma 5 in [13], we can get that for any $\alpha \in [0,1]$,

$$\psi(z^k + \alpha\Delta z^k) \leq [1 - 2(1 - \gamma\bar{u})\alpha]\psi(z^k) + o(\alpha). \tag{3.6}$$

Thus the line search (3.4) is well defined in Algorithm 3.1.

(iii) From (3.4), (3.6), it is easy to see that the sequence $\{\psi(z^k)\}$ is monotonically decreasing, which also implies that the sequence $\{\beta(z^k)\}$ is monotonically decreasing.

So we get the following result.

Lemma 3.1. *Let the smoothing function be* (2.2) *and for any* $z^k = (u^k, x^k) \in \mathbb{R}_{++} \times \mathbb{R}^n$, *assume* $rank(H'(z^k)) = m+1$, *then Algorithm* 3.1 *is well-defined and generates an infinite sequence* $\{z^k\}$ *with* $u^k \in \mathbb{R}_{++}, \{z^k\} \in \Omega$, *for all* $k \geq 0$.

Proof. From Remark 3.1(ii), we can know Algorithm 3.1 is well-defined at the k-th iteration. And if $z^0 = (\bar{u}, x^0), \bar{u} \in \mathbb{R}_{++}, rank(H'(z^0)) = m+1$, then from (3.6) we can get a point $z^1 = (u^1, x^1) \in \mathbb{R}_{++} \times \mathbb{R}^n$ which satisfies (3.4) in Algorithm 3.1. On the other hand, for any $x \in \mathbb{R}^n, z^0 = (\bar{u}, x) \in \Omega$, we have

$$\begin{aligned} u^1 - \beta(z^1)\bar{u} &= (1-\lambda_0)u^0 + \lambda_0\beta(z^0)\bar{u} - \beta(z^1)\bar{u} \\ &\geq (1-\lambda_0)\beta(z^0)\bar{u} + \lambda_0\beta(z^0)\bar{u} - \beta(z^1)\bar{u} \\ &= (\beta(z^0) - \beta(z^1))\bar{u} \geq 0. \end{aligned}$$

It means that $z^1 \in \Omega, u^1 \in \mathbb{R}_{++}$. Then by induction it yields that an infinite sequence $\{z^k\}$ is generated by Algorithm 3.1, and $u^k \in \mathbb{R}_{++}, \{z^k\} \in \Omega$. $\square$

4. Convergence analysis

In this section, we analyze the global and local convergence of Algorithm 3.1. We make the following basic hypothesis throughout this paper.

Assumption 4.1. The level set $L(z^0) = \{z \in \mathbb{R}_+ \times \mathbb{R}^n : \|H(z)\| \leq \|H(z^0)\|\}$ is bounded.

Assumption 4.2. For any $(x,u) \in L(z^0), \nabla c_E(x), \nabla c_I(x)$ are Lipschitz continuous and $c_E(x), c_I(x), \nabla c_E(x), \nabla c_I(x)$ are all bounded in norm.

From these assumptions, we can know that for any $x, y \in L(z^0)$, there exist constants $b_1, b_2, b_3, b_4, r_E, r_I > 0$ satisfying

$$\begin{aligned} &\|c_E(x)\| \leq b_1, \quad \|c_I(x)\| \leq b_2, \quad \|\nabla c_E(x)\| \leq b_3, \quad \|\nabla c_I(x)\| \leq b_4, \\ &\|\nabla c_E(x) - \nabla c_E(y)\| \leq r_E\|x-y\|, \quad \|\nabla c_I(x) - \nabla c_I(y)\| \leq r_I\|x-y\|. \end{aligned}$$

In order to analyze the local convergence of Algorithm 3.1, we also need the concept of semismoothness for vector value functions. The concept of semismoothness was originally introduced by Mifflin [21] for functions and extended by Qi and Sun [16] for vector-valued functions. Convex functions, smooth functions and piecewise linear functions are examples of semismooth function. The composition of semismooth functions is still a semismooth function [21].

Let $F : \mathbb{R}^n \to \mathbb{R}^m$ be a locally Lipschitz continuous mapping. Then, from Rademacher's theorem, F is differentiable almost everywhere and the generalized Jacobian [17] is well-defined such that

$$\partial F(x) = Co\left\{ \lim_{x^k \to x, x^k \in D_F} \nabla F(x^k)^T \right\},$$

where Co denotes a convex hull and D_F denotes a set of points at which F is differentiable.

The function F is called semismooth at $x \in \mathbb{R}^n$, if

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

exists for any $h \in \mathbb{R}^n$. The function F is further said to be strongly semismooth at x if F is semismooth at x and for any $V \in \partial F(x+h), h \to 0$,

$$F(x+h) - F(x) - Vh = \mathcal{O}(\|h\|^2).$$

Now we give some important properties of the semismooth functions as follows.

Lemma 4.1 ([5]). *Suppose that $G : \mathbb{R}^n \to \mathbb{R}^m$ is a locally Lipschitz function. Then*

(i) $G(.)$ has generalized Jacobian $\partial G(x)$ as in Clarke [17]. *Also $G'(x;h)$, the directional derivative of G at x in the direction h, exists for any $h \in \mathbb{R}^n$ if G is semismooth at x. Moreover, $G : \mathbb{R}^n \to \mathbb{R}^m$ is semismooth at $x \in \mathbb{R}^n$ if and only if all its component functions are semismooth at $x \in \mathbb{R}$.*

(ii) $G(.)$ is semismooth at x if and only if for any $V \in \partial G(x+h), h \to 0$,

$$\|Vh - G'(x;h)\| = o(\|h\|),$$
$$\|G(x+h) - G(x) - G'(x;h)\| = o(\|h\|).$$

(iii) $G(.)$ is strongly semismooth at x if and only if for any $V \in \partial G(x+h), h \to 0$,

$$\|Vh - G'(x;h)\| = \mathcal{O}(\|h\|^2),$$
$$\|G(x+h) - G(x) - G'(x;h)\| = \mathcal{O}(\|h\|^2).$$

Then we can get the following result for the smoothing function (2.1).

Lemma 4.2. *$q(\mu,\omega)$is a strongly semismooth function on $\mathbb{R}^2$.*

Proof. It is easy to see that if $\rho(s) = e^{-s}(1+e^{-s})^{-2}$, then

$$supp(\rho) = \mathbb{R}, \quad \limsup_{s\to\infty} \rho(s)\cdot|s|^3 = 0 < \infty.$$

So from the Proposition 3.1(vii) in [18], we can know that $q(\mu,\omega)$ is a strongly semismooth function on $\mathbb{R}^2$. □

Theorem 4.1. *Let the smoothing function be* (2.2) *and assume that Assumption* 4.2 *hold. Then*

(i) $H(z)$ is Lipschitz continuous at any $z := (u,x) \in \mathbb{R}_+ \times \mathbb{R}^n$.

(ii) $H(z)$ is strongly semismooth at any $z := (u,x) \in \mathbb{R}_+ \times \mathbb{R}^n$.

(iii) The Jacobi matrix of $H(z), z \in L(z^0)$ is bounded.

Proof. (i) From Assumption 4.2, Lemma 2.3 and the construction of $H(z)$, we can know that all components of $H(z)$ are Lipschitz continuous at any $z := (u,x) \in \mathbb{R}_+ \times \mathbb{R}^n$, so (i) holds.

(ii) From Lemma 4.2, we can know that $q(\mu,\omega)$ is a strongly semismooth function on $\mathbb{R}^2$, and from Assumption 4.2, $\nabla c_E(x), \nabla c_I(x)$ are Lipschitz continuous which implies

$c_i(z) = c_i(x), i \in E \cup I$ are strongly semismooth at $z = (u, x) \in \mathbb{R}_+ \times \mathbb{R}^n$. So from Theorem 19 in [19], we know that $p(z)$ is strongly semismooth at z because the composition of strongly semismooth functions is still a strongly semismooth function. Thus $H(z)$ is strongly semismooth at $z := (u, x) \in \mathbb{R}_+ \times \mathbb{R}^n$.

(iii) If $u > 0$, from (2.6) and (2.8), we can easily get that if $\nabla c_E(x), \nabla c_I(x)$ are bounded, then $H'(z)$ is bounded for any $z := (u, x) \in \mathbb{R}_{++} \times \mathbb{R}^n$. If $u = 0$, the generalized derivative of $\max\{0, c_i(x)\}$ about x_i is $l \cdot \partial c_i(x)/\partial x_i$, where $l \in [0, 1]$. So we can get that the generalized Jacobi matrix of $H(z)$ is bounded when $\nabla c_E(x), \nabla c_I(x)$ are bounded. □

The following theorem shows the global convergence for Algorithm 3.1.

Theorem 4.2. *Suppose that Assumption* 4.1 *is satisfied. Assume that for any* $z^k \in \mathbb{R}_{++} \times \mathbb{R}^n, rank(H'(z^k)) = m + 1$ *and the smoothing function is* (2.2)*. Let* $z^* := (u^*, x^*)$ *be an accumulation point of* $\{z^k\}$ *which is generated by Algorithm* 3.1*, then* z^* *is a solution of* $H(z) = 0$.

Proof. From Assumption 4.1, we can know that the accumulation point z^* of $\{z^k\}$ is nonempty and without loss of generality, we assume $z^* = (u^*, x^*)$ is the limit point of $\{z^k\}$. From Remark 3.1(iii), the sequence $\{\psi(z^k)\}$ is monotonically decreasing and we can get

$$\psi(z^k) \to \psi(z^*).$$

Now we need to show $H(z^*) = 0$. Assume, on the contrary, that $\|H(z^*)\| > 0$, then

$$\beta(z^*) > 0, \quad \psi(z^*) > 0, \quad u^* > 0.$$

Hence $H(z^*)$ is continuously differentiable at z^*. From (3.6), there exists $z^{k+1} = z^* + \lambda_k \Delta z^*, \lambda_k \in (0, 1]$ which satisfies

$$\psi(z^{k+1}) \leq [1 - 2\sigma(1 - \gamma\bar{u})\lambda_k]\psi(z^*) < \psi(z^*) \tag{4.1}$$

for all sufficiently large k. This contradicts the fact the sequence $\{\psi(z^k)\}$ converges to $\psi(z^*) > 0$. Consequently, $H(z^*) = 0$. □

We now state the local quadratic convergence of Algorithm 3.1 in the following theorem.

Theorem 4.3. *Suppose that Assumptions* 4.1 *and* 4.2 *are satisfied. Assume that for any* $z^k \in \mathbb{R}_{++} \times \mathbb{R}^n, rank(H'(z^k)) = m + 1$ *and the smoothing function is* (2.2)*. Let* z^* *be an accumulation point of the infinite sequence* $\{z^k\}$ *generated by Algorithm* 3.1*. If all* $V \in \partial H(z^*)$ *are of full rank* $m + 1$*, then* $\{z^k\}$ *converges to* z^* *quadratically, i.e.,*

$$\|z^{k+1} - z^*\| = \mathcal{O}(\|z^k - z^*\|^2). \tag{4.2}$$

Proof. From Theorem 4.2, we know that z^* is a solution of $H(z)=0$. So it follows from (3.3), (3.5), the strongly semismoothness of $H(z)$, and the boundedness of $\|H'(z)\|$, that for all z^k sufficiently close to z^*,

$$\begin{aligned}
&\|z^k+\Delta z^k-z^*\| \\
&=\|z^k+H'(z^k)^+[-H(z^k)+\beta_k\bar{z}]-z^*\| \\
&\le \|H'(z^k)^+\|\cdot\|H'(z^k)z^k-H(z^k)-H'(z^k)z^*+\beta_k\bar{z})\| \\
&=\mathscr{O}\left(\|H(z^k)-H(z^*)-H'(z^k)(z^k-z^*)\|+\beta_k\bar{u}\right) \\
&=\mathscr{O}(\|z^k-z^*\|^2)+\mathscr{O}(\psi(z^k)).
\end{aligned} \tag{4.3}$$

Because H is strongly semismooth at z^*, H is locally Lipschitz continuous around z^*, for all z^k sufficiently close to z^*, we have

$$\psi(z^k)=\|H(z^k)\|^2=\mathscr{O}(\|z^k-z^*\|^2). \tag{4.4}$$

So from (4.3), we have

$$\|z^k+\Delta z^k-z^*\|=\mathscr{O}(\|z^k-z^*\|^2). \tag{4.5}$$

Then we only need to prove $\lambda_k=1$ when k is sufficiently large and z^k is sufficiently close to z^*. First, we can verify that for all z^k is sufficiently close to z^*, when k is sufficiently large,

$$\|z^k-z^*\|=\mathscr{O}(\|H(z^k)-H(z^*)\|). \tag{4.6}$$

This is because

$$\|z^k-z^*\|\le\|\Delta z^k\|+\|z^k+\Delta z^k-z^*\|. \tag{4.7}$$

Then by (3.5), we get

$$\begin{aligned}
\|\Delta z^k\| &\le \|(H'(z^k))^+\|(\|H(z^k)\|+\|\beta_k\bar{z}\|) \\
&\le \|(H'(z^k))^+\|(1+\gamma\bar{u}\|H(z^k)\|)\|H(z^k)\| \\
&\le c\|H(z^k)\|.
\end{aligned} \tag{4.8}$$

It follows from (4.5) and (4.7) that

$$\|z^k-z^*\|\le\|\Delta z^k\|+o(\|z^k-z^*\|); \tag{4.9}$$

and from (4.8), we have

$$\|z^k-z^*\|\le c_1\|H(z^k)\|=c_1\|H(z^k)-H(z^*)\|, \tag{4.10}$$

which gives (4.6).

From (4.4)-(4.6), we have

$$\begin{aligned}
&\psi(z^k+\Delta z^k)=\|H(z^k+\Delta z^k)\|^2 \\
&=\mathscr{O}(\|z^k+\Delta z^k-z^*\|^2)=o(\|z^k-z^*\|^2) \\
&=o(\|H(z^k)-H(z^*)\|^2)=o(\psi(z^k)).
\end{aligned} \tag{4.11}$$

Consequently, when k is sufficiently large, we have

$$z^{k+1} = z^k + \Delta z^k, \tag{4.12}$$

i.e., $\lambda_k = 1$. Then from (4.5), we get

$$\|z^{k+1} - z^*\| = \mathscr{O}(\|z^k - z^*\|^2).$$

This completes the proof of (4.2). □

5. Numerical results

In this section, we carry out some numerical experiments for Algorithm 3.1. The algorithm was implemented in MATLAB. The problems that we tested are the constraint sets of a subset of the test examples for nonlinear programming from [20]. The parameters in the algorithm were

$$\delta = 0.5, \quad \sigma = 0.5 \times 10^{-4}, \quad \gamma = 0.2 \times \min\left\{1, \frac{1}{\|\bar{u}\|}\right\}, \quad \bar{u} = 0.1, \quad z_0 = (\bar{u}, \mathbf{0}),$$

and the stopping rule is $\|H(z)\| \leq 10^{-6}$.

In Table 1, columns 1-4 give the data of the problem. In particular, the first column gives the problem name. The second column gives the dimension (number of variables) of the problem. The third and fourth columns give the number of equalities and the number of inequalities, respectively. In the fifth and sixth columns we list, respectively, the average number of iterations and the average number of function evaluations needed by Algorithm 3.1 to converge from different starting points to points that satisfy the stopping criterion. In the seventh and eighth columns we list the corresponding results of the single-model trust-region method given in [7]. In the ninth and tenth columns we list the corresponding results of the multimodel trust-region method [7]. S-N denotes the smoothing Newton-like algorithm, S-T denotes the single-model trust-region algorithm in [7], M-T denotes the multimodel trust-region algorithm in [7].

Table 1: Numerical results for the test problem.

Problem data				S-N		S-T		M-T	
Prob. name	n	$\|E\|$	$\|I\|$	iter.	nfunc	iter	nfunc	iter.	nfunc
HS10	2	0	1	5.2	9.4	11.2	12.2	11.2	12.2
HS11	2	0	1	4.4	7.8	9.8	12.2	9.8	12.2
HS12	2	0	1	4	7	7.6	8.6	7.6	8.6
HS14	2	1	1	3.8	6.6	9.6	11.2	9.6	11.2
HS22	2	0	2	6.2	11.5	10.6	13	9.6	12
HS29	3	0	1	3	5	7.2	8.2	7.2	8.2
HS43	4	0	3	5.2	9.4	10.8	14	4	5
HS113	10	0	8	4.5	8	10	12.4	14.8	19.8

All problems we selected are constraints of nonlinear programming problems, so we get a feasible point of a system of equalities and inequalities by the smoothing Newton-like method. From the numerical results, we can see that the number of function evaluations and the iterations obtained by our smoothing Newton-like method are less than those obtained by the trust-region algorithms.

In summary, we have proposed a new smoothing Newton-like method for solving the system of equalities and inequalities. This smoothing Newton-like method only needs to solve one system of linear equations and to perform one line search at each iteration. It keeps the good structure of the classical Newton method and has good convergence under mild conditions.

Acknowledgments This work is supported by Guangdong Provincial Zhujiang Scholar Award Project, National Science Foundation of China (10671163, 10871031), the National Basic Research Program under the Grant 2005CB321703, and Scientific Research Fund of Hunan Provincial Education Department (06A069,06C824). The author would like to thank the referees for the helpful suggestions.

References

[1] J.E. Dennis and R.B. Schnabel, *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Englewood Cleffs, Prentice-Hall, NJ, 1983.

[2] B.N. Pshenichnyi, *Newton's method for the solution of systems of equalities and inequalities*, Math. Notes Acad. Sci. USSR, 8(1970), pp. 827–830.

[3] S.M. Robinson, *Extension of Newton's method to nonlinear functions with values in a cone*, Numer. Math., 19(1972), pp. 341–347.

[4] J.W. Daniel, *Newton's method for nonlinear inequalities*, Numer.Math., 40(1973), pp. 381–387.

[5] B.T. Polyak, *Gradient methods for solving equations and inequations*, UDSSR Comput. Math., 4(1964), pp. 17–32.

[6] J. Burke and S.P. Han, *A Gauss-Newton approach to solving generalized inequalities*, Math. Oper. Res., 11(1986), pp. 632–643.

[7] J.E. Dennis, M. El-Alem and K. Williamson, *A trust-region approach to nonlinear systems of equalities and inequalities*, SIAM J. Optim., 9(1999), pp. 291–315.

[8] X.J. Tong and S. Zhou, *A Trust region algorithm for nonlinear problems of equalities and inequalities*, Journal on Numerical Methods and Computer Applications,3(2001), pp. 53–62.

[9] L. Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Math. Oper. Res.,18(1993), pp. 227–244.

[10] F. Facchinei and J.S. Pang, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer-Verlag, New York, 2003.

[11] S.C. Billups, S.P. Dirkse and M.C. Ferris, *A comparison of algorithms for large-scale mixed complementarity problems*, Coput. Optim. Appl., 7(1997), pp.3–25.

[12] X. Chen, L. Qi and D. Sun, *Global and superlinear convergence of the smoothing Newton method and its application to general box constrained variational inequalities*, Mathematics of Computation, 67(1998), pp. 519–540.

[13] L. QI, D. SUN AND G. ZHOU, *A new look at smoothing Newton methods for nonlinear complementarity problems and box constrained variational inequalities*, Math.Program., A87(2000), pp. 1–35.
[14] C. CHEN AND O.L. MANGASARIAN, *A class of smoothing functions for nonlinear and mixed complementarity problems*, Comput. Optim. Appl., 5(1996), pp. 97–138.
[15] Z.-H. HUANG AND H. WANG, *Smoothing-type algorithm for solving linear programs by using an augmented complementarity problem*, Applied Mathematics and Computation, 177(2006), pp. 330–345.
[16] L. QI AND J. SUN, *A Nonsmooth version of Newton's method*, Math. Program., 58(1993), pp. 353–367.
[17] F.H. CLARKE, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York, 1983.
[18] L. QI AND D. SUN, *Smoothing functions and smoothing Newton method for complementarity and variational inequalitiey problems*, Journal of Optimization Theory and Applications, 113(2002), pp. 121–147.
[19] A. FISCHER, *Solution of monotone complementarity problems with locally Lipschitzian functions*, Math. Program., 76(1997), pp. 513–532.
[20] W. HOCK AND K. SCHITTKOWSKI, *Test examples for nonlinear programming codes*, Lecture Notes in Econom. and Math. Systems 187, Springer-Verlag, Berlin, 1981.
[21] R. MIFFLIN, *Semismooth and semiconvex functions in constrained optimization*, SIAM J.Control Optim., 15(1977), pp. 957–972.
[22] H.F. WALKER, *Newton-like methods for undertermined systems*, in: I.L. Allgower, K. Georg, eds.,Computational solution of Nonlinear systems of equations, Lecture Notes in Applied Mathematics, 26(1990), pp. 679–699.
[23] Y. YUAN AND W.Y. SUN, *Optimization Theorems and Methods*, Science Press, Beijing, 2001.