J. Appl. Math. & Informatics Vol. **29**(2011), No. 3 - 4, pp. 653 - 668
Website: http://www.kcam.biz

A HYBRID METHOD FOR NCP WITH P_0 FUNCTIONS[†]

QIAN ZHOU AND YI-GUI OU[*]

Abstract. This paper presents a new hybrid method for solving nonlinear complementarity problems with P_0-functions. It can be regarded as a combination of smoothing trust region method with ODE-based method and line search technique. A feature of the proposed method is that at each iteration, a linear system is only solved once to obtain a trial step, thus avoiding solving a trust region subproblem. Another is that when a trial step is not accepted, the method does not resolve the linear system but generates an iterative point whose step-length is defined by a line search. Under some conditions, the method is proven to be globally and superlinearly convergent. Preliminary numerical results indicate that the proposed method is promising.

AMS Mathematics Subject Classification : 90C33, 90C30, 65H10.
Key words and phrases : nonlinear complementarity problem, ODE-based method, smoothing trust region method, global convergence, superlinear convergence.

1. Introduction

The nonlinear complementarity problem, denoted by NCP(F), is to find a vector $x \in R^n$, such that

$$x \geq 0,\ F(x) \geq 0,\ x^T F(x) = 0 \tag{1.1}$$

where $F : R^n \to R^n$ is assumed to be continuously differentiable.

The nonlinear complementarity problem has many important applications in engineering, economy equilibrium models and game theory, and many numerical methods have been developed to solve NCP(F), such as interior-point method, nonsmooth Newton method, smoothing method, projection method, and so on, see Ref[1] for a survey. In this paper, we concentrate ourself on the smoothing

Received June 11, 2010. Revised August 11, 2010. Accepted August 30, 2010. [*]Corresponding author. [†]This work was supported by NNSF of China (No.10861006) and NSF of Hainan University(No. HD09XM87).

method whose basic idea is to approximate nonsmooth problems by a sequence of smooth problems.

Although much progress has been made in smoothing methods (see Refs[2-4] and references therein), most of them proposed for solving NCP(F) are based on line search strategy, methods based on trust region technique are relatively less. It is well known that in many cases, trust region methods are more reliable and more robust than line search methods and have been efficiently applied to solve optimization problems, see Ref[5] for details. Therefore, we expect that some more efficient methods can be designed to solve NCP(F) by incorporating the trust region technique into smoothing methods, since NCP(F) can be reformulated as a system of nonlinear equations and thus a related optimization problem by using some NCP functions. By making use of this approach, Yang and Qi[6] proposed a smoothing trust region method for solving NCP(F). Zhou[7] presented a smothing method for NCP(F), which combines trust region techniques with a conic model. Long et al[8] also provided a smoothing method in which the trust region and filter techniques are employed to tackle NCP(F). However, the disadvantage of those methods is that at each iteration, the quadratic or conic trust region subproblem may be resolved several times before obtaining an acceptable trial step, thus the average cost of computation per iteration might be expensive, especially for large-scale problems.

In recent years, some new ideas have been proposed in the literature, which combine elements of trust region methods with elements of line search methods, see Refs[9] and [10] for details. An advantage of them is that when a trial step is not accepted, the new methods perform a line search to find an iterative point instead of resolving the quadratic trust region subproblem. Therefore, the new methods require less computation than classical trust region methods. Based on this new idea, Qu et al[11] developed a new smoothing trust region method for NCP(F), in which nonmonotone techniques, conic models and a line search are combined to generate a new iterative point at each iteration. Hence this new hybrid method has both advantages of line search method and nonmonotone trust region method. However, those new methods in Refs[9-11] still need to solve the quadratic or conic trust region subproblem, which is usually a difficult task, especially for a nonconvex trust region subproblem.

As a strategy of optimization, the ODE-based methods were first introduced into solving unconstrained optimization by Brown et al.[12]. A feature of the ODE-based methods is that at each iteration, a linear system is solved to obtain a trial step, which has something in common with Leverberg-Marquardt-type method [5] for unconstrained optimization. By means of extensive experiments, Brown et al[12] have show that, when suitably implemented, ODE-based methods can compare very favorably with conventional Newton and quasi-Newton algorithms as regards reliability, accuracy and efficiency, especially for highly nonlinear minimization problems with narrow and curving valleys. The disadvantage of the ODE-based methods is that at each iteration, the linear system may be resolved several times before obtaining an acceptable trial step, thus increasing the average cost per iteration of the methods.

Motivated by the above observations, we propose a new hybrid method for solving NCP(1.1), which combines ODE-based methods with smoothing trust region methods and line search technique. A feature of the proposed algorithm is that at each iteration, a system of linear equations is only solved once to obtain a trial step, thus avoiding solving a quadratic or conic trust region subproblem. Another is that when a trial step is not accepted, the new methods perform a line search to find an iterative point instead of resolving the linear system. From a computational point of view, this approach may reduce computational complexity and thus improve computational efficiency. Under some suitable conditions, global convergence and local superlinear convergence is established if F is a P_0 function.

This paper is organized as follows. Section 2 recalls some existing results. In section 3, we describe our algorithm for solving NCP(F) and prove that it is well defined. Global convergence and locally superlinear convergence are established in sections 4 and 5, respectively. In section 6, numerical experiments are presented.

2. Preliminaries

In this section we summarize some existing results, which will be used in the sequel.

Let $\varphi : R^2 \to R$ be Fischer-Burmeister function(see Ref[1]) defined by

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

By using the function φ, NCP(1.1) can be equivalently reformulated as a system of nonlinear equations

$$\Phi(x) = 0$$

where $\Phi : R^n \to R^n$ is defined by

$$\Phi\,(x) = \begin{pmatrix} \varphi(x_1, F_1(x)) \\ \vdots \\ \varphi(x_n, F_n(x)) \end{pmatrix}$$

Then the natural merit function $\psi : R^n \to R$ defined by

$$\psi(x) = \frac{1}{2}\Phi(x)^T\Phi(x) \tag{1.2}$$

can be used to globalize the proposed algorithm. A favorable property of ψ is that it is continuously differentiable on the whole space R^n, although Φ itself is nonsmooth.

Since φ is a NCP-function, i.e.,

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

we can deduce from the definition of mapping Φ that

$$x^* \text{ solves NCP}(1.1) \Longleftrightarrow \Phi(x^*) = 0$$

whenever NCP(1.1) has a solution. However, the system $\Phi(x) = 0$ is nonsmooth. Therefore, we will use Kanzow's[13] smoothing approximation

$$\varphi_\varepsilon(a, b) = \sqrt{a^2 + b^2 + 2\varepsilon} - a - b, \ \varepsilon > 0$$

to approximate the Fischer-Burmeister function. The corresponding smooth operator $\Phi_\varepsilon : R^n \to R^n$ is defined by

$$\Phi_\varepsilon\,(x) = \left(\begin{array}{c} \varphi_\varepsilon(x_1, F_1(x)) \\ \vdots \\ \varphi_\varepsilon(x_n, F_n(x)) \end{array} \right) \tag{1.3}$$

Denote

$$\psi_\varepsilon(x) = \frac{1}{2}\Phi_\varepsilon(x)^T \Phi_\varepsilon(x)$$

NCP(1.1) can be approximated by the following nonlinear least squares problem

$$\min_{x \in R^n} \psi_\varepsilon(x) \tag{1.4}$$

From the above discussion, it immediately follows that problem (1.4) is equivalent to NCP(1.1) as $\varepsilon \to 0$.

Let $G : R^n \to R^n$ be locally Lipschitz continuous and D_G be the set of differentiable points of G. The generalized Jacobian of G at x in the sense of Clarke[14] is defined by

$$\partial G(x) = conv \partial_B G(x)$$

where

$$\partial_B G(x) = \{H \in R^{n\times n} | H = \lim_{x^k \to x, x^k \in D_G} \nabla G(x_k)\}$$

is called as B-differential of G at $x \in R^n$. Usually, ∂G is difficult to calculate, so we use the generalized **Jacobian** of the form

$$\partial_C G(x)^T := \partial G_1(x) \times \partial G_2(x) \times \cdots \times \partial G_n(x) \in R^{n\times n}$$

which can be seen as a special case of the C-differential operator discussed by Qi, see Ref[1] for details.

A local Lipschitz function $G : R^n \to R^n$ is called semismooth at $x \in R^n$ if G is directionally differentiable at x and for any $V \in \partial G(x + d)$ and $d \to 0$,

$$G'(x; d) - V d = o(\|d\|)$$

A function $G : R^n \to R^n$ is said to be a P_0 **function** if for all $x, y \in R^n$ with $x \neq y$,

$$\max_{i:\, x_i \neq y_i} (x_i - y_i)(G_i(x) - G_i(y)) \geq 0$$

Lemma 2.1 [15]. For all $x \in R^n$ and any $\varepsilon \geq 0$, we have

$$\|\Phi(x) - \Phi_\varepsilon(x)\| \leq \kappa\sqrt{\varepsilon}$$

where $\kappa = \sqrt{2n}$.

Lemma 2.2 [15]. Assume that $\{x^k\} \subset R^n$ is a convergent sequence with a limit point $x^* \in R^n$. Then the function $\Phi(x)$ is semismooth, which implies that for any $V_k \in \partial_C \Phi(x^k)$,

$$\|\Phi(x) - \Phi(x^*) - V_k(x_k - x^*)\| = o(\|x_k - x^*\|)$$

Lemma 2.3 [15]. Let $x \in R^n$ be arbitrary but fixed. Assume that x is not a solution of NCP(F). Let us define the constants

$$\gamma(x) = \max_{i \neq \beta(x)} \{\|x_i e_i + F_i(x) \nabla_i F(x)\|\} \geq 0$$

and

$$\alpha(x) = \max_{i \neq \beta(x)} \{x_i^2 + F_i(x)^2\} > 0$$

where $\beta(x) = \{i | x_i = F_i(x) = 0\}$. Let $\delta > 0$ be given, and define

$$\bar{\varepsilon}(x, \delta) = \begin{cases} 1, & \text{if } \frac{n\gamma(x)^2}{\delta^2} - \alpha(x) \leq 0, \\ \frac{\alpha(x)^2}{2} \cdot \frac{\delta^2}{n\gamma(x)^2 - \delta^2 \alpha(x)}, & \text{otherwise} \end{cases}$$

Then

$$dist(\Phi_\varepsilon^{'}(x),\ \partial_C \Phi(x)) \leq \delta$$

for all ε such that $0 < \varepsilon \leq \bar{\varepsilon}(x, \delta)$.

Notation. Throughout this paper, the symbol $\|\cdot\|$ denotes the Euclidean norm or the subordinate matrix norm. For a continuously differentiable mapping $G : R^n \to R^m$, we denote its Jacobian at a point $x \in R^n$ by $G'(x)$, whereas $\nabla G(x)$ denotes the transposed Jacobian.

3. Algorithm

In this section, we shall propose an ODE-based smoothing trust region(**STR**) algorithm for solving NCP(1.1) and prove that the proposed algorithm is well defined.

Define

$$\begin{aligned} Q_k(d) &= \tfrac{1}{2}\|\Phi_{\varepsilon_k}(x^k) + \nabla\Phi_{\varepsilon_k}(x^k)^T d\|^2 \\ &= \psi_{\varepsilon_k}(x^k) + \nabla\psi_{\varepsilon_k}(x^k)^T d + \tfrac{1}{2} d^T \nabla\Phi_{\varepsilon_k}(x^k) \nabla\Phi_{\varepsilon_k}(x^k)^T d \end{aligned}$$

Our algorithm for solving NCP(1.1) is stated as follows and then some remarks are given.

Algorithm STR

Step 0. Given η, ν, r, μ satisfying $0 < \eta < 1, \nu > 0, 0 < r < 1, 0 < \mu < 1$. Choose $x^0 \in R^n, h_0 > 1$. Set $\beta_0 = \|\Phi(x^0)$, $C_0 = (1 + \mu)\|\Phi(x^0)\|$, $\kappa = \sqrt{2n}$, $\varepsilon_0 = (\frac{\mu}{2C_0\kappa}\beta_0^2)^2$, and $k := 0$.

Step 1. Compute the solution $d_k \in R^n$ of the following linear equations

$$(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k} I)d = -\nabla\Phi_{\varepsilon_k}(x^k)\Phi_{\varepsilon_k}(x^k) \qquad (3.1)$$

Step 2. Calculate

$$r_k = \frac{Ared_k}{\Pr ed_k} \tag{3.2}$$

where

$$\begin{array}{rcl} Ared_k & = & \psi_{\varepsilon_k}(x^k) - \psi_{\varepsilon_k}(x^k + d_k) \\ Pred_k & = & \psi_{\varepsilon_k}(x^k) - Q_k(d_k) \end{array}$$

If $r_k \geq r$, then set $x^{k+1} = x^k + d_k$(**successful iteration**), $h_{k+1} = 2h_k$; Otherwise, set $h_{k+1} = \frac{1}{2}h_k$ and $x^{k+1} = x^k + \alpha_k d_k$, where α_k is calculated by the Armijo line search rule (see Remark 3.2 below).
Step 3. If $\nabla\psi(x^{k+1}) = 0$, stop.
Step 4. If $\|\Phi(x^{k+1})\| \leq \max\{\eta\beta_k, \mu^{-1}\|\Phi(x^{k+1}) - \Phi_{\varepsilon_k}(x^{k+1})\|\}$, then set $\beta_{k+1} = \|\Phi(x^{k+1})\|$ and choose ε_{k+1} such that

$$0 < \varepsilon_{k+1} \leq \min\{(\frac{\mu}{2C_0\kappa}\beta_{k+1}^2)^2,\ \frac{\varepsilon_k}{4},\ \bar{\varepsilon}(x^{k+1}, \nu\beta_{k+1})\} \tag{3.3}$$

where $\bar{\varepsilon}(\cdot,\cdot)$ is defined in Lemma 2.3; otherwise, set $\beta_{k+1} = \beta_k, \varepsilon_{k+1} = \varepsilon_k$,
Step 5. Set $k := k + 1$, and return to Step1.

Remark 3.1. The matrix $\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k}I$ is positive definite for any $h_k > 0$, thus the linear equations (3.1) is always solvable.

Remark 3.2. **Armijo line search rule**[2]: Given $\rho \in (0, 1)$ and $\sigma \in (0, \frac{1}{2})$. Set $\alpha_k = \rho^{l_k}$, where l_k is the smallest nonnegative integer such that

$$\psi_{\varepsilon_k}(x^k + \rho^l d_k) \leq \psi_{\varepsilon_k}(x^k) + \sigma\rho^l\nabla\psi_{\varepsilon_k}(x^k)^T d_k \tag{3.4}$$

Remark 3.3. Algorithm STR is different from Yang and Qi's method[6], which obtains a trial step by solving a quadratic trust region subproblem at each iteration and keeps steplength being null(i.e., $x^{k+1} = x^k$) when a trial step is not accepted. Also it is different from Qu's method[12], in which a conic trust region subproblem need to be solved per iteration. Since there is few efficient method for solving a conic model with trust region bound so far, it is not easy to implement.

Denote

$$N = \{0, 1, 2, \cdots\}$$

Without loss of generality, we assume that Algorithm STR does not terminate after a finite number of iterations, i.e., $\nabla\psi(x^k) \neq 0$ for all $k \in N$. This implies

$$\Phi(x^k) \neq 0, \forall k \in N \tag{3.5}$$

Define the index set

$$\begin{array}{rcl} K: & = & \{0\} \cup \{k|\ \|\Phi(x^k)\| \leq \max\{\eta\beta_{k-1}, \mu^{-1}\|\Phi_{\varepsilon_k}(x^k) - \Phi(x^k)\|\}\} \\ & = & \{k_0 = 0 < k_1 < k_2 < \cdots\} \end{array} \tag{3.6}$$

By Lemma 2.1, Lemma 2.3 and the rules of Step 4 in Algorithm STR, it is easy to deduce that

$$\|\Phi_{\varepsilon_k}(x^k) - \Phi(x^k)\| < \mu\|\Phi(x^k)\|,\ \forall k \in N \tag{3.7}$$

and

$$dist(\Phi'_{\varepsilon_k}(x^k), \partial_C \Phi(x^k)) \leq \nu \|\Phi(x^k)\|, \ \forall k \in K, k \geq 1 \tag{3.8}$$

Lemma 3.1. For Algorithm STR, we have
(i) $Pred_k = \frac{1}{2} d_k^T (\nabla \Phi_{\varepsilon_k}(x^k) \nabla \Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k} I) d_k + \frac{1}{2h_k} \|d_k\|^2$
(ii) $Pred_k - Ared_k = O(\|d_k\|^2)$

Lemma 3.2 [3]. $\nabla \Phi_\varepsilon(x)$ is nonsingular for all $x \in R^n$ and $\varepsilon > 0$ if F is a P_0 function.

The following Lemma shows that Algorithm STR is well-defined.

Lemma 3.3. If F is a P_0 function, then Algorithm STR is well-defined, i.e., $Pred_k > 0$, and there exists a finite nonnegative integer l such that (3.4) holds for any unsuccessful step.

Proof. To verify the first assertion, we only need to prove

$$Pred_k \neq 0, \ \forall k \in N$$

since $Pred_k \geq 0$ by using Lemma 3.1(i) and Remark 3.1. By contradiction, assume that there exists some $l \in N$ such that $Pred_l = 0$. This implies that $d_l = 0$. Therefore, it follows from Lemma 3.2 and (3.1) that

$$\Phi_{\varepsilon_l}(x^l) = 0$$

This is a contradiction to (3.7).

Now we show the existence of α_k. From (3.1) and Lemma 3.1, we have

$$\nabla \psi_{\varepsilon_k}(x^k)^T d_k = -d_k^T (\nabla \Phi_{\varepsilon_k}(x^k) \nabla \Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k} I) d_k \tag{3.9}$$

whenever $\nabla \psi_{\varepsilon_k}(x^k) \neq 0$(or equivalently $d_k \neq 0$). This implies that d_k is a descent direction of $\psi_{\varepsilon_k}(x)$ at x^k. Therefore, the Armijo line search rule is finite terminating. This proof is completed. □

4. Global convergence

In this section, we will study the global convergence of Algorithm STR. To this end, we make the following assumption.

A1 The level set $L_0 = \left\{ x \in R^n \, \middle| \psi(x) \leq (1+\mu)^2 \psi(x^0) \right\}$ is bounded.

Remark 4.1. If F is a uniform P-function or, more generally, an R_0-function, then the level set L_0 as defined in Assumption A1 is compact and thus L_0 is bounded, see Ref[1] for details.

Lemma 4.1. If F is a P_0 function, then the sequence $\{x^k\}$ generated by Algorithm STR remains in the level set L_0.

Proof. The proof is similar to that of Proposition 4.1 in Ref[6]. Here we omit it.

Lemma 4.2. Assume that F is a P_0 function. If Assumption **A1** holds, then the sequence $\{x^k\}$ generated by Algorithm STR satisfies

$$\lim_{k\to\infty} \inf \|\nabla\psi_{\tilde{\varepsilon}}(x^k)\| = 0 \tag{4.1}$$

where $\tilde{\varepsilon}$ is an accumulation point of ε_k.

Proof. Suppose that (4.1) does not hold. Then there exists $\epsilon > 0$ such that

$$\|\nabla\psi_{\tilde{\varepsilon}}(x^k)\| = \|\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k)\| \geq \epsilon,\ \forall k \tag{4.2}$$

We consider two different cases.

Case 1. there exists a subsequence of $\{h_k\}$, without loss of generality, we still denote it by $\{h_k\}$, such that $\{h_k\} \to 0^+$.

From Lemma 4.1 and Assumption A1, It follows that there exists subsequence of $\{x^k\}$ corresponding to $\{h_k\}$, without loss of generality, we still denote it by $\{x^k\}$, such that $x^k \to \bar{x}$. This together with (4.2) implies that $\nabla\psi_{\tilde{\varepsilon}}(\bar{x}) = \nabla\Phi_{\tilde{\varepsilon}}(\bar{x})\Phi_{\tilde{\varepsilon}}(\bar{x})^T \neq 0$. By the rule of defining h_k in step5, we have

$$\psi_{\tilde{\varepsilon}}(x^k) - \psi_{\tilde{\varepsilon}}(x^k + d_k) < r(-\nabla\psi_{\tilde{\varepsilon}}(x^k)^T d_k - \frac{1}{2}d_k^T\nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T d_k) \tag{4.3}$$

where $d_k = -h_k(h_k\nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T + I)^{-1}\nabla\psi_{\tilde{\varepsilon}}(x^k)$. Obviously, $\|d_k\| \to 0$ as $h_k \to 0^+$. Thus we have by using (4.3) that

$$\frac{\psi_{\tilde{\varepsilon}}(x^k) - \psi_{\tilde{\varepsilon}}(x^k - h_k G_k^{-1}\nabla\Phi_{\tilde{\varepsilon}}(x^k)}{h_k} < r\lambda_k - \frac{r}{2}h_k\lambda_k^2$$

where $G_k = h_k\nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T + I$, and $\lambda_k = \nabla\psi_{\tilde{\varepsilon}}(x^k)^T G_k^{-1}\nabla\psi_{\tilde{\varepsilon}}(x^k)$. Let $h_k \to 0^+$ on the above inequality, we obtain by using Lemma 4.1 and Assumption A1 that

$$\nabla\psi_{\tilde{\varepsilon}}(\bar{x})^T\nabla\psi_{\tilde{\varepsilon}}(\bar{x}) \leq r\nabla\psi_{\tilde{\varepsilon}}(\bar{x})^T\nabla\psi_{\tilde{\varepsilon}}(\bar{x})$$

This is a contradiction, since $0 < r < 1$ and $\nabla\psi_{\tilde{\varepsilon}}(\bar{x}) \neq 0$.

Case 2. there exists a constant $c > 0$ such that $h_k \geq c$ for all k.

Define the index set $S = \{k | r_k \geq r\}$. Under Case 2, we claim that the set S is infinite. Suppose that this assertion does not hold, then there exists an integer $k_s > 0$ such that

$$r_k < r,\ \forall k \geq k_s$$

This together with the rule of step 2 implies that $h_{k+1} = \frac{1}{2}h_k$ for all $k \geq k_s$, i.e.,

$$h_k \to 0^+\ (k \to +\infty)$$

This contradicts the assumption of Case 2. Hence the set S is infinite.

By Lemma 3.1 and (3.1), we have that for all $k \in S$,

$$\begin{array}{rcl} Ared_k & = & \psi_{\tilde{\varepsilon}}(x^k) - \psi_{\tilde{\varepsilon}}(x^{k+1}) \geq rPred_k \\ & \geq & \frac{r}{2}d_k^T(\nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T + \frac{1}{h_k}I)d_k \\ & = & -\frac{r}{2}(\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k))^T d_k \geq 0 \end{array} \tag{4.4}$$

From (3.4), (3.9) and the rules of step 2, it follows that the sequence $\{\psi_{\tilde{\varepsilon}}(x^k)\}$ is monotone decreasing. Moreover, it is bounded below. Thus $\lim\limits_{k\to\infty} \psi_{\tilde{\varepsilon}}(x^k)$ exists. This together with (4.4) implies that

$$(\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k))^T d_k \to 0,\ as\ k\to\infty,\ k\in S \tag{4.5}$$

On the other hand, it follows from Assumption A1 that there exists a constant $C>0$ such that

$$\|\nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T\| \leq C$$

Let $A_k = \nabla\Phi_{\tilde{\varepsilon}}(x^k)\nabla\Phi_{\tilde{\varepsilon}}(x^k)^T + \frac{1}{h_k}I$. Then we have by (3.1) and (4.2) that

$$\begin{array}{rcl}\|(\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k))^T d_k\| & = & (\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k))^T A_k^{-1}(\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k)) \\ & \geq & \frac{\|\nabla\Phi_{\tilde{\varepsilon}}(x^k)\Phi_{\tilde{\varepsilon}}(x^k)\|^2}{\|A_k\|} \\ & \geq & \frac{\epsilon^2}{C+\frac{1}{c}}.\end{array}$$

a contradiction to (4.5)

Since both Case 1 and Case 2 lead to a contradiction, (4.1) is proven. This proof is complete. □

Based on the above conclusions, we now analyze the global convergence of Algorithm STR.

Lemma 4.3. Assume that F is a P_0 function. If Assumption **A1** holds, then the index set K defined by (3.6) is infinite, and

$$\lim_{k\to\infty}\varepsilon_k = 0,\ \lim_{k\to\infty}\Phi(x^k) = 0, \text{ and } \lim_{k\to\infty}\Phi_{\varepsilon_k}(x^k) = 0$$

Proof. We first prove that the set K is infinite. By contradiction, assume that K is finite. Let $\hat{k}$ be the largest number in K. Then for all $k\geq\hat{k}$, $\varepsilon_k = \varepsilon_{\hat{k}}$ and $\beta_k = \beta_{\hat{k}}$. Denote

$$\tilde{\varepsilon} = \varepsilon_{\hat{k}},\ \tilde{\beta} = \beta_{\hat{k}},\ q(x) = \Phi(x^k) - \Phi_{\tilde{\varepsilon}}(x^k)$$

Then for all $k\geq\hat{k}$, we have

$$\|\Phi(x^k)\| > \max\{\eta\tilde{\beta},\ \mu^{-1}\|q(x^k)\|\} \tag{4.6}$$

and

$$\Phi(x^k) = \Phi_{\tilde{\varepsilon}}(x^k) + q(x^k) \tag{4.7}$$

From Lemma 4.2 and Assumption A1, it follows that there exists at least an accumulation point $\bar{x}\in L_0$ of $\{x^k\}$ such that

$$\nabla\psi_{\tilde{\varepsilon}}(\bar{x}) = 0 \tag{4.8}$$

Assume that this subsequence $\{x_k\}_{k\in K_1}$ converges to $\bar{x}$. By (4.8) and Lemma 3.2, we have

$$\{\Phi_{\tilde{\varepsilon}}(x^k)\}_{k\in K_1} \to \Phi_{\tilde{\varepsilon}}(\bar{x}) = 0$$

Hence there exists $\tilde{k}\geq\hat{k}$ such that for all $k\in K_1$ with $k\geq\tilde{k}$

$$\|\Phi_{\tilde{\varepsilon}}(x^k)\| \leq (1-\mu)\eta\tilde{\beta} \tag{4.9}$$

This together with (4.6) and (4.7) implies that for all $k \in K_1$ with $k \geq \tilde{k}$

$$\|\Phi_{\tilde{\varepsilon}}(x^k)\| \leq (1-\mu)\|\Phi(x_k)\| \leq (1-\mu)(\|\Phi_{\tilde{\varepsilon}}(x^k)\| + \|q(x^k)\|)$$

i.e.,

$$\|\Phi_{\tilde{\varepsilon}}(x^k)\| < (\mu^{-1}-1)\|q(x^k)\| \tag{4.10}$$

which means

$$\|\Phi(x^k)\| \leq \|\Phi_{\tilde{\varepsilon}}(x^k)\| + \|q(x^k)\| < \mu^{-1}\|q(x^k)\|, \ \forall k \in K_1 \text{ and } k \geq \tilde{k}$$

This is contradiction to (4.6). Hence the set K is infinite.

Next, $\{\varepsilon_k\} \to 0$ follows immediately from the updating rule of ε_k and the fact that the set K is infinite. Moreover, similar to the proof of Proposition 4.1 in Ref[6], we deduce

$$\|\Phi(x^k)\| \leq \gamma^j(1+\mu)\|\Phi(x^0)\|, \quad \text{as } k_j \leq k < k_{j+1}. \tag{4.11}$$

where $\gamma = \max\{\frac{1}{2},\ \eta\}$ and k_j is the largest number in K such that $k_j \leq k$. Since the set K is infinite, it follows from (4.11) and (3.7) that

$$\lim_{k\to\infty} \Phi(x^k) = 0, \text{ and } \lim_{k\to\infty} \Phi_{\varepsilon_k}(x^k) = 0$$

This proof is complete. □

Remark 4.2. In fact, Lemma 4.3 is still true if Assumption A1 is replaced by the condition that there exists at least an accumulation point in the sequence $\{x_k\}$. Here we omit its proof, since it is similar to the proof of Lemma 4.3 by slight modification.

As a consequence of the above Lemma 4.3, we get the following global convergence result.

Theorem 4.4. Assume that F is a P_0 function and Assumption A1 holds. Then every accumulation point of the sequence $\{x^k\}$ generated by Algorithm STR is the solution of NCP(1.1).

5. Superlinear convergence

In this section, we will analyze the local convergence of Algorithm STR under the following assumption.

A2 There exists $m > 0$ such that

$$d^T(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k}I)d \geq m\|d\|^2, \quad \forall d \in R^n, \ \forall k = 1, 2, \cdots.$$

We first give some useful results.

Lemma 5.1. Suppose that F is a P_0 function and that the set K_2 is an infinite subset of the index set K such that $\{x^k\}_{k\in K_2}$ converges to x^*. Suppose also that Assumption A1 holds. If all $V \in \partial_C\Phi(x^*)$ are nonsingular, then there exists $M > 0$ and $k_M > 0$ such that for all $k \geq k_M$,

$$\|(\nabla\Phi_{\varepsilon_k}(x^k))\| \leq M, \quad \|(\nabla\Phi_{\varepsilon_k}(x^k))^{-1}\| \leq M \tag{5.1}$$

Proof. Note that for any $x \in R^n$, the set $\partial_C\Phi(x)$ is compact, see Ref[1]. Hence there exists $V_k \in \partial_C\Phi(x^k)$ such that

$$\text{dist}(\Phi'_{\varepsilon_k}(x^k), \partial_C\Phi(x^k)) = \|\Phi'_{\varepsilon_k}(x^k) - V_k\|$$

which, together with (3.8), implies that for all $k \in K_2$, we have

$$\|\Phi'_{\varepsilon_k}(x^k) - V_k\| \le \nu\beta_k \tag{5.2}$$

By Lemma 4.3, we have $\{\beta_k\} \to 0$. Combined with the nonsingularity of all $V \in \partial_C\Phi(x^*)$ and the upper semicontinuity of $\partial_C\Phi(\cdot)$ at x^*, it follows from (5.2) that there exist $M > 0$ and $k_M > 0$ such that for all $k \in K_2$ with $k \ge k_M$,

$$\|(\Phi'_{\varepsilon_k}(x^k))^{-1}\| \le M \quad \text{and} \quad \|(\nabla\Phi_{\varepsilon_k}(x^k))\| \le M$$

This proof is complete. □

Lemma 5.2 [2]. If $A \in R^{n\times n}$ is nonsingular and $\|A^{-1}\Delta A\| < 1$, then the matrix $A + \Delta A$ is nonsingular and satisfies

$$\|(A + \Delta A)^{-1}\| \le \frac{\|A^{-1}\|}{1 - \|A^{-1}\Delta A\|} \tag{5.3}$$

Lemma 5.3. Let $A_k = \nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k}I$. If the conditions of Lemma 5.1 hold, then we have that for $h_k \ge 2M^2$

$$\|A_k^{-1}\| \le 2M^2, \quad \forall k \in K, \ k \ge k_M \tag{5.4}$$

Proof. By using (5.1) and $h_k \ge 2M^2$, we have

$$\|\frac{1}{h_k}(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T)^{-1}\| \le \frac{1}{2}, \quad \forall k \in K, \ k \ge k_M$$

which, together with (5.1) and (5.3), implies that

$$\|A_k^{-1}\| \le \frac{\|\nabla\Phi_{\varepsilon_k}(x^k)^{-1}\|^2}{1 - \|\frac{1}{h_k}(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T)^{-1}\|} \le 2M^2$$

for all $k \in K$ with $k \ge k_M$. This proof is complete. □

Lemma 5.4. Assume that F is a P_0 function. If Assumptions **A1** and **A2** hold, then there exists a positive inter $\bar{k}$ such that $r_k \ge r$ for all $k \ge \bar{k}$.

Proof. From Lemma 4.3, (3.1), Assumptions A1 and A2, we have

$$\|d_k\| \le \|(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k}I)^{-1}\|\|\nabla\Phi_{\varepsilon_k}(x^k)\Phi_{\varepsilon_k}(x^k)\| \longrightarrow 0 \tag{5.5}$$

Thus it follows from Lemma 3.1(i) and Lemma 4.3 that for sufficiently large k

$$\begin{array}{rcl} Ared_k - rPred_k & = & \psi_{\varepsilon_k}(x^k) - \psi_{\varepsilon_k}(x^k + d_k) - rPred_k \\ & = & -\nabla\psi_{\varepsilon_k}(x^k)^T d_k - \frac{1}{2} d_k^T \nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T d_k \\ & & +o(\|d_k\|^2) - rPred_k \\ & = & (1-r)Pred_k + o(\|d_k\|^2) \\ & \geq & \frac{(1-r)}{2} d_k^T(\nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k} I)d_k + o(\|d_k\|^2) \\ & \geq & \|d_k\|^2(\frac{m(1-r)}{2} + \frac{o(\|d_k\|^2)}{\|d_k\|^2}) \end{array}$$

which implies that there exists a positive integer $\bar{k}$ such that

$$Ared_k \geq rPred_k, \quad \forall k \geq \bar{k}$$

This shows that $r_k \geq r$ for all $k \geq \bar{k}$. This proof is complete. □

Remark 5.1. From Lemma 4.3 and Lemma 5.4, it follows that the set $\hat{K}$ is infinite, where $\hat{K} = \{k \in K | r_k \geq r\}$.

Using the above conclusions, we now analyze the local convergence rate of Algorithm STR.

Theorem 5.5. Suppose that F is a P_0 function. Suppose also that Assumptions **A1** and **A2** hold. If for an accumulation point x^* of the subsequence $\{x^k\}_{k\in\hat{K}}$, all $V \in \partial_C\Phi(x^*)$ are nonsingular, then we have
(i) x^* is a solution of $\Phi(x) = 0$, and thus a solution of NCP(1.1).
(ii) The whole sequence $\{x^k\}$ converges to x^* superlinearly.

proof. By using Lemma 4.3, the conclusion (i) is obvious. In what follows we show the validity of conclusion (ii).

Since $\partial_B\Phi(x^*) \subseteq \partial_C\Phi(x^*)$, Proposition 2.5 in Ref[16] and Lemma 2.1 show that x^* is an isolated solution of $\Phi(x) = 0$ and hence also of NCP(1.1). Furthermore, theorem 4.4 implies that x^* is also an isolated accumulation point of the sequence $\{x^k\}$. Thus there exists a constant $\delta > 0$ such that x^* is the unique accumulation point of the sequence $\{x^k\}$ in neighbourhood $N(x^*, \delta) = \{x \in R^n | \, \|x - x^*\| \leq \delta\}$.

Set

$$K_0 = \{k \in \hat{K} | \; x^k \in N(x^*, \delta)\}$$

In view of Lemma 4.3 and Remark 5.1, it is obvious that K_0 is infinite and $\{x^k\}_{k\in K_0}$ converges to x^*. By Lemma 5.1 and Lemma 5.4, there exist $M > 0$ and $\tilde{k} \in K_0$ such that for all $k \in K_0$ with $k \geq \tilde{k}$,

$$r_k \geq r, \quad \|(\Phi'_{\varepsilon_k}(x^k))^{-1}\| \leq M \quad \text{and } \|\nabla\Phi_{\varepsilon_k}(x^k)\| \leq M,$$

and (5.2) with some $V_k \in \partial_C\Phi(x^k)$ holds. Furthermore, it follows from the rule of step 2 and Lemma 5.4 that

$$\lim_{k\to\infty} h_k = +\infty \tag{5.6}$$

Let $A_k = \nabla\Phi_{\varepsilon_k}(x^k)\nabla\Phi_{\varepsilon_k}(x^k)^T + \frac{1}{h_k}I$. Then it follows from Lemma 5.1, Lemma 5.3, Lemma 2.2, (3.1) and (5.6) that for all $k \in K_0$ with $k \geq \tilde{k}$,

$$\begin{array}{rcl} \|x^{k+1} - x^*\| & = & \|x^k - x^* - A_k^{-1}\nabla\Phi_{\varepsilon_k}(x^k)\Phi_{\varepsilon_k}(x^k)\| \\ & \leq & \|A_k^{-1}\| \cdot \|A_k(x^k - x^*) - \nabla\Phi_{\varepsilon_k}(x^k)\Phi_{\varepsilon_k}(x^k)\| \\ & \leq & \|A_k^{-1}\|\|\nabla\Phi_{\varepsilon_k}(x^k)\|\|(\nabla\Phi_{\varepsilon_k}(x^k)^T - V_k)(x^k - x^*)\| \\ & & +\|A_k^{-1}\|\|\nabla\Phi_{\varepsilon_k}(x^k)\|\|V_k(x^k - x^*) - \Phi(x^k) + \Phi(x^*)\| \\ & & +\|A_k^{-1}\|\|\nabla\Phi_{\varepsilon_k}(x^k)\|\|\Phi(x^k) - \Phi_{\varepsilon_k}(x^k)\| + \frac{\|A_k^{-1}\|\|x^k - x^*\|}{h_k} \\ & \leq & 2M^3(v\beta_k\|x^k - x^*\| + o(\|x^k - x^*\|) + \kappa\sqrt{\varepsilon_k}) \\ & & +o(\|x^k - x^*\|) \end{array} \tag{5.7}$$

By the local Lipschitz continuity of Φ, for all $k \to \infty$ and $K \in K_0$,

$$\beta_k = \|\Phi(x^k)\| = O(\|x^k - x^*\|)$$

and from (3.2),

$$\varepsilon_k = O(\|x^k - x^*\|^4)$$

Thus it follows from (5.7) that

$$\|x^{k+1} - x^*\| = o(\|x^k - x^*\|) \quad \text{as } k \to \infty, k \in K_0 \tag{5.8}$$

From (5.8) and the proof of theorem 3.1 in [11], we obtain

$$\|\Phi(x^{k+1})\| = o(\|\Phi(x^k)\|), \quad \text{as } k \to \infty, k \in K_0 \tag{5.9}$$

So there exists $\hat{k} \geq \tilde{k}$ such that for all $k \in K_0$ with $k \geq \hat{k}$,

$$\|x^{k+1} - x^*\| \leq \delta \quad \text{and} \quad \|\Phi(x^{k+1})\| \leq \eta\|\Phi(x^k)\| = \eta\beta_k$$

which implies that $x^{k+1} \in N(x^*, \delta)$ and $k + 1 \in K$. Since $r_k \geq r$, we deduce $k + 1 \in K_0$.

Repeating the above process, we may prove that $k \in K_0$ for all $k \geq \hat{k}$. This together with (5.8) implies that $\{x^k\}$ converges to x^* superlinearly. □

6. Numerical examples

To illustrate the computational behavior of the proposed algorithm STR, we implement it with the code written in MATLAB 7.1. The testing is performed on a PC computer with HPdx2810SE Pentium(R) Dual-Core CPU E5300 @ 2.60GHZ 2.00GB. Throughout the computational experiments, the stop criteria is $\|\nabla\psi(x^{k+1})\| \leq 10^{-6}$, the parameters used in Algorithm STR are chosen as follows: $\eta = 0.9$, $r = 0.01$, $\mu = 0.5$, $\nu = 0.9$, and $h_0 = 100$.

To validate the algorithm STR from a computational point of view, we compare it with the algorithm denoted by YTR in Ref[6] and Jacobian smoothing method denoted by JSM in Ref[1]. Algorithms YTR and JSM are implemented in the same way. The used termination condition for the algorithm YTR is $\|\nabla\psi(x^{k+1})\| \leq 10^{-6}$ with the parameters $\eta = 0.9$, $\mu = 0.5$, $v = 0.9$, $\Delta_0 = 1$,

$c_1 = 0.75$, $c_2 = 0.01$, $c_3 = 0.5$, $c_4 = 2$, $\Delta_{\min} = 10^{-10}$. The used termination condition for the algorithm JSM is $\|\Phi(x^{k+1})\| \leq 10^{-6}$ with the parameters $\rho = a = \eta = 0.5$, $\gamma = 0.9$, $\delta = 0.25(1-a)$, $\kappa = \sqrt{2n}$.

Table 1. Test results for three different algorithm

Problem	initial point	*STR*	*YTR*	*JSM*
$P1$	$(1,0,1,0)^T$	5/0.088596	5/0.086723	22/1.531659
	$(100,0,0,0)^T$	6/0.093388	55/0.124041	28/1.963543
$P2$	$(1,2,3)^T$	9/0.069986	11/0.088534	22/1.349413
	$(100,100,100)^T$	6/0.092930	22/0.090983	27/1.531170
$P3$	$(1,1,1,1)^T$	5/0.082463	5/0.089761	$**$
	$(100,1,15,4)^T$	7/0.082941	11/0.089949	29/2.077492
$P4$	$(0,0,0,0,0)^T$	129/0.094000	155/0.360000	$*$
	$(1,1,1,1,1)^T$	131/0.203000	144/0.344000	$*$
$P5$	$(1,1,1,1,1)^T$	47/0.109637	11/0.088168	47/0.045222
	$(0,0,0,0,0)^T$	46/0.104076	16/0.091314	$**$
$P6$	$ones(8,1)$	6/0.028983	21/0.088337	22/0.010210
	$ones(16,1)$	6/0.030522	16/0.087152	22/0.020398
$P7$	$(0,0)^T$	5/0.391620	4/0.315380	22/0.852860
	$(10,10)^T$	5/0.403436	8/0.731152	23/1.071349
$P8$	$(0,0,0,0)^T$	6/1.028329	6/1.672262	23/1.608630
	$(1,2,3,4)^T$	7/1.204731	596/75.590994	41/2.553601
$P9$	$zeros(8,1)$	12/0.032583	16/0.089010	30/0.010203
	$zeros(16,1)$	7/0.032541	16/0.089457	40/0.020196
$P10$	$zeros(300,1)$	5/1.042489	11/2.240013	465/56.845228
	$zeros(500,1)$	5/4.001045	11/8.880846	$*$
$P11$	$(0,0,0)^T$	6/0.614638	6/0.602766	22/1.127613
	$(10,10,10)^T$	6/0.607659	10/1.051917	24/1.571039
$P12$	$(0,0,0)^T$	6/0.553468	6/0.554196	22/1.039595
	$(10,10,10)^T$	6/0.698658	9/0.851529	$**$
$P13$	$(0,0,0,0)^T$	7/1.397530	7/1.032007	25/1.838657
	$(10,10,10,10)^T$	8/1.573075	11/2.379361	27/1.970828
$P14$	$(1,1,1)^T$	5/0.519001	7/1.187778	20/1.279688
	$(10,10,10)^T$	5/0.515478	8/1.676702	23/1.459699
$P15$	$(1,1,1)^T$	5/0.509390	$**$	$**$
	$(10,10,10)^T$	6/0.580846	$**$	$**$
$P16$	$zeros(300,1)$	5/1.253284	$*$	26/5.138878
	$zeros(500,1)$	5/5.511264	$*$	26/18.102833
$P17$	$zeros(300,1)$	5/1.022114	7/2.097643	26/4.106432
	$zeros(500,1)$	5/3.894817	7/8.717575	26/17.654814
$P18$	$zeros(20,1)$	8/5.985018	7/6.565590	25/10.201406
	$ones(20,1)$	8/5.942147	6/4.696455	23/8.818761

We choose 18 test problems for our experiments. The first 6 test problems are chosen from Refs[17] and [18], which are given in Appendix. Other 12 test problems are chosen from Ref[18] and Tests 7-18 correspond to Problems 1-12 in Ref[18], respectively. Table 1 lists the numerical results which are give in the form of $k/CPU(s)$, where k and $CPU(s)$ denote the the number of iteration and the CPU time of algorithm, respectively. If the CPU time exceeds 300 seconds, we denote it by the sign $*$. If the problem is unsolvable, we denote it by the sign $**$.

Comparing the numerical results of Algorithm STR with that of Algorithms YTR and JSM, we find that there are quite a number of test problems for which Algorithm STR performs better than Algorithms YTR and JSM, in the terms of iterative numbers and CPU time. Therefore, we could say that Algorithm STR is an efficient method in some sense.

While it would be unwise to draw some firm conclusions from the rather limited numerical results, they indicate some promise for the new approach proposed in this paper. Further improvement is expected for more sophisticated implementation.

References

1. J.Y.Han, N.H.Xiu and H.D.Qi, *Nonlinear Complementarity: Theory and Algorithm*, Shanghai Science and Technology Press, Shanghai, 2006.
2. X.J.Chen, *Smoothing methods for complementarity problems and their applications: A Survey*, Journal of the Operations Research Society of Japan, **43(1)**(2000), 32-47.

3 Z.S.Yu, K.Su and J.Lin, *A smoothing Levenberg-Marquardt method for the extended linear complementarity problem*, Applied Mathematical Modelling, **33(8)**(2009), 3409-3420.

4. S.P.Rui and C.X.Xu, *A smoothing inexact Newton method for nonlinear complementarity problems*, Journal of Computational and Applied Mathematics, **233**(2010), 2332-2338.
5. Y.X,Yuan,W.Y.Sun, *Optimization Theory and Methods*, Science Press, Beijing, 1997.
6. Y.F.Yang and L.Q.Qi, *Smoothing trust region methods for nonlinear complementarity problems with P0-functions*, Annals of Operations Research, **133**(2005), 99-117.
7. Y.Zhou, *A smoothing conic trust region filter method for the nonlinear complementarity problem*, Journal of Computational and Applied Mathematics, **229**(2009), 248-263.
8. J.Long, C.F.Ma and P.Y.Ni, *A new filter method for solving nonlinear complementarity problems*, Applied Mathematics and Computation, **185**(2007), 705-718.
9. J.Nocedal and Y.X.Yuan, *Combing trust region and line search techniques*, In: Y.X.Yuan, ed, *Advances in Nonlinear programming*, Kluwer, 1998, 153-175.
10. E.M.Gertz, *A quasi-Newton trust region method*, Mathematical Programming(Series A) **100** (2004), 447-470.
11. S.J.Qu, M.Goh and X.J.Zhang, *A new hybrid method for nonlinear complementarity problems*, Computational Optimization and Applications, Doi 10.1007/s10589-009-9309-7.
12. A.A.Brown and M.C.Biggs, *Some effective methods for unconstrained optimization based on the solution of system of ordinary differentiable equations*, Journal of Optimization Theory and Application **62**(1989), 211-224.
13. C.Kanzow, *Some non-interior continuation methods for linear complementarity problems*, SIAM Journal on Matrix Analysis and Application **17**(1996), 851-868.
14. F.H.Clarke, *Optimization and Nonsmooth Analysis*, John Wiley and Sons, New York, 1983.
15. Kanzow C and Pieper H, *Jacobian smoothing methods for nonlinear complementarity problem*, SIAM Journal on Optimization **9**(1999), 342-373.
16. L.Q.Qi, *Convergence analysis of some algorithms for solving nonsmooth equations*, Mathematics of Operations Research **18**(1993), 227-244.

17. J.S.Chen, C.H.Ko and S.H.Pan, *A neural network based on the generalized Fischer-Burmeister function for nonlinear complementarity problems*, Information Sciences **180** (2010), 697-711.

18. J.L.Zhang and X.S.Zhang, *A smoothing Levenberg-Marquardt method for NCP*, Applied Mathematics and Computation **178** (2006), 212-228.

Appendix

Test 1. Test function:

$$F(x) = \begin{pmatrix} 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ 2x_1^2 + x_2^2 + x_1 + 3x_3 + 2x_4 - 2 \\ 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 3x_4 - 1 \\ x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3 \end{pmatrix}$$

Test 2. Test function:

$$F(x) = \begin{pmatrix} x_1 - 2 \\ x_2 - x_3 + x_2^3 + 3 \\ x_2 + x_3 + 2x_3^3 - 3 \end{pmatrix}$$

Test 3. Test function:

$$F(x) = \begin{pmatrix} -x_2 + x_3 + x_4 \\ x_1 - \frac{4.5x_3+2.7x_4}{x_2+1} \\ 5 - x_1 - \frac{0.5x_3+0.3x_4}{x_3+1} \\ 3 - x_1 \end{pmatrix}$$

Test 4. Test function:

$$F(x) = \begin{pmatrix} x_1^2 + x_2^2 - x_4 \\ x_2^2 + x_5^2 - x_3x_4 \\ -exp(2x_3) + x_4 \\ exp(x_5 - x_1) - x_4 + x_2^2 \\ 1 - x_1 - x_3 \end{pmatrix}$$

Test 5. Test function: $F(x) = (F_1(x), F_2(x), \cdots, F_5(x))^T$, where

$$F_i(x) = 2(x_i - i + 2)exp(\sum_{j=1}^{5}(x_j - j + 2)^2),\ i = 1, 2, \cdots, 5$$

Note that F is not a P_0-function.

Test 6. Test function: $F(x) = Mx + q$, where $M \in R^{n\times n}$ and $q \in R^n$ are defined as follows
$M_{ii} = 4(i-1) + 1,\ i = 1, 2, \cdots, n.$
$M_{ij} = M_{ii} + 1,\ j \neq i,\ i, j = 1, 2, \cdots, n.$
$q = -(1, 1, \cdots, 1)^T$

Yi-gui Ou received his Ph.D from the University of Science and Technology of China on the subject. operations research and cybernetics. His research interests focus on theory and algorithm on optimization.

Department of Applied Mathematics, Hainan University, Haikou, Hainan 570228, China.
e-mail: ouyigui@tom.com