



Smoothing Newton method for NCP with the identification of degenerate indices[☆]

Haodong Yu^{*}, Dingguo Pu

Department of Mathematics, Tongji University, Shanghai, 200092, PR China

ARTICLE INFO

Article history:

Received 21 October 2009

Received in revised form 25 April 2010

MSC:

90C33

Keywords:

Nonlinear complementarity problems

Degenerate indices

Smoothing methods

Global convergence

Superlinear convergence

ABSTRACT

We present a new smoothing Newton method for nonlinear complementarity problems (NCP(F)) by using an NCP function to reformulate the problem to its equivalent form. Compared with most current smoothing methods, our method contains an estimating technique based on the active-set strategy. This technique focuses on the identification of the degenerate set for a solution x^* of the NCP(F). The proposed method has the global convergence, each accumulation point is a solution of the problem. The introduction of the active-set strategy effectively reduces the scale of the problem. Under some regularity assumption, the degenerate set can be identified correctly near the solution and local superlinear convergence is obtained as well.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

We consider the nonlinear complementarity problem (NCP(F)) with the following form:

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

where $x \in \mathbb{R}^n$ and $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ is a continuously differentiable function. The problem is called the linear complementarity problem if the function F is affine.

The nonlinear complementarity problem has attracted great interest due to its important applications in economics, engineering, mechanics, etc. (see [1] for a review). Among many methods proposed for solving NCP(F), one of the most efficient approaches is to reformulate this problem as an optimization problem or a system of nonsmooth equations by using some NCP functions.

A function $\phi: \mathbb{R}^2 \rightarrow \mathbb{R}$ is called an NCP function if the following property is satisfied:

$$\phi(a, b) = 0 \Leftrightarrow a \geq 0, \quad b \geq 0 \quad \text{and} \quad ab = 0.$$

By rewriting the complementarity problem by the NCP function, problem (1) is equivalent with the nonlinear systems as follows:

$$\Phi(x) = \begin{pmatrix} \phi(x_1, F_1(x)) \\ \vdots \\ \phi(x_n, F_n(x)) \end{pmatrix} = 0. \quad (2)$$

The nature merit function of (2) is $\Psi(x) := \frac{1}{2} \Phi(x)^T \Phi(x) = \frac{1}{2} \|\Phi(x)\|^2$.

[☆] Research supported by National Natural Science Foundation of China (Grant No. 10771162).

^{*} Corresponding author.

E-mail address: nianchuixiao@msn.com (H. Yu).

One of the most powerful NCP functions is Fischer–Burmeister function defined as

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

The F–B function is continuously differentiable for any $(a, b)^T \neq 0$. But it is only semismooth at zero, which forces (2) to be semismooth equations.

Current Newton-type methods for NCP(F) can be divided into two classes. The first class, based on the generalized Jacobian (see [2–6]), is called nonsmooth method. The second class, trying to use a sequence of smooth equations to approximate the nonsmooth systems, is called smoothing method (see [7–13]). Both of these approaches have to deal with the nonsmooth of $\Phi(x)$, which comes from the semismooth of F–B function at zero. Therefore, the identification of the degenerate set $\beta(x^*) = \{j | x_j^* = 0 = F_j(x^*)\}$ of a solution x^* turns to be very important for both theoretical and practical study.

In [14], Facchinei, Fischer and Kanzow proposed a technique for the accurate identification of active constraints for nonlinear programs with inequality constraints. The key of their method is to define an identification function $\rho(x, \lambda)$ which converges to the KKT set at a “slower” rate than the pair (x, λ) itself does. Using their technique, Kanzow and Qi [15] presented an active-set QP-free Newton method for variational inequality problems. Under Robinson’s [16] regularity condition, the approximation set of their method is eventually equal to the set of active constraints. Another way for identifying the degenerate indices, proposed in [17], is to use the proximal point algorithm (PPA) of [18]. Their method does not rely on concrete algorithms for NCP(F) or the local uniqueness of the solution. However, their method has to assume the function F to be monotonous, which is too strong for most practical problems. (See [19–21] for more about active-set methods for complementarity problems.)

On the other hand, none of these methods uses smoothing technique. However, smoothing method, as a widely used technique, can directly use standard Newton step and is more convenient to use in many aspects.

In this paper, we propose a new smoothing Newton method for solving nonlinear complementarity problem with which F is a P_0 function. The method uses an estimate technique, which is based on the active-set strategy presented in [14,15], to identify the degenerate indices during the iteration. At each iteration, an estimate condition is used to construct the estimate set. Then, a trail step d^k is computed by solving a reduced linear system. If the trail step gives a full descent to Ψ , the ‘fast step’ will be accepted and the smoothing parameter ε is updated. Otherwise, a line search is carried out on an adjusted direction \tilde{d}^k . In this case, the smoothing parameter is updated only when some update condition is satisfied. The merits of this method are: each accumulation point is a solution of the NCP(F); under some regularity assumption, the estimate set is eventually equal to the degenerate set of the solution x^* , and the local superlinear convergence of the algorithm is obtained as well; the introduction of the active-set strategy reduces the scale of the problem efficiently.

This paper is organized as follows. In Section 2, some background materials and lemmas are given. In Section 3, the proposed algorithm is stated, and we prove the global convergence of our method in Section 4. The local convergence and the analysis of the identification technique is established in Section 5. Finally, the numerical results of the algorithm are listed.

Some words about the notation: Let mapping $G: \mathbb{R}^n \rightarrow \mathbb{R}^n$ be locally Lipschitzian. Denote $G'(x)$ be the Jacobian of G at a point $x \in \mathbb{R}^n$, and $\nabla G(x)$ be the transposed Jacobian of G . For any $x \in \mathbb{R}^n$ and an index set $\mathcal{J} \subseteq \{1, \dots, n\}$, we denote $x_{\mathcal{J}}$ as the vector with components $x_i, i \in \mathcal{J}$. Denote $\|\cdot\|$ be the Euclidean norm of \mathbb{R}^n .

2. Definitions and lemmas

We use Kanzow’s [22] smoothing function to approximate the F–B function:

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

Smoothing methods try to approximate the solution of NCP(F) by solving a series of systems as follows

$$\Phi_{\varepsilon}(x) := \begin{pmatrix} \phi_{\varepsilon}(x_1, F_1(x)) \\ \vdots \\ \phi_{\varepsilon}(x_n, F_n(x)) \end{pmatrix}. \quad (3)$$

Like the definition of $\Psi(\cdot)$, we define $\Psi_{\varepsilon}(x) = \frac{1}{2} \Phi_{\varepsilon}(x)^T \Phi_{\varepsilon}(x) = \frac{1}{2} \|\Phi_{\varepsilon}(x)\|^2$.

Proposition 2.1 ([23]). *The function Ψ is continuously differentiable with $\nabla \Psi(x) = H^T \Phi(x)$ for an arbitrary element $H \in \partial \Phi(x)$.*

Denote $G = (G_1, \dots, G_n)^T$. Qi [2] defined the C-subdifferential of G at x to estimate the generalized Jacobian $\partial G(x)$:

$$\partial_C G(x)^T := \partial G_1(x) \times \partial G_2(x) \times \dots \times \partial G_n(x).$$

Proposition 2.2 ([8]). Assume that $\{x^k\} \subseteq \mathbb{R}^n$ is a convergent sequence with a limit point $x^* \in \mathbb{R}^n$. Then the following statements hold.

(i) The function Φ is semismooth, which implies that for any $V_k \in \partial_C \Phi(x^k)$,

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

(ii) If F' is Lipschitz continuous, then the function Φ is strongly semismooth, which implies that for any $V_k \in \partial_C \Phi(x^k)$,

$$\|\Phi(x^k) - \Phi(x^*) - V_k(x^k - x^*)\| = O(\|x^k - x^*\|^2).$$

We now restate the concept of the P_0 function and P_0 matrix.

Definition 2.3. A function $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is said to be a P_0 function if for all $x, y \in \mathbb{R}^n$ with $x \neq y$,

$$\max_{i: x_i \neq y_i} (x_i - y_i)(G_i(x) - G_i(y)) \geq 0. \quad (4)$$

Furthermore, if the inequality holds strictly, then the function G is called a P function.

Definition 2.4. A matrix M is said to be a P_0 matrix if for all $x \in \mathbb{R}^n$, $x \neq 0$, there exists a scalar $x_k \neq 0$ such that

$$x_k(Mx)_k \geq 0. \quad (5)$$

If the function G is continuous on a nonempty convex set C , it is well known that G is a P_0 function on C if and only if $\nabla G(x)$ is a P_0 matrix for all $x \in C$. We also have the following result.

Proposition 2.5 ([7]). $\nabla \Phi_\varepsilon(x)$ is nonsingular for all $x \in \mathbb{R}^n$ and $\varepsilon > 0$ if F is a P_0 function.

Lemma 2.6 ([8]). Let $x \in \mathbb{R}^n$ be arbitrary but fixed. Assume that x is not a solution of $\text{NCP}(F)$. Let us define the constants

$$\gamma(x) := \max_{i \notin \beta(x)} \{ \|x_i e_i + F_i(x) \nabla F_i(x)\| \} \geq 0$$

and

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

where $\beta(x) := \{i \mid x_i = F_i(x) = 0\}$ and e_i is the i th column of the identical matrix I . 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

$$\text{dist}(\Phi'_\varepsilon(x), \partial_C \Phi(x)) \leq \delta$$

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

Assume that \tilde{x} is a solution of $\text{NCP}(F)$. We introduce the following index set

$$\alpha = \alpha(\tilde{x}) := \{i \mid \tilde{x}_i > 0\},$$

$$\beta = \beta(\tilde{x}) := \{i \mid \tilde{x}_i = 0 = F_i(\tilde{x})\},$$

$$\chi = \chi(\tilde{x}) := \{i \mid F_i(\tilde{x}) > 0\}.$$

Definition 2.7. Assume that \tilde{x} is a solution of the $\text{NCP}(F)$, then \tilde{x} is said to be a R -regular solution if $F'(\tilde{x})_{\alpha\alpha}$ is nonsingular and the Schur complement of it

$$F'(\tilde{x})_{\beta\beta} - F'(\tilde{x})_{\beta\alpha} F'(\tilde{x})_{\alpha\alpha}^{-1} F'(\tilde{x})_{\alpha\beta}$$

is a P matrix.

The regularity condition immediately implies the next result; see Robinson's [16] for details.

Proposition 2.8. Assume that \tilde{x} is a R -regular solution of the $\text{NCP}(F)$, then there exists $c_1 > 0$ and $b_1 > 0$ such that

$$\|\Phi(x)\| \geq c_1 \|x - \tilde{x}\| \quad (6)$$

for all x with $\|x - \tilde{x}\| \leq b_1$.

3. Description of algorithm

In this section, we present our algorithm for P_0 nonlinear complementarity problems. We shall also show that the algorithm is well defined.

For convenience of expression, we denote $H_k := \Phi'_{\varepsilon_k}(x^k)$ and $g_k := \nabla \Psi_{\varepsilon_k}(x^k)$. Obviously,

$$g_k = \Phi'_{\varepsilon_k}(x^k)^T \Phi_{\varepsilon_k}(x^k) = H_k^T \Phi_{\varepsilon_k}(x^k). \quad (7)$$

We also use the index set $\mathcal{I} = \{1, \dots, n\}$ for the variables x .

Now we give the description of our algorithm in detail.

Algorithm 3.1. Step 0. Initialization:

Choose $x^0 \in \mathbb{R}^n$, $\lambda, \eta, \alpha, \nu, \sigma \in (0, 1)$, $\delta > 0$, $\gamma > 0$, and the toleration $\epsilon \geq 0$. Set $\beta_0 := \|\Phi(x^0)\|$, $\rho_0 := \|\Phi(x^0)\|$, $C_0 := (1 + \alpha)\|\Phi(x^0)\|$, $\kappa := \sqrt{2n}$, $\varepsilon_0 := (\frac{\alpha}{2C_0\kappa}\beta_0^2)^2$ and $k := 0$.

Step 1. If $\|\nabla \Psi(x^k)\| \leq \epsilon$, then terminates.

Step 2. Set $\delta_k = \min\{\delta, \|\Phi(x^k)\|^\nu\}$, and

$$I_k := \{i \in \mathcal{I} \mid \|(x_i^k, F_i(x^k))\| \leq \delta_k\}. \quad (8)$$

Step 3. Let

$$\omega^k = \begin{pmatrix} \omega_{I_k}^k \\ \omega_{\bar{I}_k}^k \end{pmatrix}$$

by defining

$$\omega_{I_k}^k = \min\{\rho_k h_k, g_{I_k}^k\} \quad \text{and} \quad \omega_{\bar{I}_k}^k = g_{\bar{I}_k}^k$$

where h_k denotes the $|I_k|$ -dimensions vector $(1, \dots, 1)^T$ ($|I_k|$ stands for the cardinality of I_k).

Step 4. Let $H_k = (H_{I_k}^k, H_{\bar{I}_k}^k)$. Let $d_{I_k}^k$ be the solution of the reduced system

$$(H_{\bar{I}_k}^k)^T H_{\bar{I}_k}^k d_{\bar{I}_k}^k = -\omega_{\bar{I}_k}^k. \quad (9)$$

Step 5. Let

$$d^k = \begin{pmatrix} -x_{I_k}^k \\ d_{\bar{I}_k}^k \end{pmatrix} \quad \text{and} \quad \tilde{d}^k = \begin{pmatrix} -\omega_{I_k}^k \\ d_{\bar{I}_k}^k \end{pmatrix}.$$

If

$$\Psi(x^k + d^k) \leq \frac{\eta^2}{2} \beta_k^2 \quad (10)$$

then set $x^{k+1} = x^k + d^k$ (we call this ‘fast step’), and

$$\beta_{k+1} = \|\Phi(x^{k+1})\|, \quad \varepsilon_{k+1} = \min \left\{ \left(\frac{\alpha}{2C_0\kappa} \beta_{k+1}^2 \right)^2, \frac{\varepsilon_k}{4}, \bar{\varepsilon}(x^{k+1}, \gamma \beta_{k+1}) \right\}. \quad (11)$$

Then, goto step 7. If (10) is not satisfied, goto step 6.

Step 6. Search t_k as the maximal element in $\{\lambda^s \mid s = 0, 1, 2, \dots\}$ which satisfies that

$$\Psi_{\varepsilon_k}(x^k + t_k \tilde{d}^k) \leq (1 - \sigma t_k^2) \Psi_{\varepsilon_k}(x^k) \quad (12)$$

and set $x^{k+1} = x^k + t_k \tilde{d}^k$. If

$$\|\Phi(x^{k+1})\| \leq \max \left\{ \eta \beta_k, \frac{1}{\alpha} \|\Phi(x^{k+1}) - \Phi_{\varepsilon_k}(x^{k+1})\| \right\} \quad (13)$$

then

$$\beta_{k+1} = \|\Phi(x^{k+1})\|, \quad \varepsilon_{k+1} = \min \left\{ \left(\frac{\alpha}{2C_0\kappa} \beta_{k+1}^2 \right)^2, \frac{\varepsilon_k}{4}, \bar{\varepsilon}(x^{k+1}, \gamma \beta_{k+1}) \right\}. \quad (14)$$

Otherwise, set $\beta_{k+1} := \beta_k$ and $\varepsilon_{k+1} := \varepsilon_k$.

Step 7. Set $k := k + 1$, $\rho_{k+1} := \|\Phi(x^{k+1})\|$, and return to step 1.

Remark. (1) The set I_k is used as an approximation set. The setting $\nu \in (0, 1)$ is crucial in the identification of the degenerate set.

(2) The reduced linear system (9) is motivated by the Jacobian smoothing equation

$$\Phi'_{\varepsilon_k}(x^k)d = -\Phi_{\varepsilon_k}(x^k)$$

and its equivalent reformulation

$$\nabla \Phi_{\varepsilon_k}(x^k) \Phi'_{\varepsilon_k}(x^k)d = -\nabla \Phi_{\varepsilon_k}(x^k) \Phi_{\varepsilon_k}(x^k) = -g_k.$$

(3) In Step 5, we first use the direction d^k as the trial direction. If the merit function $\Psi(x^k)$ has a full descent on the trial direction, we adapt the fast step $x^{k+1} = x^k + d^k$. Especially, we point out that in this case, we have $x_{I_k}^{k+1} = 0$. If the ‘fast step’ is not accepted, we take a line search on \tilde{d}^k for the smoothing function $\Phi_{\varepsilon_k}(x^k)$ to guarantee the global convergence of the algorithm.

(4) The smoothing parameter ε_k is updated under two circumstances. If the ‘fast step’ is accepted, or if the update condition (13) is satisfied, ε_k will be recomputed. The update rule (14) of ε_{k+1} consists of three parts. The first part is important in the proof of the superlinear convergence of the algorithm. The second part enables the sequence ε_k to converges to zeros. The remaining part of (14) is used to control the distance between the smoothing Jacobian H_k and the C-subdifferential; see [8] for reference.

We now turn to the analysis of Algorithm 3.1. Without loss of generality, we assume that the terminate toleration $\epsilon = 0$, and the iteration does not terminate in finite steps. We also denote the next index set

$$K := \{0\} \cup \{k \in \mathbb{N} \mid \|\Phi(x^k)\| \leq \max\{\eta\beta_{k-1}, \alpha^{-1}\|\Phi(x^k) - \Phi_{\varepsilon_{k-1}}(x^k)\|\}\}. \quad (15)$$

To prove Algorithm 3.1 is well defined, we first cite the following lemma.

Lemma 3.1 ([8]). *The following two statements hold:*

(a) *We have*

$$\|\Phi(x^k) - \Phi_{\varepsilon_k}(x^k)\| \leq \alpha \|\Phi(x^k)\| \quad (16)$$

for all $k \geq 0$.

(b) *We have*

$$\text{dist}_F(\Phi'_{\varepsilon_k}(x^k), \partial_C \Phi(x^k)) \leq \gamma \|\Phi(x^k)\| \quad (17)$$

for all $k \in K$ with $k \geq 1$.

Recall that $\alpha \in (0, 1)$, the first assertion of this lemma implies the next result.

Lemma 3.2. *For all $k \geq 0$, we have $\Phi(x^k) = 0$ if and only if $\Phi_{\varepsilon_k}(x^k) = 0$.*

Lemma 3.3 ([7]). *$\nabla \Phi_{\varepsilon}(x)$ is nonsingular for all $x \in \mathbb{R}^n$ and $\varepsilon > 0$ if F is a P_0 function.*

Lemma 3.4. *For all $k \geq 0$, we have*

$$\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k \leq 0.$$

And the equality holds only when x^k solves (1).

Proof. By (9) and the definition of \tilde{d}^k , we get

$$\begin{aligned} \nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k &= -(g_{I_k}^k)^T \omega_{I_k}^k + (g_{\bar{I}_k}^k)^T d_{\bar{I}_k}^k \\ &= -(g_{I_k}^k)^T \omega_{I_k}^k - (d_{\bar{I}_k}^k)^T (H_{\bar{I}_k}^k)^T H_{\bar{I}_k}^k d_{\bar{I}_k}^k. \end{aligned} \quad (18)$$

By the expression of $\omega_{I_k}^k$ and the nonnegativeness of ρ_k , it is not difficult to get that

$$-(g_{I_k}^k)^T \omega_{I_k}^k \leq 0.$$

Since the matrix $(H_{\bar{I}_k}^k)^T H_{\bar{I}_k}^k$ is positive semidefinite, the second part of (18) is also nonpositive. Hence, we have shown $\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k \leq 0$.

To prove the second statement, we first assume by contradiction that there exists some x^k such that $\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k = 0$ and $\Psi(x^k) \neq 0$. By Lemmas 3.2 and 3.3, we have $\Phi_{\varepsilon_k}(x^k) \neq 0$ and that the matrix $H_k = \Phi'_{\varepsilon_k}(x^k)$ is nonsingular. Hence, the matrix $(H_{\bar{I}_k}^k)^T H_{\bar{I}_k}^k$ is positive definite.

Therefore, from (18) and the assumption that $\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k = 0$, we have

$$g_{I_k}^k = 0 \quad \text{and} \quad d_{I_k}^k = 0.$$

From (9) in Step 4, we actually obtain that $g_k = 0$. Since $g_k = \nabla \Psi_{\varepsilon_k}(x^k)$ and the matrix H_k is nonsingular, we deduce that $\Phi_{\varepsilon_k}(x^k) = 0$, which contradicts with the previous conclusion that $\Phi_{\varepsilon_k}(x^k) \neq 0$. This completes the proof. \square

Now we can show that Algorithm 3.1 is well defined.

Proposition 3.5. *Algorithm 3.1 is well defined.*

Proof. Since $\varepsilon_k > 0$, we obtain by Lemma 3.3 that the matrices H_k are always nonsingular, which immediately implies that $(H_k^k)^T H_k^k$ are also nonsingular. Therefore, the reduced linear systems (9) always have one unique solution.

We now only have to prove that the line search in Step 6 of the algorithm is well defined. The proof is by contradiction. Assume that for all $s \geq 0$, $s \in \mathbb{N}$, the line search condition (12) is not satisfied. In other words, for any $s \geq 0$,

$$\Psi_{\varepsilon_k}(x^k + \lambda^s \tilde{d}^k) > (1 - \sigma \lambda^{2s}) \Psi_{\varepsilon_k}(x^k).$$

Therefore, for all $s \geq 0$

$$\frac{\Psi_{\varepsilon_k}(x^k + \lambda^s \tilde{d}^k) - \Psi_{\varepsilon_k}(x^k)}{\lambda^s} > -\sigma \lambda^s \Psi_{\varepsilon_k}(x^k).$$

Let $s \rightarrow \infty$, we get that

$$\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k \geq 0.$$

However, as we have assumed, the iteration does not terminate in finite steps. Therefore, Lemma 3.4 implies that $\nabla \Psi_{\varepsilon_k}(x^k) \tilde{d}^k < 0$. This contradiction indicates that the assertion of the proposition is correct. \square

4. Global convergence

In this section, we analyze the global convergence of the proposed algorithm. We shall show that any accumulation point of the iterative sequence is a solution of the NCP(F). We begin with the following results.

Proposition 4.1 ([8]). *The sequence $\{x^k\}$ generated by Algorithm 3.1 remains in the level set*

$$L_0 := \{x \in \mathbb{R}^n \mid \Psi(x) \leq (1 + \alpha)^2 \Psi(x^0)\}. \quad (19)$$

The proof of this proposition, which can be found in [8], also implies the next lemma.

Lemma 4.2 ([8]). *If $K = \{k_0, k_1, k_2, \dots\}$ with $k_0 = 0$, then for all $k \in \mathbb{N}$:*

$$\|\Phi(x^k)\| \leq r^j (1 + \alpha) \|\Phi(x^0)\|$$

where $r = \max\{\frac{1}{2}, \eta\}$, and the index $j \in \mathbb{N}$ denotes the largest element $k_j \in K$ such that $k_j \leq k$.

Assumption 4.3. The level set L_0 defined in Proposition 4.1 is bounded.

The assumption used here is quite trivial, because L_0 is compact if F is an R_0 function (see [24]). With this assumption, we can establish the following lemma.

Lemma 4.4. *If Assumption 4.3 holds, then the index set K is infinite.*

Proof. Assume that K is finite and \tilde{k} is the largest integer in K . Then, for all $k > \tilde{k}$, the ‘fast step’ is not accepted and Step 6 is always carried out, i.e. $x^{k+1} = x^k + t_k \tilde{d}^k$. Moreover, we have

$$\varepsilon_k = \varepsilon_{\tilde{k}}, \quad \beta_k = \beta_{\tilde{k}} = \|\Phi(x^{\tilde{k}})\| \quad (20)$$

$$\|\Phi(x^k)\| > \eta \|\Phi(x^{\tilde{k}})\| > 0 \quad (21)$$

for all $k > \tilde{k}$. Thus, by Lemma 3.2, there exists a positive lower bound for $\{\Psi_{\varepsilon_k}(x^k)\}_{k > \tilde{k}}$. On the other hand, by (12), the sequence $\{\Psi_{\varepsilon_k}(x^k)\}$ must be monotone decreasing when $k > \tilde{k}$. Hence,

$$\lim_{k \rightarrow \infty} \Psi_{\varepsilon_k}(x^k) = m > 0$$

for some constant m . Together with the line search condition (12), we obtain $\lim_{k \rightarrow \infty} t_k = 0$.

Since L_0 is bounded, the iterative sequence must have some accumulation points. Let $\{x^k\}_{k \in I \subseteq K}$ be a subsequence which converges to an accumulation point x^* . Furthermore, we can choose a subsequence $\{x^k\}_{k \in K_1 \subseteq I}$, on which the estimated set $I_k = I_*$ for a subset $I_* \subseteq I$. Therefore, $\{g_{I_k}^k\}_{k \in K_1} \rightarrow [\nabla \Psi_{\varepsilon_k}(x^*)]_{I_*} = g_{I_*}$ (i.e. $\{\omega_{I_k}^k\}_{k \in K_1} \rightarrow \omega_{I_*}^*$), and $\{g_{I_k}^k\}_{k \in K_1} \rightarrow [\nabla \Psi_{\varepsilon_k}(x^*)]_{\tilde{I}_*} = g_{\tilde{I}_*}$.

On the other hand, since $\varepsilon_k > 0$, we can choose a subset $K_2 \subseteq K_1$ such that $\{H_k\}_{K_2} \rightarrow H^*$ for a nonsingular matrix H^* . By taking $k \rightarrow \infty$ on K_2 , we get the limit form of the reduced linear systems (9):

$$(H_{\cdot, I_*}^*)^T H_{\cdot, I_*}^* d_{I_*} = -g_{I_*} \quad (22)$$

which implies that $d_{I_k}^k \rightarrow d_{I_*}$. Recall that $\omega_{I_k}^k \rightarrow \omega_{I_*}^*$, hence we get that $\{\tilde{d}\}_{K_2} \rightarrow \tilde{d}^*$.

We have shown that $\lim_{k \rightarrow \infty} t_k = 0$. Then, in the line search of Step 6 of Algorithm 3.1, we have $\{s_k\} \rightarrow \infty$, and for $k \in K_2$ large enough

$$\frac{\Psi_{\varepsilon_k}(x^k + \lambda^{s_k-1} \tilde{d}^k) - \Psi_{\varepsilon_k}(x^k)}{\lambda^{s_k-1}} > -\sigma \lambda^{s_k-1} \Psi_{\varepsilon_k}(x^k).$$

Let $\{k\}_{K_2} \rightarrow \infty$, we get

$$\nabla \Psi_{\varepsilon_k}^T(x^*) \tilde{d}^* \geq 0.$$

Therefore, x^* must be a solution of (1), which contradicts with (21). \square

Proposition 4.5. Every accumulation point of the iterative sequence is a solution of the NCP(F).

Proof. In Lemma 4.4, we have shown that K is an infinite set. Let x^* be an accumulation point of $\{x^k\}_K$ with a subsequence $\{x^k\}_{M \subseteq K}$ converging to it. We obtain from Lemma 4.2 that

$$\|\Phi(x^*)\| = \lim_{k \in M} \|\Phi(x^k)\| \leq \lim_{j \rightarrow \infty} r^j (1 + \alpha) \|\Phi(x^0)\| = 0$$

where $\{j\}$ is the exponent sequence defined as in Lemma 4.2. Therefore, any accumulation point is a solution of (1). \square

5. Local convergence

In this section, we analyze the local convergence of Algorithm 3.1 under Robinson's regularity condition. Firstly, we establish the following result.

Lemma 5.1. Let $\{x^k\}$ be a sequence generated by Algorithm 3.1, and x^* be an accumulation point of the iterative sequence. Assume that x^* is an R-regular solution of NCP(F). Then, there exists some scalar c_2 such that

$$\left\| \left((H_{\cdot, I_k}^k)^T H_{\cdot, I_k}^k \right)^{-1} \right\| \leq c_2.$$

if $k \in \mathbb{N}$ large enough and x^k is sufficiently close to x^* .

Proof. In the proof of Lemma 3.4, we have shown that the matrices $(H_{\cdot, I_k}^k)^T H_{\cdot, I_k}^k$ are always nonsingular. By Lemma 3.2 of [8], for any $\varepsilon > 0$ and $x \in \mathbb{R}^n$, we have

$$\left[\text{dist}_F(\nabla \Phi_\varepsilon(x), \partial_C \Phi(x)^T) \right]^2 = \sum_{i=1}^n \left[\text{dist}_2(\nabla \Phi_{\varepsilon, i}(x), \partial \Phi_i(x)) \right]^2. \quad (23)$$

Thus, by the proof of Proposition 3.4 in [8], together with the upper semicontinuity of the generalized Jacobian and that $\{\varepsilon_k\} \rightarrow 0$, we get that the sequence $\{H_k\}$ is bounded. Moreover, each limit point of $\{H_k\}$ belongs to $\partial_C \Phi(x^*)$.

By contradiction, we assume the statement is not true. Then there exists some subsequence $\{x^k\}_L$ which converges to x^* and

$$\left\| \left((H_{\cdot, I_k}^k)^T H_{\cdot, I_k}^k \right)^{-1} \right\| \rightarrow \infty \quad (k \in L). \quad (24)$$

Subsequencing if necessary, we suppose that $I_k = I_*$ for all $k \in L$ and some fixed set $I_* \subseteq \mathcal{I}$. We can also assume that $\{H_k\}_L \rightarrow H_*$, where the matrix $H_* \in \partial_C \Phi(x^*)$. The R-regularity of x^* implies that H_* must be nonsingular. Denote $H_* = (H_{\cdot, I_*}^*, H_{\cdot, I_*}^*)$. It follows that the matrix $(H_{\cdot, I_*}^*)^T H_{\cdot, I_*}^*$ is nonsingular. However, since $\{H_k\}_L \rightarrow H_*$, we have $\{H_{\cdot, I_k}^k\}_L \rightarrow H_{\cdot, I_*}^*$. It follows from (24) that $(H_{\cdot, I_*}^*)^T H_{\cdot, I_*}^*$ is a singular matrix. This is a contradiction. Hence the statement is proved. \square

Lemma 5.2. Assume that x^* is an accumulation point and is an R-regular solution of NCP(F), then for all $x^k \in \mathbb{R}^n$ sufficiently close to x^* and $k \in \mathbb{N}$ large enough, there exists some $c_3 > 0$ such that

$$\|\tilde{d}^k\| \leq c_3 \|\Phi_{\varepsilon_k}(x^k)\|$$

where \tilde{d}^k is computed in Step 6 of Algorithm 3.1.

This lemma is slightly different from Lemma 6 in [15]. Notice that the matrices $\{H_k\}$ are bounded and always nonsingular, then the proof of this lemma is almost the same with that in [15].

In Section 2, we introduced the index sets α , β , χ . Here, we focus on the degenerate indices set $\beta = \{i \mid x_i^* = 0 = F_i(x^*)\}$ for some solution x^* . We first have the following result.

Lemma 5.3. *Let x^* be a solution of $NCP(F)$, then $I_k \subseteq \beta$ for all $x^k \in \mathbb{R}^n$ sufficiently close to x^* .*

Proof. Denote

$$\zeta := \min\{\|(x_i^*, F_i(x^*))\| \mid i \notin \beta\} > 0.$$

Since the function $F(\cdot)$ is continuously differentiable and $\Phi(x^*) = 0$, there is a neighbor $N(x^*, b_2)$ and a constant $L > 0$ such that for any $\|x^k - x^*\| \leq b_2$, we have

$$\|F(x^k) - F(x^*)\| \leq L\|x^k - x^*\|$$

and

$$\delta_k \leq \|\Phi(x^k)\|^v \leq \frac{\zeta}{6}.$$

Suppose $i \in I_k$, for any x^k such that

$$\|x^k - x^*\| \leq \min\left\{\frac{\zeta}{6L}, b_2, \frac{\zeta}{6}\right\}$$

we have

$$|x_i^k - x_i^*| \leq \|x^k - x^*\| \leq \frac{\zeta}{6}, \quad |x_i^k| \leq \delta_k \leq \frac{\zeta}{6}$$

and

$$|F_i(x^k)| \leq \delta_k \leq \frac{\zeta}{6}.$$

Therefore, we obtain

$$|x_i^*| \leq |x_i^* - x_i^k| + |x_i^k| \leq \frac{\zeta}{6} + \frac{\zeta}{6} \leq \frac{\zeta}{3}$$

and

$$|F_i(x^*)| \leq |F_i(x^*) - F_i(x^k)| + |F_i(x^k)| \leq L \cdot \frac{\zeta}{6L} + \frac{\zeta}{6} \leq \frac{\zeta}{3}.$$

This implies that

$$\|(x_i^*, F_i(x^*))\| \leq |x_i^*| + |F_i(x^*)| \leq \frac{2\zeta}{3} < \zeta.$$

Hence, $i \in \beta$ for all $x^k \in \mathbb{R}^n$ sufficiently close to x^* . The proof is completed. \square

Lemma 5.4 ([15]). *Assume that x^* is an R-regular solution of $NCP(F)$ and is an accumulation point of the iterative sequence $\{x^k\}$ generated by Algorithm 3.1, then the whole sequence $\{x^k\}$ converges to x^* .*

We now show that under the R-regular condition, the approximation set I_k is equal to the degenerate index set in a small neighbor of x^* .

Lemma 5.5. *Suppose that x^* is an R-regular solution of $NCP(F)$ with the entire sequence $\{x^k\}$ converges to x^* , then $I_k = \beta$ for all k large enough.*

Proof. We have shown in Lemma 5.3 that $I_k \subseteq \beta$ when x^k is close to x^* . Hence, we only have to prove that $\beta \subseteq I_k$ for k sufficiently large.

Assume that $i \in \beta$, then $x_i^* = F_i(x^*) = 0$. It follows from Proposition 2.8 that for all x^k with $\|x^k - x^*\| \leq b_1$, we have

$$\|x^k - x^*\| \leq \frac{1}{c_1} \|\Phi(x^k)\|$$

where b_1 is defined as in Proposition 2.8. Therefore,

$$|x_i^k| = |x_i^k - x_i^*| \leq \|x^k - x^*\| \leq \frac{1}{c_1} \|\Phi(x^k)\|. \quad (25)$$

On the other hand, the continuous differentiability of $F(\cdot)$ implies that there is some constant $b_3 > 0$ such that for all x^k satisfying $\|x^k - x^*\| \leq b_3$:

$$\|F(x^k) - F(x^*)\| \leq L_1 \|x^k - x^*\|$$

for some $L_1 > 0$. Hence,

$$|F_i(x^k)| = |F_i(x^k) - F_i(x^*)| \leq \|F(x^k) - F(x^*)\| \leq L_1 \|x^k - x^*\| \leq \frac{L_1}{c_1} \|\Phi(x^k)\|. \quad (26)$$

It follows from (25) and (26) that for k sufficiently large, $\|x_i^k, F_i(x^k)\| \leq \frac{1+L_1}{c_1} \|\Phi(x^k)\|$. Since $\{x^k\} \rightarrow x^*$, $\{\|\Phi(x^k)\|\} \rightarrow 0$, and $v \in (0, 1)$, then for k large enough

$$\frac{1+L_1}{c_1} \|\Phi(x^k)\| \leq \|\Phi(x^k)\|^v = \delta_k.$$

Hence, we obtain $i \in I_k$, which indicates that $\beta \subseteq I_k$ for all k sufficiently large. The proof is completed. \square

Intuitively, since the degenerate indices will be eventually identified, this means that x_{β}^k can be set to be always equal to zero when x^k is sufficiently close to x^* . Based on this observation and notice that K is infinite, it is reasonable to make the following assumption.

Assumption 5.6. For $k \in K$ sufficiently large, we have $\|x_{\beta}^k\| = o(\|x_{\beta}^k - x_{\beta}^*\|)$.

Proposition 5.7 ([7]). Let $G : \mathbb{R}^n \rightarrow \mathbb{R}^n$ be locally Lipschitzian and $x^* \in \mathbb{R}^n$ with $G(x^*) = 0$ such that all elements in $\partial G(x^*)$ are nonsingular, and assume that there are two sequence $\{x^k\} \subset \mathbb{R}^n$ and $\{d^k\} \subset \mathbb{R}^n$ with

$$\lim_{k \rightarrow \infty} x^k = x^* \quad \text{and} \quad \|x^k + d^k - x^*\| = o(\|x^k - x^*\|).$$

Then

$$\|G(x^k + d^k)\| = o(\|G(x^k)\|).$$

Theorem 5.8. Suppose that x^* is an accumulation point of the sequence $\{x^k\}$ generated by Algorithm 3.1. If x^* is an R -regular solution of NCP(F), then

- (i) The whole sequence $\{x^k\}$ converges to x^* .
- (ii) For all k large enough, the 'fast step' of Algorithm 3.1 is always accepted, i.e., x^{k+1} is always computed by $x^{k+1} = x^k + d^k$.
- (iii) The sequence $\{x^k\}$ converges to x^* Q -superlinearly.

Proof. The first statement follows immediately from Lemma 5.4. To establish the second statement, we first want to show that

$$\|x^k + d^k - x^*\| = o(\|x^k - x^*\|) \quad \text{for } k \in K, k \rightarrow \infty \quad (27)$$

where d^k is computed in Step 5 of Algorithm 3.1. As is shown in Lemma 5.5, $I_k = \beta$ for all k large enough. Without loss of generality, we assume in the following proof that the equality $I_k = \beta$ always holds.

Therefore, if $i \in \beta$, then $d_i^k = -x_i^k$ and $x_i^* = 0$. Hence,

$$|x_i^k + d_i^k - x_i^*| = |x_i^k - x_i^k| = 0 = o(\|x^k - x^*\|). \quad (28)$$

On the other hand, for $i \in \bar{\beta}$, it follows from the system (9) that

$$\begin{aligned} (H_{i_k}^k)^T H_{i_k}^k (x_{i_k}^k + d_{i_k}^k - x_{i_k}^*) &= -g_{i_k}^k + (H_{i_k}^k)^T H_{i_k}^k (x_{i_k}^k - x_{i_k}^*) \\ &= -[H_k^T \Phi_{\varepsilon_k}(x^k)]_{\bar{\beta}} + (H_{\bar{\beta}}^k)^T H_{\bar{\beta}}^k (x_{\bar{\beta}}^k - x_{\bar{\beta}}^*) \\ &= -(H_{\bar{\beta}}^k)^T [\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - H_{\bar{\beta}}^k (x_{\bar{\beta}}^k - x_{\bar{\beta}}^*)] \\ &= -(H_{\bar{\beta}}^k)^T [\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - \Phi'_{\varepsilon_k}(x^k)(x^k - x^*)] - (H_{\bar{\beta}}^k)^T H_{\bar{\beta}}^k (x_{\bar{\beta}}^k - x_{\bar{\beta}}^*) \\ &= -(H_{\bar{\beta}}^k)^T [\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - \Phi'_{\varepsilon_k}(x^k)(x^k - x^*)] + o(\|x^k - x^*\|). \end{aligned} \quad (29)$$

The last equality is by Assumption 5.6. By Lemma 3.1(b), there exists some $V_k \in \partial_C \Phi(x^k)$ such that

$$\text{dist}_F(\Phi'_{\varepsilon_k}(x^k), \partial_C \Phi(x^k)) = \|\Phi'_{\varepsilon_k}(x^k) - V_k\| \leq \gamma \beta_k. \quad (30)$$

Therefore, from Proposition 2.2, together with (30) and the update rules of ε_k , we obtain

$$\begin{aligned} & \|\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - \Phi'_{\varepsilon_k}(x^k)(x^k - x^*)\| \\ & \leq \|\Phi_{\varepsilon_k}(x^k) - \Phi(x^k)\| + \|\Phi(x^k) - \Phi(x^*) - V_k(x^k - x^*)\| + \|(\Phi'_{\varepsilon_k}(x^k) - V_k)(x^k - x^*)\| \\ & \leq \kappa\sqrt{\varepsilon_k} + o(\|x^k - x^*\|) + \gamma\beta_k\|x^k - x^*\| \\ & \leq \frac{\alpha}{2C_0}\beta_k^2 + o(\|x^k - x^*\|) + \gamma\beta_k\|x^k - x^*\|. \end{aligned}$$

Notice that

$$\beta_k = \|\Phi(x^k)\| = O(\|x^k - x^*\|), \quad k \in K \text{ and } k \rightarrow \infty.$$

Hence,

$$\|\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - \Phi'_{\varepsilon_k}(x^k)(x^k - x^*)\| = o(\|x^k - x^*\|).$$

This, together with Lemma 5.1 and (29), implies that

$$\begin{aligned} \|x_{I_k}^k + d_{I_k}^k - x_{I_k}^*\| & \leq c_2 \|H_{\beta}^k\| \cdot [\|\Phi_{\varepsilon_k}(x^k) - \Phi(x^*) - \Phi'_{\varepsilon_k}(x^k)(x^k - x^*)\| + o(\|x^k - x^*\|)] \\ & = o(\|x^k - x^*\|). \end{aligned}$$

Hence, we have proven (27). Thus, it follows from Proposition 5.7 that for all $k \in K$

$$\|\Phi(x^k + d^k)\| = o(\|\Phi(x^k)\|). \quad (31)$$

This implies that for $k \in K$ large enough, $k+1 \in K$ and $x^{k+1} = x^k + d^k$. Repeating this process, we obtain that if k large enough, then $k \in K$ and the condition (10) is always satisfied. Therefore, the second statement is established.

Since the iterative sequence is finally always computed by the ‘fast step’, then $x_{\beta}^k = 0$ for k sufficiently large, and the Q-superlinear convergence of Algorithm 3.1 follows immediately from (27). \square

6. Numerical results

In this section, we illustrate some computational results of some test problems to show the efficiency of the proposed method. The program code of Algorithm 3.1 was implemented in MATLAB 7.5 environment and run on a Intel Pentium 4. We set the toleration be $\epsilon = 1.0e - 6$ and set the parameters be $\eta = 0.8$, $\alpha = 0.7$, $\lambda = 0.5$, $\nu = 0.6$, $\sigma = 0.15$, $\gamma = 10$, $\delta = 1$.

Problem 1. Let $F(x) = (f_1(x), f_2(x), f_3(x), f_4(x))^T$, where

$$\begin{aligned} f_1(x) &= 3x_1^2 + 2x_1x_2 + 2x_2^2 + x_3 + 3x_4 - 6 \\ f_2(x) &= 2x_1^2 + x_1 + x_2^2 + 10x_3 + 2x_4 - 2 \\ f_3(x) &= 3x_1^2 + x_1x_2 + 2x_2^2 + 2x_3 + 9x_4 - 9 \\ f_4(x) &= x_1^2 + 3x_2^2 + 2x_3 + 3x_4 - 3. \end{aligned}$$

This example is Kojima-Shindo problem which comes from [25]. It has two solutions: a degenerate solution $x^1 = (\frac{\sqrt{6}}{2}, 0, 0, 0.5)^T$ and a non-degenerate solution $x^2 = (1, 0, 3, 0)^T$.

Problem 2. Let $F(x) = (f_1(x), f_2(x), f_3(x))^T$, where

$$\begin{aligned} f_1(x) &= x_1 - 2 \\ f_2(x) &= x_2 - x_1 - x_3 + x_2^3 + 3 \\ f_3(x) &= x_2 + x_3 + 2x_3^3 - 3. \end{aligned}$$

This problem has a degenerate solution $(2, 0, 1)^T$.

Problem 3. Let $F(x) = (f_1(x), f_2(x), f_3(x), f_4(x))^T$, where

$$\begin{aligned} f_1(x) &= -x_2 + x_3 + x_4 \\ f_2(x) &= x_1 - (4.5x_3 + 2.7x_4)(x_2 + 1) \\ f_3(x) &= 1 - x_1 - (0.5x_2 + 0.3x_4)(x_3 + 1) \\ f_4(x) &= 3 - x_1. \end{aligned}$$

This example is from Mathiesen's [26], here we give a slightly different version. It has a family of solutions $(\varpi, 0, 0, 0)^T$, where $\varpi \in [0, 3]$. If $\varpi = 0, 1, 3$, the solution is degenerate.

Table 1
Numerical results for Algorithm 3.1.

Problem	Start point	Iter	NF	$\Psi(x)$	$\ \nabla\Psi(x)\ $	Fast	$I_k = \beta$	Solution
Pro 1	(4, 3, 2, 1)	9	15	3.5e–23	5.9e–12	4	5	Degenerate
Pro 1	(1, 3, 4, 1)	8	12	1.6e–16	1.3e–08	5	1	Non-degenerate
Pro 1	(–2, –1, 0, 2)	7	10	7.3e–16	2.7e–08	4	6	Degenerate
Pro 1	(5, 0, 0, 4)	7	10	9.6e–26	3.1e–13	4	6	Degenerate
Pro 2	(1, 1, 1)	5	5	4.6e–19	6.8e–10	5	2	Degenerate
Pro 2	(4, 3, 2)	11	16	2.8e–26	1.7e–13	5	6	Degenerate
Pro 2	(–100, 100, 100)	8	8	9.3e–26	3.1e–13	7	5	Degenerate
Pro 2	(–2, –2, –2)	39	46	9.3e–13	9.6e–07	5	8	Degenerate
Pro 3	(3, 3, 3, 3)	7	7	1.5e–15	3.9e–08	7	5	$\varpi = 0$
Pro 3	(–1, –2, –3, –4)	6	7	1.2e–14	1.1e–07	5	5	$\varpi = 0$
Pro 3	(2, 1, 2, 0)	6	6	8.9e–12	3.0e–07	6	2	$\varpi = 0.189$
Pro 3	(4, 0, 1, 1)	55	56	1.6e–12	1.3e–07	3	4	$\varpi = 1$
Pro 4	(6, 6, 6)	8	14	8.1e–28	2.8e–14	6	6	Degenerate
Pro 4	(1, 32, 42)	9	15	8.1e–28	2.8e–14	5	6	Degenerate
Pro 4	(–5, –8, –9)	19	26	1.7e–15	4.2e–08	16	4	Degenerate
Pro 4	(–200, 200, 200)	18	23	8.1e–28	2.8e–14	6	10	Degenerate

Problem 4. Let $F(x) = (f_1(x), f_2(x), f_3(x))^T$, where

$$f_1(x) = x_1^2 + \sin(x_1)$$

$$f_2(x) = x_2^3 + x_1 x_3$$

$$f_3(x) = x_3^2 - 200 + x_1 x_2.$$

This problem is from [27] and has a degenerate solution $(0, 0, \sqrt{200})^T$.

The computational results of these test problems are listed in the following table, in which **Iter** denotes the number of iterations, **NF** means the number of evaluations of the function F , **Fast** denotes the number of ‘fast step’ taken during the iteration, and $I_k = \beta$ means the iteration from which the degenerate set is correctly identified.

The last column ‘Solution’ lists the main characteristics of the computed results of each start point. Both Problems 2 and 4 have one unique solution which is degenerate, hence their ‘Solution’s are always ‘degenerate’. Problem 1 has both degenerate and non-degenerate solutions. As to Problem 3, since there exists a family of solutions, we list the parameter ϖ of each computed solution.

As is shown in the column **Iter** and **NF** of Table 1, for most initial points, the solutions are found very quickly and the number of function evaluation is also very small. On the other hand, from the column **Fast**, it can be seen that, in general, the number of the iteration in which ‘fast step’ are accepted occupied at least half of all the iterations. This means the adaptation of the active-set strategy effectively reduces the scale of the problem, since the algorithm only have to solve the reduced systems in the ‘fast step’ case. It can also be seen from the column $I_k = \beta$ that the degenerate indices are usually identified at least several steps before termination. Furthermore, the value of **NF** and **Iter** are usually close to each other. This illustrates that, in the case when ‘fast step’ is not accepted, the computation amount of the ‘adjustment step’ is very small. This illustrates that the ‘fast step’ plays a major role during the whole iteration, and it also shows that the scale of the problem is reduced effectively.

For comparison, we also implemented the numerical experiments on the underlying smoothing Newton method of Algorithm 3.1. That is, at each iteration, d^k is computed by solving the following equation:

$$H_k d = -\Phi_{\varepsilon_k}(x^k) \quad (32)$$

and the line search step and update rule of ε_{k+1} is the same with Step 6 of Algorithm 3.1. We denote this method as Algorithm GSN (GSN denotes: ‘general smoothing Newton’ method). We tested all the test problems with all the initial points under the same environment. The results is listed in Table 2. Since the active-set strategy is not introduced here, the column **Fast** and $I_k = \beta$ do not appear in this table.

We can see from Table 2 that there exists a failure on Problem 4 for Algorithm GSN. Except for this example, Algorithm GSN generally needs more iterations and function evaluations. Algorithm GSN seems to be more easier to meet numerical difficulties, especially when the iterative sequence converges to a degenerate solution. The instances of this characteristic can be found in the results of Problems 1 and 3. Similar with Algorithm 3.1, Algorithm GSN can obtain both degenerate and non-degenerate solutions. However, from the results of Problem 1, Algorithm 3.1 seems to be more suitable when we want to find a non-degenerate solution.

7. Conclusion

In this paper, we present a smoothing Newton method which adopts the active-set strategy to identify the degenerate indices. The introduction of the identification technique also enables the algorithm only have to solve reduced Newton

Table 2
Numerical results for Algorithm GSN.

Problem	Start point	Iter	NF	$\Psi(x)$	$\ \nabla\Psi(x)\ $	Solution
Pro 1	(4, 3, 2, 1)	42	47	2.8e–13	5.3e–07	Non-degenerate
Pro 1	(1, 3, 4, 1)	9	15	5.0e–17	7.1e–09	Non-degenerate
Pro 1	(–2, –1, 0, 2)	127	717	3.9e–24	2.0e–12	Degenerate
Pro 1	(5, 0, 0, 4)	10	14	5.0e–18	2.3e–09	Degenerate
Pro 2	(1, 1, 1)	5	5	1.1e–27	3.3e–14	Degenerate
Pro 2	(4, 3, 2)	6	8	6.1e–24	2.5e–12	Degenerate
Pro 2	(–100, 100, 100)	8	9	2.7e–25	5.2e–13	Degenerate
Pro 2	(–2, –2, –2)	8	9	6.2e–17	7.9e–09	Degenerate
Pro 3	(3, 3, 3, 3)	10	13	1.8e–14	1.3e–07	$\varpi = 0$
Pro 3	(–1, –2, –3, –4)	12	24	3.2e–15	5.6e–08	$\varpi = 0$
Pro 3	(2, 1, 2, 0)	80	251	2.8e–11	5.3e–06	$\varpi = 0$
Pro 3	(4, 0, 1, 1)	8	8	5.5e–14	2.3e–07	$\varpi = 0.9114$
Pro 4	(6, 6, 6)	13	16	2.3e–15	4.8e–08	Degenerate
Pro 4	(1, 32, 42)	12	15	3.3e–15	5.7e–08	Degenerate
Pro 4	(–5, –8, –9)	–	–	–	–	–
Pro 4	(–200, 200, 200)	14	17	2.7e–15	5.2e–08	Degenerate

systems at each iteration. The update of the smoothing parameter is carried out in two cases: the ‘fast step’ is accepted or the condition (13) is satisfied.

The algorithm proposed has global convergence for P_0 -NCP(F). Under the R-regularity condition, the degenerate indices can be estimated correctly near the solution, and the algorithm also has local superlinear convergence. The numerical results of the test problems shows that the algorithm performs well, especially on searching the degenerate solutions.

We believe that this method can be used to solve other problems, e.g. the mixed nonlinear complementarity problems. On the other hand, how to use the identification technique for the NCP(F) under weaker regularity condition need further study. Finally, we will also discuss in further study that whether the active-set strategy can combine with the regularization technique (see [28]) so that more powerful algorithms for NCP(F) can be designed.

References

- [1] M.C. Ferris, J.S. Pang, Engineering and economic applications of complementarity problems, *SIAM Rev.* 39 (1997) 669–713.
- [2] L. Qi, C-differentiability, C-differential operators and generalized newton methods, Tech. Report, School of Mathematics, The University of New South Wales, Sydney, Australia, January 1996.
- [3] T.D. Luca, F. Facchinei, C. Kanzow, A semismooth equation approach to the solution of nonlinear complementarity problems, *Math. Program.* 75 (1996) 407–439.
- [4] L. Qi, Regular pseudo-smooth NCP and BVIP functions and globally and quadratically convergent generalized Newton methods for complementarity and variational inequality problems, *Math. Oper. Res.* 24 (1999) 440–471.
- [5] N. Yamashita, M. Fukushima, Modified Newton methods for solving a semismooth reformulation of monotone complementarity problems, *Math. Program.* 76 (1997) 469–491.
- [6] D. Sun, R.S. Womersley, H. Qi, A feasible semismooth asymptotically Newton method for mixed complementarity problems, *Math. Program.* 94 (2002) 167–187.
- [7] Y.F. Yang, L. Qi, Smoothing trust region methods for nonlinear complementarity problems with P_0 -functions, *Ann. Oper. Res.* 133 (2005) 99–117.
- [8] C. Kanzow, H. Pieper, Jacobian smoothing methods for nonlinear complementarity problems, *SIAM. J. Optim.* 9 (1999) 342–373.
- [9] H. Qi, L. Liao, A smoothing newton method for general complementarity problems, *Comput. Optim. Appl.* 17 (2000) 231–253.
- [10] B. Chen, X. Chen, C. Kanzow, A penalized Fischer–Burmeister NCP-function, *Math. Program.* 88 (2000) 211–216.
- [11] J. Zhang, X. Zhang, A smoothing Levenberg–Marquardt method for NCP, *Appl. Math. Comput.* 178 (2006) 212–228.
- [12] Z. Yu, K. Su, J. Lin, A smoothing Levenberg–Marquardt method for the extended linear complementarity problem, *Appl. Math. Modelling* 33 (2009) 3409–3420.
- [13] Y. Zhou, A smoothing conic trust region filter method for the nonlinear complementarity problem, *J. Comput. Appl. Math.* 229 (2009) 248–263.
- [14] F. Facchinei, A. Fischer, C. Kanzow, On the accurate identification of active constraints, *SIAM. J. Optim.* 9 (1998) 14–32.
- [15] C. Kanzow, H.D. Qi, A QP-free constrained Newton-type method for variational inequality problems, *Math. Program.* 85 (1999) 81–106.
- [16] S.M. Robinson, Strongly regular generalized equations, *Math. Oper. Res.* 5 (1980) 43–62.
- [17] N. Yamashita, H. Dan, M. Fukushima, On the identification of degenerate indices in the nonlinear complementarity problem with the proximal point algorithm, *Math. Program.* 99 (2004) 377–397.
- [18] N. Yamashita, M. Fukushima, The proximal point algorithm with genuine superlinear convergence for the monotone complementarity problem, *SIAM J. Optim.* 11 (2001) 364–379.
- [19] S.M. Robinson, A reduction method for variational inequalities, *Math. Program.* 80 (1998) 161–169.
- [20] F. Facchinei, J. Soares, A new merit function for nonlinear complementarity problems and a related algorithm, *SIAM. J. Optim.* 7 (1997) 225–247.
- [21] A.N. Daryina, A.F. Izmailov, M.V. Solodov, A class of active-set Newton methods for mixed complementarity problems, *SIAM. J. Optim.* 15 (2004) 409–429.
- [22] C. Kanzow, Some noninterior continuation methods for linear complementarity problems, *SIAM. J. Matrix.* 17 (1996) 851–868.
- [23] F. Facchinei, A. Fischer, C. Kanzow, Regularity properties of a semismooth reformulation of variational inequalities, *SIAM. J. Optim.* 8 (1998) 850–869.
- [24] A. Fischer, Solution of monotone complementarity problems with locally Lipschitzian functions, *Math. Program.* 76 (1997) 513–532.
- [25] J.S. Pang, Gabriel, S.A. Gabriel, NE/SQP: a robust algorithm for the nonlinear complementarity problem, *Math. Program.* 60 (1993) 295–337.
- [26] L. Mathiesen, An algorithm based on a sequence of a linear complementarity problems applied to a Walrasian equilibrium model: an example, *Math. Program.* 37 (1987) 1–18.
- [27] C. Oberlin, S.J. Wright, An accelerated Newton method for equations with semismooth Jacobians and nonlinear complementarity problems, *Math. Program.* 117 (2009) 355–386.
- [28] D. Sun, A regularization Newton method for solving nonlinear complementarity problems, *Appl. Math. Optim.* 40 (1990) 315–339.