AMO-Advanced Modeling and Optimization, Volume 14, Number 3, 2012

A Smoothing Perturbed Spectral Projected Gradient Method for Constrained Semismooth Equation with Application [1]

Zhensheng Yu, Chang Liu and Yangjuan Kong

College of Science, University of Shanghai for Science and Technology, Shanghai, 200093, P.R.China

Abstract: By using some smoothing and perturbed techniques, in this paper, we develop a smoothing spectral projected gradient algorithm (SSPG) to solve the system of constrained semismooth equations. The global convergence of the proposed algorithm is established based on an inexact nonmonotone line search. As an application, we consider a smoothing reformulation of KKT systems of the semi-infinite programming (SIP) problem and present the numerical tests to show the efficiency of the SSPG algorithm.

Keywords: Constrained semismooth equations, Smoothing technique, Spectral projection gradient method, Nonmonotone line search, Semi-infinite programming

AMS Classification 90C30 65K05

1. Introduction

In this paper, we consider the systems of bound constrained nonlinear equations:

$$F(x) = 0, \text{s.t. } x \in \Omega = \{l \le x \le u\}. \tag{1}$$

where $l = (l_1, l_2, \cdots, l_n)^T$, $u = (u_1, u_2, \cdots, u_n)^T$ with $-\infty < l_i < u_i < +\infty$ for $i = 1, 2, \cdots, n$, $F : R^n \to R^n$ is defined on an open set U containing the feasible set Ω and is locally Lipschitz continuous.

Systems of nonlinear equations arise in various applications, for instance, some variational inequality and mixed complementarity problems can be converted into the form (1), see for examples [11, 23]. Moreover, the equations with convex constraints come from the problems such as the power flow equations [6,25], chemical equilibrium systems [14,15] and economic equilibrium problems [5]. These comments suggest that the numerical solution of the nonlinear systems of equations with constraints deserve research and experimentation.

The common methods for (1) are optimization-based ones in which the global minimum is zero and the minimizer is the solution of (1). The typical optimization problem in these methods is of the form:

$$\begin{aligned} &\min f(x) = \tfrac{1}{2}\|F(x)\|^2 \\ &\text{s.t. } x \in \Omega. \end{aligned}$$

Based on this reformulation, various numerical methods have been developed in recent years. Among others, the well known methods include Newton type methods, trust region type methods and projection type methods [2, 3, 12, 14,

[1] **AMO-Advanced Modeling and Optimization. ISSN: 1841-4311**

15, 18, 19, 20]. Under some conditions, the methods mentioned above enjoy global and even fast local convergence properties.

In 1988, Barzilai and Borwein [1] introduced a spectral gradient method for unconstrained optimization. Since it requires little computational work, it has received successful applications in unconstrained and constrained optimizations [4, 5, 10] as well as nonlinear equations [6, 8, 9] with smooth mapping $F(x)$. Preliminary numerical tests show that the spectral method works quite well even for large scale problems.

We note that although the spectral methods are very success in smooth problems, there is few literatures available for the nonsmooth problems. Under the monotonity assumption of $F(x)$, Zhang and Zhou [22] introduced some interesting modifications of the spectral methods in order to efficiently handle unconstrained nonsmooth equations. Later, the method was extended to monotone nonsmooth equation with bound constraints [21].

In this paper, we consider extending the spectral projected gradient method to deal with the semismooth equations (1). By combine the smoothing and perturbed technique, we design a smoothing perturbed spectral projected gradient method to solve problem (1). Based on the nonmonotone line search in [24], we establish the global convergence of the proposed method. As an application, we consider a smoothing reformulation of KKT systems of the semi-infinite programming (SIP) problem and present the numerical tests to show the efficiency of the proposed algorithm.

The paper is organized as follows. In Section 2, we give some mathematical preliminaries and describe the smoothing perturbed spectral projected gradient algorithm. In Section 3, we analyze the convergence of the proposed method. In Section 4, we consider the smoothing reformulation of the KKT system of semi-infinite programming, and give the numerical examples to test the efficiency of the algorithm. Some comments are made in the last section.

2. Preliminaries and Algorithm

In this section, we first recall some concepts and properties related to semismooth functions and smoothing functions which will be used later, the detailed description can be seen in [14]. The definition of semismoothness is as follows:

Definition 2.1 Let $H : R^n \to R^n$ be a locally Lipschitz function. We say that H is semismooth at x if

(i) H is directionally differentiable at x and
(ii) for any $h \to 0$ and $V \in \partial H(x + h)$

$$H(x + h) - H(x) - Vh = o(\|h\|).$$

where $\partial H(\cdot)$ denotes the generalized Jacobian in the sense of Clarke [7].

To describe our algorithm, we introduce the definition of smoothing function as follows:

Definition 2.2 Let F be a Lipschitz continuous function in R^n, $D \subset R^n$ be a compact set.
(i) We call $G(t, x) : R \times R^n \to R^n$ a smoothing approximation function of F if it satisfies: (a) $G(0, x) = F(x)$; (b) for any $t > 0, G(t, x)$ being smooth (continuously differentiable) with respect to the second variable $x \in D$; (c)

$$\lim_{t\downarrow 0, z\to x} G(t, z) = F(x).$$

(ii) $G(t,x)$ is called a regular smoothing function of F if for any $t>0, G(t,x)$ is smooth and for any compact set $D \subseteq R^n$ and $\bar{t}>0$, there exists a constant $C>0$ such that for any $x \in D$ and $t \in (0,\bar{t}]$

$$\|G(t;x)-F(x)\| \leq Ct.$$

As suggested in [18], in this paper, we will view t as a variable. Based on the smoothing idea, we consider the corresponding equivalent smoothing systems of problem (1) as follows:

$$\Phi(t,x)=\begin{pmatrix} t \\ G(t,x) \end{pmatrix}=0, \tag{2}$$

$$l \leq x \leq u.$$

here $G(t,x)$ is a smoothing approximation of F(x).

Denote $w=(t,x)$ and a merit function of (2) as

$$\Psi(w)=\frac{1}{2}\|\Phi(t,x)\|^2.$$

The equivalent optimization is defined as:

$$\min \Psi(w), \text{ s.t. } x \in \Omega. \tag{3}$$

According to the definition of $\Phi(t,x)$, it is easy to show that $\Psi(w)$ is continuously differentiable for $t>0$ and

$$\nabla\Psi(w)=\nabla\Phi(w)\Phi(w),$$

where

$$\nabla\Phi(w)=\begin{pmatrix} 1 & 0 \\ \partial_t G(t;x) & \partial_x G(t;x) \end{pmatrix},$$

Define $W=R\times\Omega$ and

$$d_G(1)=P_W(w-\gamma\nabla\Psi(w))-w=\begin{pmatrix} -\gamma\nabla_t\Psi(w) \\ P_X(x-\gamma\nabla_x\Psi(w))-x \end{pmatrix}, \tag{4}$$

where $\gamma>0$ is a constant, P_W is an orthogonal projection operator onto W. Then a stationary point of (3) is characterized by

$$\|d_G(1)\|=0.$$

In what follows, we define our perturbed projected gradient direction: Let $\alpha \in (0,1)$ be a constant and $\beta_0=\alpha\min\{1,\|d_G^0(1)\|^2\}$. for $k=1,2,\cdots$, we define a sequence $\{\beta_k\}$ by

$$\beta_k=\begin{cases} \beta_{k-1}, & if\ \alpha\min\{1,\|d_G^0(1)\|^2\}>\beta_{k-1}, \\ \alpha\min\{1,\|d_G^0(1)\|^2\}, & otherwise. \end{cases}$$

For $\lambda>0$, and $\bar{t}>0$, $\overline{w}=(\bar{t},0)$, at current iteration point $w_k=(t_k;x_k)$ satisfying $t_k>0$, we define the perturbed projected gradient direction by

$$d_G^k(\lambda)=P_W(w_k-\lambda\nabla\Psi(w_k)+\beta_k\overline{w})-w_k. \tag{5}$$

To state the algorithm better, we first introduce the spectral gradient method [1] for unconstrained minimization problem:

$$\min f(x), \quad x \in R^n. \tag{6}$$

where $f: R^n \to R$ is continuously differentiable and its gradient $\nabla f(x)$ is available. Spectral gradient method is defined by

$$x_{k+1}=x_k-\lambda_k\nabla f(x_k), \tag{7}$$

where the scalar λ_k is given by

$$\lambda_k = \frac{\langle s_{k-1}, s_{k-1}\rangle}{\langle s_{k-1}, u_{k-1}\rangle}, \tag{8}$$

where $s_{k-1} = x_k - x_{k-1}, u_{k-1} = \nabla f(x_k) - \nabla f(x_{k-1})$.

In what follows, we describe the smoothing spectral projected gradient (SSPG) algorithm as follows:

Algorithm 2.1

Step 0. Given some constants $0 < \eta_{min} < \eta_{max} < 1$, σ, $\rho \in (0,1)$, $\delta > 0$, $\alpha > 0$, $\bar{t} > 0$ with $\alpha\bar{t} < 1$, $0 < \varepsilon < 1$, let $t_0 = \bar{t}$, $w_0 = (t_0, x_0)$, $\alpha_0 = 1$, choose a positive sequence $\{\varepsilon_k\}$ satisfying

$$\sum_{k=1}^{\infty} \varepsilon_k < +\infty. \tag{9}$$

Set $C_0 = \Psi(w_0)$, $Q_0 = 1$, $k = 0$.

Step 1. Compute $d_G^k(1)$, If $\|d_G^0(1)\| = 0$, stop.

Step 2. If $\alpha_k > 1/\varepsilon$ or $\alpha_k < \varepsilon$, set $\alpha_k = \delta$.
Step 3. Set $\lambda_k = 1/\alpha_k$.

Step 4. (Nonmonotone Line search)

Step 4.1. Let $\gamma_k = \{1, \frac{t_k}{|t_k+\partial_t G(w_k)G(w_k)|}, \frac{\eta\|\Phi(w_k)\|}{\|\nabla\Psi(w_k)\|}, \frac{\eta\|\Psi(w_k)\|}{\|\nabla\Psi(w_k)\|^2}\}$ and compute $d_k = d_G^k(\lambda_k)$, set $\tau_k = 1$.

Step 4.2. Set $w_+ = w_k + \tau_k d_k$.
Step 4.3. If

$$\Psi(w_k + \tau d_k) \le C_k + \varepsilon_k + \sigma\tau\nabla\Psi(w_k)^T d_k, \tag{10}$$

does not hold, then set $\tau_k := \rho\tau_k$, go to Step 4.2, otherwise, go to Step 5.

Step 5. Compute $s_k = w_{k+1} - w_k$, $y_k = \nabla\Psi(w_{k+1}) - \nabla\Psi(w_k)$ and

$$\alpha_{k+1} = \frac{\langle s_k, y_k\rangle}{\langle s_k, s_k\rangle}.$$

Choose $\eta_k \in [\eta_{min}, \eta_{max}]$ and compute

$$Q_{k+1} = \eta_k Q_k + 1,\ C_{k+1} = \frac{\eta_k Q_k(C_k + \varepsilon_k) + \Psi(w_{k+1})}{Q_{k+1}}. \tag{11}$$

Set $k := k + 1$, and go to Step 1.

The following lemma plays an important role in our convergence analysis, the proof can be found in [18].

Lemma 2.1 Assume w_k is not a stationary point of (3), $t_k > 0$ and $d_G^k(\lambda)$ is generated by algorithm 2.1, then we have:

$$\nabla\Psi(w_k)^T d_G^k(\lambda_k) \le -\frac{\lambda_k}{\gamma_k}(1 - \alpha\bar{t})\|d_G^k(1)\|. \tag{12}$$

3. Global convergence

In this section, we analyze the global convergence of Algorithm 2.1. The proof of the following lemma can be obtained similar to Lemma 2.2 in [6].

Lemma 3.1 Assume $\{w_k\}$ is generated by Algorithm 2.1, then we have

$$\Psi_k \leq C_k \leq C_{k-1} + \varepsilon_{k-1}. \tag{13}$$

Based on Lemma 3.1, we can easily to obtain that the algorithm is well defined under certain conditions.

Lemma 3.2 Let $\{w_k\}$ be generated by Algorithm 2.1 and satisfy $t_k > 0$, then the algorithm is well defined.

Define the level set $\mathrm{L} = \{\mathrm{w}|\Psi(\mathrm{w}) \leq \Psi(\mathrm{w}_0) + \varepsilon\}$. Then according to Lemma 3.1, we know that the sequence $\{w_k\} \subseteq \mathrm{L}$. In what follows, we assume the level set is a compact set.

The following lemma shows that if the algorithm does not stop at a stationary point of (3) in any finite step, then we have $t_k > 0$ for every k certain conditions. This result implies that $\Phi(w)$ and $\Psi(w)$ are continuously differentiable at any point generated by Algorithm 2.1.

Lemma 3.3 Let $w_k = (t_k, x_k)$ be generated by Algorithm 2.1, if w_k is not a stationary point of (3) and $\lambda_k < 1$, then for any $k \geq 0$, we have

$$t_k \geq \beta_k \bar{t} > 0. \tag{14}$$

Proof. We prove the proposition by induction.

For $k = 0$, from Algorithm 2.1 and the choice of β_0, it holds

$$t_0 \geq \beta_0 \bar{t} > 0.$$

Suppose that (14) holds for k, we need to prove the conclusion for $k + 1$.
By the computation of $d_G^k(\lambda_k)$ and γ_k, we have

$$(d_G^k(\lambda_k))_t = \lambda_k[-\gamma_k(t_k + \partial_t G_k^T G_k) + \beta_k \bar{t}] \geq -\lambda_k t_k + \lambda_k \beta_k \bar{t}.$$

by the definition of β_k, it is easy to deduce that the sequence $\{\beta_k\}$ is monotone, hence we have

$$\begin{aligned} t_{k+1} - \beta_{k+1}\bar{t} &= t_k + (d_G^k(\lambda_k))_t - \beta_{k+1}\bar{t} \\ &= \geq (1 - \lambda_k)t_k + \lambda_k \beta_k \bar{t} - \beta_{k+1}\bar{t} \\ &\geq (1 - \lambda_k)t_k + \lambda_k \beta_k \bar{t} - \beta_k \bar{t} \\ &= (1 - \lambda_k)(t_k - \beta_k \bar{t}) > 0. \end{aligned}$$

Therefore we have the desired result (14).

Now, we give the global convergence theorem as follows:

Theorem 3.1 Let $\{w_k\}$ be an infinite sequence generated by Algorithm 2.1, then any limit point point of $\{w_k\}$ is a stationary point of problem (3).

Proof. Let w^* be an accumulation point of $\{w_k\}$, and relabel $\{w_k\}$ a subsequence converging to w^*. By the Step 4 in Algorithm 2.1, we have

$$\Psi(w_{k+1}) \leq C_k + \varepsilon_k + \sigma\tau_k \nabla\Psi(w_k)^T d_k.$$

By Lemma 2.1 and the computation of γ_k and λ_k, we have

$$\Psi(w_{k+1}) \leq C_k + \varepsilon_k - \frac{\lambda_k}{\gamma_k}(1 - \alpha\bar{t})\sigma\tau_k \|d_G^k(1)\|^2 \leq C_k + \varepsilon_k - \varepsilon(1 - \alpha\bar{t})\sigma\tau_k \|d_G^k(1)\|^2.$$

From the definition of C_k, we have

$$\begin{aligned} C_{k+1} &= \frac{\eta_k Q_k (C_k+\varepsilon_k)+\Psi_{k+1}}{Q_{k+1}} \\ &\leq \frac{(\eta_k Q_{k+1}+1)(C_k+\varepsilon_k)-\varepsilon(1-\alpha\bar{t})\sigma\tau_k\|d_G^k(1)\|^2}{Q_{k+1}} \\ &\leq C_k+\varepsilon_k-\frac{\varepsilon(1-\alpha\bar{t})\sigma\tau_k\|d_G^k(1)\|^2}{Q_{k+1}} \end{aligned}$$

By the assumption and (9), we have

$$\sum_{k=0}^{\infty}\frac{\varepsilon(1-\alpha\bar{t})\sigma\tau_k\|d_G^k(1)\|^2}{Q_{k+1}} < +\infty. \tag{15}$$

Since $\eta_{max} < 1$, we have

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

Hence by (15), we have

$$+\infty > \sum_{k=0}^{\infty}\frac{\varepsilon(1-\alpha\bar{t})\sigma\tau_k\|d_G^k(1)\|^2}{Q_{k+1}} > \frac{1}{1-\eta_{max}}\sum_{k=0}^{\infty}\varepsilon(1-\alpha\bar{t})\sigma\tau_k\|d_G^k(1)\|^2.$$

Therefore, we obtain

$$\lim_{k\to\infty}\|d_G^k(1)\| = 0 \ or \ \liminf_{k\to\infty}\tau_k = 0.$$

If $\lim_{k\to\infty}\|d_G^k(1)\| = 0$, then we have the desired result.
If $\liminf_{k\to\infty}\tau_k = 0$, we assume that there exists an infinite sequence $\{\tau_k\}_K$ such that

$$\lim_{k\in K, k\to\infty}\tau_k = 0.$$

In this case, from Step 4 of the Algorithm 2.1, there exists an index $\bar{k}$ large enough such that for all $k \geq \bar{k}$, τ_k/ρ fails to satisfy the condition (10), which means

$$\Psi(w_k+\tau_k/\rho d_k) > C_k+\varepsilon_k+\sigma\tau_k/\rho\nabla\Psi(w_k)^T d_k > \Psi_k+\sigma\tau_k/\rho\nabla\Psi(w_k)^T d_k,$$

hence

$$\frac{\Psi(w_k+\tau_k/\rho d_k)-\Psi_k}{\tau_k/\rho} > \sigma\nabla\Psi(w_k)^T d_k. \tag{16}$$

By the mean value theorem, it can be written as

$$\nabla\Psi(w_k+\theta_k d_k)^T d_k > \sigma\nabla\Psi(w_k)^T d_k,$$

where $\theta_k > 0$ is a scalar in the interval $[0, \tau_k/\rho]$, that goes to zero as $k \in K$ goes to infinity. Taking a convenient subsequence such that $d_k/\|d_k\|$ is convergent to d^*, and taking limits in (16) we deduce that $(1-\sigma)\nabla\Psi(w^*)^T d^* \geq 0$. Since $(1-\sigma) > 0$ and $\nabla\Psi(w_k)^T d_k < 0$ for all k, then $\nabla\Psi(w^*)^T d^* = 0$, which means the desired result.

4. Numerical tests

This section makes some numerical examples to illustrate the computational behavior of Algorithm 2.1. We consider the KKT system of a semi-infinite programming (denoted by SIP). The SIP problem is to find $x \in R^n$ such that

$$\min f(x) : x \in X.$$

where $X = \{x \in R^n : g(x,v) \leq 0,\}, \forall v \in V$, $V = [a,b] \subset R^2$ is a nonempty compact subset, $f : R^n \to R; g : R^n \to R$ are twice continuously differentiable functions.

Let

$$V(x) = \{v \in V : g(x,v) = 0\}$$

For the sake of completeness, we give the KKT system of the SIP and its corresponding smoothing reformulation, the detailed discussion can be found in [17][19]. under certain conditions, there exists p positive numbers μ_i such that

$$\nabla f(x) + \sum_{i=1}^{p} \mu_i \nabla_x g(x, v_i),$$

where $v_i \in V(x)$ for $i = 1, 2, \cdots p$ and $p \leq n$. Hence the KKT systems of SIP is as follows:

$$\begin{gathered} \nabla f(x) + \textstyle\sum_{i=1}^{p} \mu_i \nabla_x g(x, v_i) \\ g(x,v) \leq 0, v \in V, \\ \mu_i > 0, g(x, v_i) = 0, i = 1, 2, \cdots, p, \\ \phi(x, v_i) = 0, i = 1, 2, \cdots, p. \end{gathered} \tag{17}$$

where $\phi(x,v) = v - P(a,b,v + \nabla_v g(x,v))$, and the function P is the mid-function defined as

$$(P(a,b,w))_i = \begin{cases} a_i, \ if \ w_i < a_i, \\ w_i, \ if \ a_i \leq w_i \leq b_i, \\ b_i, \ if \ b_i < w_i. \end{cases}$$

In the KKT system (17), x is called a stationary point of the SIP problem. We use a infinite set V_N to approximate V with

$$V_N = \{v_i = a + \frac{i(b-a)}{N} : i = 1, 2, \cdots, N\}.$$

Denote

$$G_N(x) = \max_{v \in V_N} g(x, v).$$

Then the approximate system of (17) can be written as

$$H(z) = 0, \mu \geq 0, y \geq 0,$$

where $z = (x, \mu, v, y) \in R^n \times R^p \times R^p \times R$, and

$$H(z) = \begin{pmatrix} \nabla f(x) + \sum_{i=1}^{p} \mu_i \nabla_x g(x, v_i) \\ G_N(x) + y \\ g(x, v_i), (i = 1, 2, \cdots, p), \\ \phi(x, v_i), (i = 1, 2, \cdots, p). \end{pmatrix} \tag{18}$$

$G_N(x)$ and $\phi(x;v)$ are nonsmooth, but semismooth. In order to use Algorithm 2.1 for solving (18), we choose the following smoothing functions of $G_N(x)$ and $\phi(x;v)$, see [19].

$$G_s(t,x) = \begin{cases} tln(\sum_{i=1}^{N} e^{g_i(x)}/t), \ if \ t > 0 \\ G(x), \ if \ t = 0. \end{cases}$$

$$\phi_s(t,x,v) = \begin{cases} v - [\frac{a+\sqrt{(a-v-(\nabla_v g(x,v)))^2+4t^2}}{2} + \frac{b+\sqrt{(b-v-(\nabla_v g(x,v)))^2+4t^2}}{2}], \ if \ t > 0 \\ v - mid(a,b,v+\nabla_v g(x,v)), \ if \ t = 0. \end{cases}$$

Based on the above reformulation, we can use Algorithm 2.1 to solve the approximation KKT systems of SIP.
We implemented Algorithm 2.1 in Matlab 7.5, where the parameters used in Algorithm 2.1 are set as follows:

$$\sigma = 0.001, \ \rho = 0.5, \ \delta = 2, \ \alpha = 0.5, \ \eta = 0.9, \ \bar{t} = 0.9, \ \varepsilon = 10^{-5}, \ \varepsilon_k = \frac{1}{(k+1)^2}.$$

The starting point u_0 and y_0 for all examples are set $t_0 = \bar{t}, \ u_0 = 0.05e(p), \ y_0 = 0.5$, where $e(p)$ represents $p - order$ unity vector. To keep $t_k > 0$, we set $\lambda_k := 1/\lambda_k$ when $\lambda_k > 1$. The nonmonotone parameter η_k is set as $0.55, \ 0.75$ and 0.8 respectively. The test problems are drawn from [15]:

Problem 1 .

$$f(x) = x_1^2 + x_2^2 + x_3^2, \ g(x,v) = x_1 + x_2 e^{x_3 v} + e^{2v} - 2sin(4v),$$
$$V = [0,1], \ p = 1, \ (x_0, v_0) = (1,1,1,1).$$

Problem 2.

$$f(x) = \frac{1}{3}x_1^2 + \frac{1}{2}x_1 + x_2^2, \ g(x,v) = (1 - x_1^2 v^2)^2 - x_1 v^2 - x_2^2 + x_2,$$
$$V = [-1,1], \ p = 1, \ (x_0, v_0) = (-1,-1,1).$$

Problem 3.

$$f(x) = x_1^2 + (x_2 - 3)^2, \ g(x,v) = x_2 - 2 + x_1 sin(v/x_2 - 0.5),$$
$$V = [0,10], \ p = 1, \ (x_0, v_0) = (1,-1,1)$$

Problem 4.

$$f(x) = \frac{1}{2}x^T x, \ g(x,v) = 3 + 4.5sin(4.7\pi(v - 1.23)/8) - \Sigma_{i=1}^{n} x_i v^{i-1},$$
$$V = [0,1], \ n = 10, \ p = 1, \ (x_0, v_0) = (1, 1, \cdots, 1.)$$

The computed results are reported in Table 1, where p is the guess of number in active set at solution point; NH and NdH represent the computing number of function and its derivative defined in constrained equations; Ndis indicates the dividing number for region V ; CPU is the total cost time (in second) for solving SIP problems; $d_G(w_k)$ and $f(x_k)$ indicate the final values of the projected gradient and the objective function in SIP. The results reported in Table 1 shows that Algorithm 2.1 performs well for these test problems.

5. Final remarks

In this paper, we extend the spectral projected gradient method to nonsmooth equation and establish the global convergence. Compared with the existing methods such as semismooth and smoothing projected Newton methods, our method does not need to solve a system of linear equations at each iteration. The numerical tests for the KKT systems of the semi-infinite programming show the efficiency of the proposed method.

Table 1. Tests results

Problem	p	NG	NdG	N_{dis}	CPU	$\|d_G(w_k)\|$	$f(x_k)$
1: $\eta^k = 0.55$	1	12	12	1280	6.9595	2.020e-11	0
$\eta^k = 0.75$	1	10	10	1280	5.1045	1.077e-11	0
$\eta^k = 0.8$	1	8	8	1280	4.3547	6.359.0e-13	1.4634
2: $\eta^k = 0.55$	1	39	39	1280	24.3748	1.048e-14	0.0611
$\eta^k = 0.75$	1	28	28	1280	17.0849	1.388e-17	0.0674
$\eta^k = 0.8$	1	25	25	1280	15.1721	2.289e-42	0.6687
3: $\eta^k = 0.55$	1	5	0	1280	107.7571	8.863e-6	3.5073
$\eta^k = 0.75$	1	4	0	1280	105.4672	3.316e-6	3.4084
$\eta^k = 0.8$	1	5	0	1280	111.4100	1.106e-10	3.6524
4: $\eta^k = 0.55$	1	41	44	1280	101.2510	1.757e-6	5
$\eta^k = 0.75$	1	42	42	1280	101.3971	2.389e-6	5
$\eta^k = 0.8$	1	35	41	1280	102.4249	2.497e-6	5

Acknowledgments

This work are supported by Innovation Program of Shanghai Municipal Education Commission(No.10YZ99), National Natural Science Foundation of China (No.11171221) and National Project and Liberal Base Cultivate Fund of University of Shanghai for Science and Technology(No.12GXM03).

References

[1] J. Barzilai, J. M. Borwein. Two point stepsize gradient methods. IMA J Numer Anal 1988; 8: 141-148.

[2] S. Bellavia, M. Macconi and B. Morini. An affine scaling trust-region approach to bound-constrained nonlinear systems, Appl Numer Math 2003; 44: 257-280.

[3] S. Bellavia and B. Morini. Subspace trust-region methods for large bound-constrained nonlinear equations. SIAM J Numer Anal 2006; 44: 1535-1555.

[4] E. G. Birgin and J. M. Martinez. Large-scale active-set box-constrained optimization method with spectral projected gradients. Comput Optim Appl 2002; 22: 101-125.

[5] E. G. Birgin, J. M. Martinez and M. Raydan. Nonmonotone spectral projected gradient methods on convex sets. SIAM J Optim 2000; 10: 1196-1121.

[6] W. Cheng and D. H. Li. A derivative-free nonmonotone line search and its application to the spectral residual method,. IMA J Numer Anal 2009; 29: 814-825.

[7] F. H. Clarke. Optimization and Nonsmooth Analysis. John Wiley and Sons, New York, NY, 1983.

[8] W. La Cruz and M. Raydan. Nonmonotone spectral methods for large-scale nonlinear systems. Optim Method Soft 2003; 18: 583-599.

[9] W. La Cruz, J. M. Martinez and M. Raydan. Spectral residual method without gradi ent information for solving large-scale nonlinear systems of equations. Math Comput 2006; 75: 1449-1466.

[10] Y. H. Dai and R. Fletcher. Projected Barzilai-Borwein methods for large-scale box- constrained quadratic programming. Numer. Math 2005; 100: 21-47.

[11] C. Kanzow. Strictly feasible equation-based method for mixed complementarity problems. Numer Math 2001; 89: 135-160.

[12] D. N. Kozakevich, J. M. Martinez and S. A. Santos. Soving nonlinear systems of equations with simple bounds. Comput Appl Math 1997; 16: 215-235.

[13] D. H. Li and M. Fukushima. A modified BFGS method and its global convergence in nonconvex minimization. J Comput Appl Math 2001; 129: 15-35.

[14] L. Qi, J. Sun. A nonsmooth version of Newton's method. Math. Prog 1993; 58: 353-367.

[15] L. Qi, X. J. Tong and D. H. Li. Active-set projected trust region algorithm for box-constrained nonsmooth equations. J Optim Theory Appl 2004; 120: 601-625.

[16] L. Q. Qi, C. Ling, X. J. Tong, G. L. Zhou. A smoothing projected Newton-type algorithm for semi-infinite programming. Comput Optim Appl 2009; 42: 1-30.

[17] M. Raydan. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. SIAM J Optim 1997; 7: 26-33.

[18] X. J. Tong, S. Z. Zhou. A smoothing projected Newton-type method for semismooth equations with bound constraints, J Industirial Management Optimization 2005; 1: 235-250.

[19] M. Ulbrich. Nonmonotone trust-region methods for bound-constrained semismooth equations with applications to nonlinear mixed complementarity problems. SIAM J Optim 2000; 11: 889-917.

[20] C. Wang, Y. Wang and C. Xu. A projection methods for a system on nonlinear monotone equations with convex cosntraints. Math Meth Oper Res 2007; 66: 33-46.

[21] Z. S. Yu, J. Lin, J. Sun, Y. H. Xiao, L. Y. Li, Z. H. Li. Spectral gradient projection method for monotone nonlinear equations with convex constraints. Appl Numer Math 2009; 59: 2416-2423.

[22] L. Zhang and W. Zhou. Spectral gradient projection method for solving nonlinear monotone equations. J Comput Appl Math 2006; 196: 478-484.

[23] Y. B. Zhao and D. Li. Monotonicity of fixed point and normal mapping associated with variational inequality and is applications. SIAM J Optim 2001; 11: 962-973.

[24] H. Zhang, W. W. Hager. A nonmonotone line search technique and its application to unconstrained optimization. SIAM J Optim 2004; 4: 1043-1056.